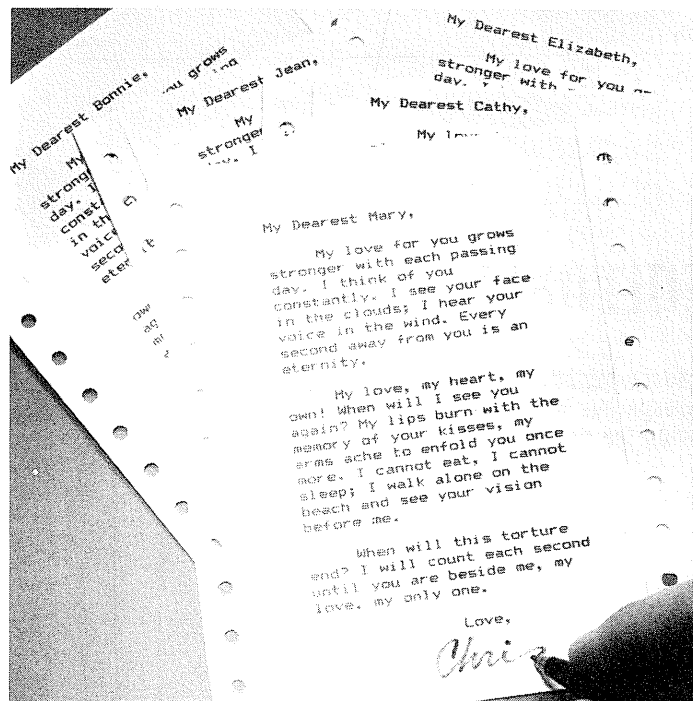


We're
known
for our
fine print.



Epson.

The type you get out of most printers you wouldn't send to your maiden aunt, much less use for your *important* correspondence. And up to now, in order to get a dot matrix hardcopy you could really call correspondence quality, you had to spend on the high side of a thousand bucks.

Not any more.

The Epson MX-80 challenges any dot matrix printer anywhere to match our type at our price. Or even come close.

Our emphasized print mode gives you a tack-sharp, clean, easy-to-read face with true descenders—at a fraction of the price of daisy wheel printers. We give you a user-defined choice of twelve different weights and sizes of letters in 40, 80, 66 or 132 columns. We give you adjustable tractors so you can do anything from labels to memos to manuscripts. Fast and clean.

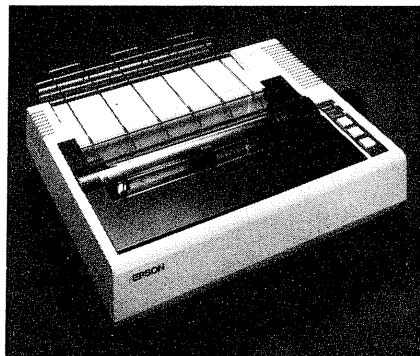
But if you think print quality is the only thing we have

to sell, you're wrong. The MX-80 may be the most revolutionary printer to come out in the past ten years.

For starters, it features the world's first *disposable* print head—after it's printed between 50 and 100-million characters, just throw it away. A new one costs less than \$30 and you can change it yourself with one hand. Plus, the MX-80 prints bidirectionally and 80 CPS with a logical seeking function to minimize print head travel time and maximize throughput. Finally—and this is the

best part—you can buy an MX-80 right now for less than \$650.

And that's what we call a lot of fine print for the money.



EPSON
EPSON AMERICA, INC.

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

EDITORIAL

All of us at 80-U.S. wish our readers, our advertisers, our dealers and supporters a very Merry Christmas and a prosperous New Year. May all of your problems turn into interesting challenges during the coming year!



80-U.S. JOURNAL

Editor/Publisher

I Mike Schmidt

Sales/Promotion

Margaret Farrell

Associate Editor

Terry R Dettmann

Contributing Editors

Phil Pilgrim

Larry S Panattoni

Bill Schroeder

Spencer Hall

R C Bahn

Art/Layout/Design

Grace McNamara

Circulation

Margaret Murray

Typesetting

Jill Blackburn

Reviewers

Cameron C Brown

Pat Perez

W W Harper II

Printed by

Craftsman Press

Seattle, Washington

DISTRIBUTORS

Canada

Micromatic Systems Inc

1303 Powell Street

Vancouver, BC Canada

V5B 1G6

Europe

Hofacker Verlag

Tegernseer Strasse 18

D-8150 Holzkirchen/Obb

West Germany

United Kingdom

The Software House

146 Oxford Street

London, W.1 England

© 1981 80-Northwest Publishing Inc. All rights reserved. Reproduction for other than personal, non-commercial purposes is prohibited. No patent liability is assumed with respect to the use of the information contained herein. While every precaution has been taken in the preparation of this publication, the publisher assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of any information contained herein. Please address all correspondence to: 80-U.S. Journal, 3838 South Warner Street, Tacoma, Washington 98409 (206) 475-2219

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

Writers/Authors: We constantly seek material from contributors. Send your material (double space, upper/lower case please) and allow approximately 4 weeks for review. You may send programs or text files on diskette or tape, they will be returned provided you include return postage. Generous compensation is made for non-trivial works which are accepted for publication. The Journal pays on acceptance rather than on publication.

The Cover

Our cover design and the drawing on it are the work of Ken Dankwardt, of Dankwardt Ink, Seattle Washington

The 80-U.S. JOURNAL Vol IV Number 1 Jan/Feb 1981

ISSN Publication #0199-1035

Published bimonthly in Jan, Mar, May, Jul, Sep and Nov by 80-Northwest Publishing Inc., 3838 S. Warner St., Tacoma, WA 98409

Subscription price in the United States is \$16. one year, \$31. two years and \$45. three years.

Canadian subscriptions: Contact Micromatic Systems Inc 1303 Powell St., Vancouver, BC Canada V5B 1G6. All other subscriptions: \$30. per year, remitted in US funds

All except US, Canada and Central America are sent Airmail.

Printed in the United States of America

POSTMASTER: Please send change of address form 3579 and undelivered copies to 80-U.S. Journal, 3838 S. Warner St., Tacoma, WA 98409

Return Postage Guaranteed
Second Class Postage Paid at Tacoma, Washington and at additional entry points.



THE TRS-80 USERS JOURNAL

*TRS-80 is a trademark of the Tandy Corporation

Volume IV Number 1

January/February 1981

IN THIS ISSUE

15	Time	Staff
16	A Calendar Maker Program	T R Dettmann
20	Day of Week/Days between Dates program	Anon
21	A Functional Subroutine Library	David R Pepple
26	Nine Z-Subroutines	Spencer Hall
120	BREAK - with JKL	Al Domuret
85	Build Your Own Printer Interface	Larry Panattoni
110	Files and Foibles	T R Dettmann
88	A Graphics Editor	William K Mason
48	A Pocket Computer Application	Bill Vick
38	Inside VTOS 4.0 and NEWDOS80	Pete Carr
34	3 Machine Language Utilities for the Basic Programmer .	J. McKenney
92	The Making of a Computer Program	Mike Schmidt
115	The Model III Makes its Debut	Staff
117	Two Computers Feed One Printer	Gary Rittenbach
96	Profile II for Model II - The ultimate Data Base?	Bill Schroeder
100	Playmate - A Computer that Talks!	Don Scarberry
78	Estimate Programming Jobs (with a Program!)	T R Dettmann
74	Slalom Run! A Fun & Games Ski Program	Greg Perry
65	How to Read Disk Directories	Debbi Tesler
42	How to Make Versafile more Versatile	Mike Zielinski

Reviews

124	The Basic Compiler	Truman Krumholz
104	DATORG	T R Dettmann
108	Membership Billing Program	T R Dettmann
108	Lemonade or Champagne?	Woodrow W Harper II
72	LNW Expansion Interface	Yvon Kolya

Departments

128	Advertiser Index	
18	Anatomy of the Calendar Program	R C Bahn
128	Back Issue Availability	
30	Captain 80	Bob Liddil
2	Editorial	
8	Interface	
10	Items at Random	
4	Letters to the Editor	
13	New Products	
118	Notes	
52	System/Command	Phil Pilgrim
125	Unclassified Ads	
106	View from the Top of the Stack	Staff

LETTERS

In response to your editorial for the Sep/Oct 1980 issue, I agree heartily with your comments about "copying" which ought better be known by its proper name of theft. But I want to take serious issue with your use of the term excessive when referring to the prices of custom work.

The freedom that a microcomputer offers includes privacy, more security than a main frame, and the ease of custom programming to tailor all software for the user's needs rather than the user tailoring his uses for what the computer can do. On large systems and minis the software can often exceed the price of the hardware. With micros this is usually not the case even for custom software. And the figures are so much lower in any event.

Micro users have been taught to believe that software is cheap and reliable. Neither is the case, obviously. Good software reflects time and energy as well as creativity. It must cost money. It seems that micros are sold by Radio Shack and other outlets with the intimation that software "only costs a few more bucks." In fact software is often given away with a sale showing the customer that its value is negligible.

I'm addressing myself to serious business uses here. Database managers, bookkeeping programs, and other serious business programs can come packaged at high but reasonable prices. Custom work to produce reports needed by the user must cost more. Even at that they are a bargain if amortized over the years. Many hours may be spent on a serious game, but those hours are multiplied many times over for business applications. It seems that your use of the word "excessive" continues this feeling that anything over \$14.95 is over priced for a game or \$150 is too much for a business program. I realize you sell packaged software, but custom work is the real freedom of the microcomputer. For the owner who cannot do his own work, custom workers like myself don't need to have words like excessive used to reflect the many hours we must charge for.

I've been a subscriber from the start, since I've always found that serious hobbyists find the secrets of any system long before the rest of us professionals. I'm astounded that more of my fellow custom programmers don't do the same.

John Revelle
Rohnert Park, CA

(The implication of that editorial was that if every program released will be copied, then custom programmers would have to charge excessive prices, knowing full well that one copy is all they could sell. Ed.)

I've seen comments from both sides of the "Pirating" issue, and I'd like to add a few comments of my own.

There is a middle ground to all of this, and I have to believe that's where I'm at. I don't

believe in "libraries", and I certainly respect the time spent by authors of quality software. I have written several short programs for my own business ... so I empathize.

On the other hand, a good friend and I have gone into computer as a "team", or partnership... We have no way to review the programs that would appear to interest us, so we each subscribe to several magazines, and we each spend a good amount on software. And then we share the programs that are worth having.

I feel that after the useless or misrepresented programs are culled out, our average cost per program is relatively fair for value received. If we each had to do this as an individual, I couldn't say that.

My own feelings are that the market is increasing in quality as the sophistication (and experience) of the programmers increases. Still, there are a great many whose only qualification is that they took time to write a program. I suspect that's contributing to the disdain that some computerists have for copyright notices.

Another comment is on the tone of the editorializing lately. Pirating is wrong, but sermonizing to a group of readers of the intellect yours have (check your demographics) certainly must be resented. How about the tone set by Dennis Kitsz, who says about his fine program "Keepit": "Reproduction or unauthorized distribution of this program, if not strictly speaking illegal, is obnoxiously immoral and profoundly dubious. So please don't." I think that is all that is needed.

One final thing. After I looked through several issues of 80-U.S., I intend to put my money where my mouth is. Please find enclosed my subscription request.

Ron Manor
Olympia, WA

We have been members of a software library for several months, and I feel obligated to respond to your editorial in the Sep/Oct 1980 issue.

We live in a remote area and know few people with TRS-80 computers, and there is no club within a reasonable distance. Until now we have therefore purchased all software on the basis of advertisements and guesswork. Needless to say, we have made mistakes.

We therefore felt a software library would be an ideal solution for us. We have a simple method of using it honorably. The rented tapes are here for two weeks. Typically, during that time they are used heavily at first. By the end of the rental period, many of the tapes are no longer selected by any member of the family. I mail them back and they are not even missed. If a program cannot sustain interest for two weeks, then clearly it has no merit for us.

There is no point in making a copy, even if I wanted to, because no one wants to use the program.

A few programs have been exceptions. For example, Galactic Empire really delighted me, but it was so annoying not to be able to save the game in progress. As it takes a long time to play, I decided to buy it anyway, because I enjoyed it more than any game I have ever tried. I saw ads advertising all three games in this series on a disk at a savings over the single game price. Because I had rented Empire and knew I liked it, I felt confident I would like them all and ordered the disk. To my surprise and delight, I found a save game feature was included in the purchased version (apparently an updated version from the one I had rented). Note that I never would have bought the disk with all three games had I not first rented the program. The rental library served an important function, it kept me from wasting money on something I didn't enjoy.

Granted not all library users have sufficient integrity to buy the programs they find worthwhile, and I agree your concern is legitimate. Nonetheless, I wanted to express our appreciation for the way a software library has helped us to stop wasting our software money.

Wynne Keller
Solon, ME

I agree with the points you raised in your editorial in the Sep/Oct 1980 issue regarding software piracy, but I think that part of the present problem was caused by the software authors themselves. I have felt ripped off many times after buying software that looked good on paper but turned out to be not worth a damn. Like the hawkers selling Mt. St. Helens ash to tourists, the software authors rushed out to "milk" the public for all they could get in the "early days". As a result, the temptation to help one's self to any available software is hard to resist. I generally won't purchase commercial software anymore unless I can see it demonstrated first. I have also found that "money-back" guarantees aren't worth much; the vendors stall for as long as they can before returning your money (although God, do they scream for payment from you in the first place). So let us not be too quick to put all of the blame on the user community ...

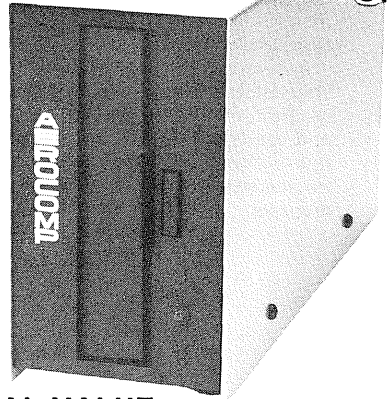
R B Reyes, Ph.D.
Pittsburgh, PA

(These letters are but a small representation of the mail received regarding the "Piracy problem".

There are still two distinct sides to this problem:

1. An author is entitled to fair compensation for his work. This compensation is not available to the author if the work is copied at liberty by anyone for their friends. The author's case can be stated very simply. How would you like to work all week and have the person next to

UNIQUE!



40 & 80 Track "FLIPPY" Drives for TRS-80*

THE BEST!

● **40-Track "FLIPPY" \$349.95**
(Model 40-1) Single-sided, "FLIPPY", 48TPI. (40 Track; single density unformatted 125K bytes/side; double density unformatted 250K bytes/side).

● **80-TRACK "FLIPPY" \$459.95**
Single-sided, "FLIPPY", 96TPI. (80 track; single density unformatted 250K bytes/side; double density unformatted 500K bytes/side).

All models are capable of single or double density and are complete with power supply and silver enclosure.

* SPECIAL PACKAGES *

#1 40-Track FLIPPY drive
2-Drive cable
Newdos/80
Freight & Ins. \$459.00 (reg \$528.00)

#2 80-Track FLIPPY drive
2-Drive cable
Newdos/80
Freight & Ins. \$569.00 (reg \$638.00)

#3 TWO (2) 40-Track FLIPPY drives
4-Drive cable
Newdos/80
Freight & Ins. \$785.00 (REG \$893.00)

#4 TWO (2) 80-Track FLIPPY drives
4-Drive cable
Newdos/80
Freight & Ins. \$999.00 (reg \$1113.85)

● **DISK OPERATING SYSTEMS**
Newdos (40 track) \$109.00
Newdos/80 (40 track) \$149.00
VTOS 4.0 \$125.00
DOSPLUS \$99.95

● **DISKETTES, SOFT SECTOR, 5 1/4" (box of 10)**
Single-sided, single density \$29.95
Double-sided, double density \$39.95

● **CABLES**
2-drive \$24.95
4-drive \$34.95

WRITE AEROCOMP TODAY FOR MORE VALUES !!!

FREE TRIAL OFFER

Order your AEROCOMP Disc Drive and use it with your system for up to 14 days. If you are not satisfied for ANY REASON (except misuse or improper handling), return it, packed in the original shipping container, for a full refund. We have complete confidence in our products and we know you will be satisfied! ORDER TODAY!

WARRANTY

We offer you a 120 day unconditional warranty on parts and labor against any defect in materials and workmanship. In the event service, for any reason, becomes necessary, our service department is fast, friendly and cooperative.

100% TESTED

AEROCOMP Disc Drives are completely assembled at the factory and ready to plug in when you receive them. Each drive is 100% bench tested prior to shipment. We even enclose a copy of the test checklist, signed by the test technician, with every drive. AEROCOMP MEANS RELIABILITY!

ORDER NOW!!

To order by mail, specify Model Number(s) of Drive, cable, ect. (above), enclose check, money order, VISA or MASTERCHARGE card number and expiration date, or request C.O.D. shipment. Texas residents add 5% sales tax. Add \$5.00 per drive for shipping and handling. Please allow 2 weeks for personnel checks to clear our bank. No personnel checks will be accepted on C.O.D. shipments-cash, money orders or certified checks only. You will receive a card showing the exact C.O.D. amount before your shipment arrives. Be sure to include your name and shipping address. WE SHIP PROMPTLY! In the event there is a slight delay, you will be notified of the shipping date and we will NOT charge your bankcard until the day we ship!

REAL VALUE

AEROCOMP offers the best value in microcomputer disc drives on the market today! Reliability, features and cost tough to beat. We deliver...and we stand behind our products, as evidenced by the only FREE TRIAL OFFER in the industry. Examine your systems needs and order today!

MYSTERY REMOVED

There appears to be some confusion in the terminology used to describe disc drives and their features. Here's what we mean:

● **FLIPPY** Allows the use of both sides of a diskette with a single-headed drive by simply turning the diskette over (model 40-1&80-1).

● **TRACK DENSITY** Specified in tracks per inch (TPI). Refers to the number of tracks per radial inch on the diskette. Typically 48 TPI=40 usable tracks and 96 TPI=80 usable tracks.

● **DOUBLE DENSITY** Refers to recording density in bits per inch (bpi). Typically single density means data can be recorded up to 2,938 bpi; double density means data can be recorded up to 5,876 bpi.

● **DOUBLE SIDED** Refers to number of read/write heads. Single-sided is one head, read/write one side only; double-sided is dual heads allowing read/write operations on both sides of the diskette. A double sided drive appears as two separate drives to the controller.

● **ACCESS TIME** The time required for the head to move from one track to the next. Typically 5 to 40 milliseconds (ms).

COMPARE AND BUY AEROCOMP!

	"FLIPPY"	ACCESS TIME (track to track)	HEAD LOAD SOLENOID	DISC EJECTOR	CAPACITY (unformatted single density)	EASY-ENTRY DOOR	FREE TRIAL
AEROCOMP	YES	5ms.	YES	YES	250K bytes (both sides)	YES	YES
RADIO SHACK*	NO	40ms	YES	NO	109K bytes	NO	NO
PERCOM	YES	25ms	YES	NO	250K bytes (both sides)	YES	NO
MPI	NO	5ms	YES	YES	125K bytes	YES	NO
SHUGART	NO	40ms	YES	NO	109K bytes	NO	NO
TANDON	NO	5ms	NO	NO	125K bytes	NO	NO

Factual material from current manufacturer's data sheets is believed reliable but cannot be guaranteed. Comparing Aerocomp Model 40-1 to similar models

The TRS-80* expansion interface limits the track to track access time to 12ms

*Trademark of Tandy/Radio Shack

**CALL TOLL FREE FOR FAST SERVICE
(800) 824-7888, OPERATOR 24**
FOR VISA / MASTERCHARGE / C.O.D. ORDERS
California dial (800) 852-7777, Operator 24. Alaska and Hawaii dial (800) 824-7919, Operator 24.
TOLL FREE LINES WILL ACCEPT ORDERS ONLY!
For Applications and Technical information, call (214) 337-4346 or drop us a card.

Dealers inquiries invited

AEROCOMP
Redbird Airport, Bldg. 8
P.O. Box 24829
Dallas, TX 75224

Letters

you receive your paycheck?

2. As these letters show, the user has to take a chance. There is no good way to see what you are buying before you pay your money, after which it may be too late. Vendors are reluctant to refund money, for the simple reason that the customer can easily copy the program and then return it, saying that it doesn't work, or he doesn't like it.

Any scheme to satisfy both the author and the consumer would be appreciated. If you have one in mind, send it in, we will give it publicity and see how well it can stand up.)

Help! I just purchased an old IBM Selectric typewriter with I/O capabilities. I am hoping to interface it to my Level II TRS-80, and would appreciate hearing from anyone with some information on it and how it can be done. The machine has a 34-pin connector, arranged in alternating rows of 9 and 8, and was previously used by Allegheny Airlines as part of their reservation system. Thanks for any help you can provide.

Steven Greene
6300 Rockhurst Road
Bethesda, MD 20034

... much has been said about PASCAL being the "computer language of the future", etc., etc. It's a nice language, no question. But there are problems with implementing it on micros. PASCAL was never intended for an interactive environment, much less for a microcomputer environment; it runs most efficiently on monster machines with gigabyte memories and a batch queueing system. The two fullscale compilers available for the TRS-80 are hopeless memory hogs even though neither implements the full language; the UCSD system compiler leaves practically no room for user programs, and PASCAL/MT, while not using as much memory during compilation, is annoyingly slow. Unless someone can come up with an efficient compiler (or a way of implementing 256K memories on the TRS-80) PASCAL is simply not going to replace Basic as the micro language choice.

R B Reyes, Ph.D
Pittsburgh, PA

I own a TRS-80 Level II with 16K expansion interface and 1 disk drive. Yesterday I went to the 3rd Personal Computer World Show held at the Cunard Hotel, Hammersmith, London. From one of the many stands there I purchased my first 80-U.S. I couldn't wait to get home to start reading it so I began on the train and promptly went past my stop!

I found your magazine very informative

and written in such a way that non-technical idiots like me can understand it. There are many computer magazines on sale in the U.K., most of which are too technical. As a TRS-80 user I like to be able to read a magazine about that machine. In the U.K. one mainly reads about the PET!

I visited the states earlier this year (Florida, to be precise) and among the wonderful things I saw out there I found a TANDY, (sorry, RADIO SHACK) computer store. We have them in the U.K. where everything is twice the American price. (Radio Shack operates under the name TANDY in the U.K.). We are well behind on what's new and items I saw in the store in Florida will not be on sale in this country until later this year.

I was so impressed with your (magazine) that I would like to take a year's subscription.

W R Luxton
Romford, Essex, England

"Notes" for the Sep/Oct 1980 issue tells how to install PEEK and POKE on the Model II. I have been wanting this ever since we purchased a Model II. There is a change that I had to make in the first of your patches to make it work for us. Some of your readers may have to make the same change.

The patch as given in NOTES was:

PATCH BASIC A=6771 F=C5CD2061

For our system, I had to enter the patch as follows:

PATCH BASIC A=6771 F=AFCD2061

With this change everything is great.

I would like to make contact with others who are using the Phase One Systems operating system called OASIS on a Model II. I find it makes our micro run almost like our IBM SYSTEM/34.

Donald M Dealy
EDP Director
231 Washington Street
S Attleboro, MA 02703

(The change you mention was a typo on our part, and was listed in Corrections in the following issue. We only repeat it here because there was a large amount of interest in that change, and some may have missed the corrections given in the Nov/Dec 1980 issue. Ed)

I had to write and tell you how much I appreciate your willingness to send me the NFL-PIX program and wait for the money to arrive afterwards. There aren't many places dealing in software that I know of that will do that. By sending it ahead you allowed me to get my scores into the program and not be three weeks behind the games.

...I also have the Radio Shack upper/lower case modification installed in my computer, and that is the main culprit for why the NFL-PIX program will not print properly. After many hours of frustrated

attempts to alter the program to print, no luck!...After I had loaded the upper/lower case modification driver routine into the computer and then the NFL-PIX, it worked just fine and the printer is now able to receive the output.

It seems that the character generator chip if not driven conflicts with the way the program takes characters from the screen and sends them to the printer. I hope that this will be of help to anyone who is having trouble with their system and the NFL-PIX program....

Bob Walters
Eugene, OR

I hope you can help me. I am looking for TRS-80 computer related peripherals, including programming books, software, disk system, and a selectric style printer.

Let me back up and introduce myself. My name is Thomas Martin, and I am a prisoner. For fifteen years I was deeply involved in electronic communications technology. Prior to my incarceration, I owned and operated a firm that designed, built to order and installed electronic hardware on pleasure craft. Unfortunately, I became involved in a crime of passion and there was a death. I am sentenced to life in prison.

When I arrived in prison, I found the remnants of an electronics lab in the school. I was able to repair and renovate some of the equipment, but much of it was beyond repair, outdated, or outmoded. I was able to convince the school administration to buy a TRS-80 computer, but then the funds dried up and we have a computer and nothing more.

My ambition is to set up a complete computer complex to teach other prisoners, particularly those with short sentences, the basics of computers so that when they return to society they will have a start in the right direction in a rapidly growing area of employment.

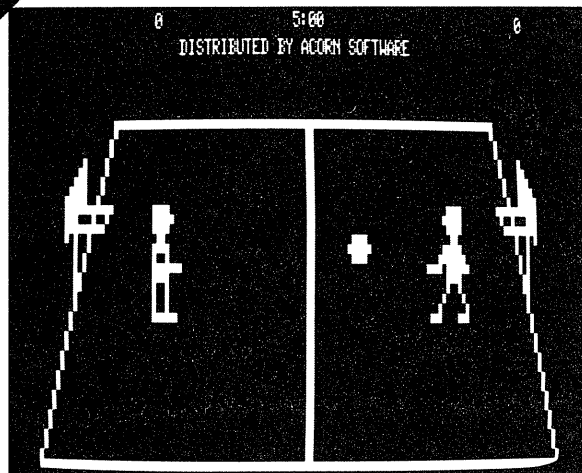
Could you suggest anyone who might be able to donate, perhaps for a tax write-off, any of the equipment or supplies useful to the TRS-80? Your first reaction may be that the state should pay for state prisoners' needs. This is perfectly true. However, the state seems only interested in warehousing human beings; later turning them out into society on their parole date much worse than they came in. Prisoners must help each other. Individuals like me who have the intelligence, training and inclination to help must teach the deprived ones useful skills. Could you help me to help others?

Thomas Martin
State Correctional Institute
PO Box 244
Graterford, PA 19426

(We'll start the ball rolling with a 3 year subscription to our Journal - anyone else want to help? Ed)

FOR
TRS-80*

NEW! BASKETBALL



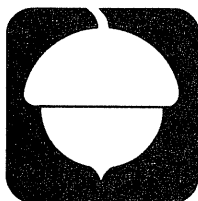
by John Allen

New machine language action game, with sound, from the author of the acclaimed "PINBALL!"

You have to be fast to keep up with the action as you try to outscore your opponent in five minutes of one-on-one basketball. Compete against a friend or your computer.

Steal the ball, duck around your opponent and slant toward the basket for a lay up! The graphics are based on a 3-dimensional depiction of a basketball court, and ball dribbling sounds add to the realism. It's all there but the cheers—so real you'll wonder how the ball keeps from coming through the screen of your TRS-80! Dribble, Dribble!

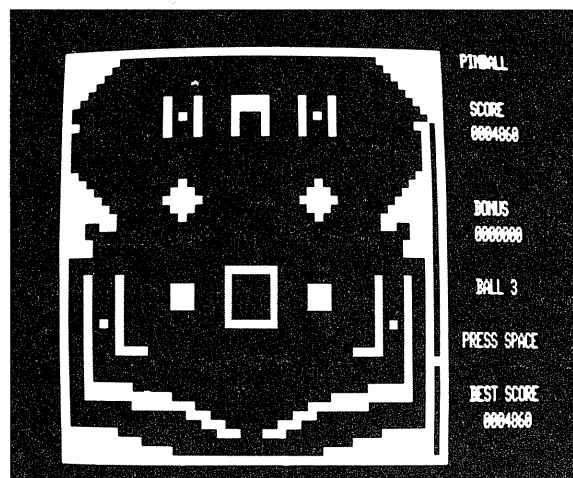
Available for Level II, 16K. \$14.95 for tape, \$20.95 on disk.



Acorn
Software Products, Inc.

634 North Carolina Avenue, S.E., Washington, D.C. 20003

FAST, REAL-TIME ACTION WITH SOUND



PINBALL

by John Allen

Get your flipper fingers ready for action in this real-time, machine language game.

Lots of sound and flashing graphics make this fast action game so much like the real thing that you'll have to remind yourself not to shake your TRS-80*. Choose from five playing speeds to match your skill — but be prepared for a lot of practice if you ever hope to master the fastest speed.

Can you beat your friends' scores? Will you avoid the dreaded "Bermuda Square?" Get PINBALL today and find out. Available for \$14.95 on tape or \$20.95 on disk.

*TRS-80 is a trademark of Tandy Corp.

These and other popular Acorn programs are available now at fine computer stores. Ask for them.

DEALER INQUIRIES INVITED

INTERFACE

80-U.S. talks with Roger Billings of LOBO Drives International about the problems connected with producing hardware and software operating systems.

In the Fall of 1979 Lobo Drives International designed an expansion interface that would revolutionize the TRS-80 Model I market. The expansion interface (the Model LX80) would turn the TRS-80 Model I from a home computer with a maximum disk capacity of about 356 kilobytes into a mass storage unit capable of storing well over 40 megabytes, suitable for business use, yet priced below the cost of a standard business system.

Unlike the TRS-80 expansion interface which allows the use of up to four single sided, single density mini floppies, the Lobo Drives LX80 allows use of up to four mini floppies, four 8" floppies (single or double sided) and multiple hard disks (5¼", 8" and/or 14"), all at the same time!

Included with the LX80 are two additional parallel ports, one being a Centronics printer port and the other an expansion/screen printer port. Also available as options are dual serial ports for the use with the popular printers and modems. Thirty-two kilobytes of Random Access Memory can also be added.

However, the hardware alone was not enough. A new operating system had to be developed to make the LX80 a reality. Many versions were examined, but none were suitable for the use intended. Thus, an unexpected delay occurred in getting the LX80 to the marketplace, until a suitable operating system had been created expressly for the LX80 which allowed full use of the hardware capability.

80-U.S.: Since Lobo Drives is in the business of selling disk drives, why start a project with a controller such as the LX80?

Lobo: Lobo Drives was known primarily for packaging disk drives (5¼" and 8" floppies). At the time, the reason for getting into the controller market was that we felt the original manufacturers such as Apple, Tandy, and some of the S-100 people were not meeting all of the needs of the personal computer users. There were a number of people who had gone beyond games and

balancing their checkbooks and into the business of attempting to do serious computing work on their relatively small machines. We felt that getting into the controller business was a natural extension of what we had been doing. The controller business has allowed us to put larger capacity drives into situations where people had needs for more computing power.

80-U.S.: In choosing an operating system, what types of problems did you have in developing software?

Lobo: If you will recall, at the time we were working on a controller board, the company was less than a year old. We started looking for somebody to do the operating system and frankly our experience was somewhat limited. That was a shortcoming on my part. Initially we did not investigate the total system needs as well as we should have. Our immediate problem was in finding people who were interested in getting involved with the Radio Shack operating system. Our conclusion was to go to the people who originally had written the Radio Shack operating system. We felt that putting this piece of hardware onto the Radio Shack Model I was going to be a difficult project, even for someone who knew and developed the existing Radio Shack operating system. Keep in mind, suddenly here was a piece of hardware that ran 5¼" floppies, 8" floppies, single density, double density, and had a dual serial port option. So, to say that this problem would be complex was perhaps an understatement.

80-U.S.: You originally told people that this product would be delivered in late January in production quantities (January of 1980). Obviously, you misjudged and failed to meet that. What problems did you encounter? What caused those delays?

Lobo: We first initiated work in late October, 1979, at that point in time assuming everything worked we anticipated having an operating system

going no later than mid December. The original intention was to take VTOS 3.0 and get that up and running on the interface. As the project developed there were a number of delays and frankly a number of those problems were the result of our lack of experience in this area. Still, by mid November 1979 we had good reason to believe we would have the operating system by the first of the year. At this point everything seemed to fall apart; one of the key individuals involved encountered major health problems in his family, and as time passed and the project developed it became evident that 3.0 was just not going to do the job. After evaluating a number of alternatives the decision was made to develop a totally new system. This was not an easy decision for us. We knew we had made commitments. We also knew it was possible to quickly produce a product that would not do the whole job. To make a long agonizing story short, we bit the bullet and went for the quality product. About the time the new operating system was fully underway, disaster struck again. We lost the key individual to a better job. Fortunately for us, the individual had a personal commitment to help us and as a result continued to work part time for the next three or four months. During this period we made many coordinating trips. Frankly, this was a very frustrating experience for everyone concerned; it's not that we weren't trying, it was just a question of not having enough time available. We had hardware ready to ship in January 1980, and we are only now beginning to see what we really want for an operating system for this piece of hardware.

80-U.S.: Having had this experience, what are you doing to see that it is not repeated?

Lobo: Well, the first thing is not to become tied to a single supplier. We began several months ago to find other people who know not only the Radio Shack operating system (TRSDOS), but also know NEWDOS. We have made arrangements with two software houses to bring software in-house although they remain independent firms. The two firms committed to working with us on a full time basis are Evans & Neal Associates of Albuquerque, New Mexico, and Galactic Software of Wisconsin. Galactic in particular has been working on the LX80 operating system and has already made a number of patches. We have had meetings in their facility to introduce them to the interface and demonstrate its full capability. In addition, we are working hard to improve the supporting documentation. Our ultimate objective is to fully utilize the hardware we have developed and doing that in conjunction with the software. We anticipated introducing software packages for the Model I, II, III, and that these packages will be available for use with the hardware. We are now committed to full time software support with established firms in the software business. ●

Toll-free order no.:
1-800-527-4196

LOOK!!

TRS-80*Owners Save on Equipment & Software!

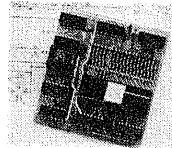


Percom Mini-Disk Systems

These Percom mini-disk systems store more data, are more reliable. Access times are fastest possible with your Expansion Interface. Heavy duty power supplies run cooler, last longer. Low noise three-wire ac power cord is safer. Enclosures are finished in compatible silver enamel. Prices:

TFD-100™ (40-track, 102 Kbytes/side)	
One-Drive Add-On	\$ 349.95
Two-Drive Add-On	699.45
Three-Drive Add-On	1049.95
TFD-200™ (77-track, 197 Kbytes)	
One-Drive Add-On	634.95
Two-Drive Add-On	1268.95
Three-Drive Add-On	1903.95

Price includes Percom upgrade PATCHPAK™ program.



DATA SEPARATOR™

This PC board plug-in adapter for the TRS-80* virtually eliminates data read errors (CRC error — Track locked out!) which occur on high density inner disk tracks, a problem that has plagued TRS-80* systems.

The Percom Data Separator™ is installed in the Expansion Interface without modifying the host system. Caution: Opening the TRS-80* Expansion Interface may void the limited 90-day warranty: \$29.95.

Percom OS-80™

An advanced easy-to-use disk operating system that works with Level II BASIC commands. Resides in only 7-Kbyte of memory. May be extended indefinitely with disk-resident utilities. Supplied on 5" disk with example programs: \$29.95 with instructions.

CIRCLE J Software

Two extremely useful utilities for Percom's OS-80™ DOS:

1. Machine Language Save/Load Utility. On 5" disk with bonus patch program that allows RS Renum-ber Utility to run under OS-80™. \$14.95, with instructions.
2. VARKEEP — Adds NAME SAVE and NAME KEEP commands to OS-80™. Use one set of common data with two or more BASIC programs. Also runs under Radio Shack DOS. On 5" disk, with instructions: \$14.95.

Z80ZAP

Super fast machine language disk modification utility. Read, Write, Display, and Modify sectors; remove passwords; apply patches, fixes; make backups and much more. On 5" disk with instructions: \$29.95.

100% machine language word processor...

SPECIAL DELIVERY (by Software Etc.)

Use MAILFORM to create name and address lists, EXTRACT to find names by ZIP, address, gender, age, etc.; SORT to sort an entire list on any field in seconds. Print personalized letters written with either the Electric Pencil or Scripsit* using MAILRITE. Prints labels from Mailfile created under MAILFORM. Runs under Percom's OS-80™, Radio Shack's TRSDOS* \$125 (disk).

How to Order

Order by calling Access Unlimited toll-free on 1-800-527-4196†.

Mail orders also accepted. Orders may be charged to a VISA or Master Charge account, or paid by a cashier's check, certified check or money order. We accept COD orders with 25% deposit. Sorry, we cannot accept personal checks. We pay shipping and insurance charges on orders over \$1,000.00. Add approximate insurance and shipping charges for under \$1,000.00. If in doubt about these charges, ask when you call in your order. Texas residents include 5% sales tax. Minimum order: \$20.00. Allow 2 to 4 weeks for delivery.

†Texas residents call (214) 494-0206

Prices and specifications subject to change without notice

™ trademark of Percom Data Company, Inc.

* RADIO SHACK and TRS-80 are trademarks of Tandy Corporation

Inexpensive Color Graphics: the Percom Electric Crayon™

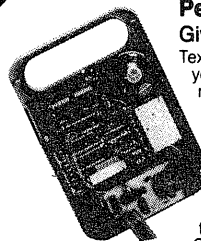
Spectacular multicolor graphics, sharp 2-color alphanumerics with your TRS-80*, a color tv and the Percom Electric Crayon™. Up to eight colors. Resolution with full display memory (6 Kbytes) is 256 X 192 picture elements. Microprocessor controlled, the Electric Crayon™ is not only a full color graphics system but also a complete, self-contained control computer with a dual bidirectional parallel I/O port — provision for second dual port. Interface the TRS-80* via your Expansion Interface or Printer Cable Adapter. Supplied with 1 Kbyte display RAM, EGOS™ operating system and comprehensive users manual with example programs \$249.95. Optional TRS-80* interconnecting cable: \$24.95.



Percom's Speak-2-Me-2™

Give your TRS-80* the gift of speech

Texas Instruments' Speak & Spell™ is the voice of your TRS-80* computer with this clever interface module manufactured by Percom. Your own Level II BASIC programs announce, command, implore with sentences and expressions formed from Speak & Spell's™ vocabulary. The Speak-2-Me-2™ PC module installs in the battery compartment of your Speak & Spell™. Power is supplied from an ordinary calculator power pak. Comes with interconnecting cable (for TRS-80* EI or Printer Cable Adapter), operating software and users manual: \$69.95 (Speak & Spell™ not included.)



Games People Play

Far out!

Scott Adams' Adventures (disks):

Adventure Sample	\$ 6.95
Adventure 1-9	14.95 ea
Adventure 1 & 2	24.95
Adventure 3, 4 & 5	39.95
Adventure 6 & 7	24.95
Adventure 6, 7 & 8	39.95

The Galactic Trilogy (disks):

Galactic Empire	\$19.95
Galactic Trader	19.95
Galactic Revolution	39.95

Space is this hunter's domain (disks):

Starfleet Orion	\$24.95
Invasion Orion	24.95

Space Battles — from Adventure (disk):

Super fast, real-time war game with exploding missiles, full space ship control. \$19.95	
--	--

Space Invaders — by Level IV (disk):

TRS-80* machine language version of Apple 'Super Invaders'. Sound effects: \$19.95	
--	--

Far in!

Datestones of Ryn (disk)	\$24.95
Temple of Apsai (disk)	29.95
Morloc's Tower (disk)	24.95

Disk System Interconnecting Cables

Improvement over RS cable design places drive 0, which includes the cable termination, at the end of the cable to eliminate the reflected noise of an unterminated cable. Better data integrity. Prices:

Two-Drive Cable	\$24.95
Four-Drive Cable	34.95

Power Line Filter

115/250 V, 50-400 Hz. Instructions included for easy installation in standard mini-box chassis: \$19.95

Minidiskettes

10 Disks in a convenient plastic organizer box	\$34.90
Single Disk	3.49

Disk Drive ID Tabs

1" x 1-1/4" self-adhering plastic drive identification tabs. Compatible silver with engraved black drive number. Two tabs (Nos. 0, 1): \$2.50; three tabs (Nos. 0, 1, 2): \$3.25; four tabs (0, 1, 2, 3): \$4.50.

ACCESS UNLIMITED

315 N. Shiloh • Ste. D1 • Garland, TX 75042

(214) 494-0206

™ trademark of Texas Instruments Corporation

‡ trademark of Michael Shrayser Software, Inc.

ITEMS

At Random

The New Look

We start the year with a new logo, a new cover, new layout, new size, printer and new additions to the staff. Our cover, which was constant since July 1979 started to fade with the last issue - from black to beige on the border. This issue the border is gone. The reason for changing the logo was that many were calling and asking "is this 80-us?" (as in "us" as a group). Our cover designer took one look and said he could fix that. There should be no doubt about it now.

We have also moved from a newspaper press to a commercial web press. The subscription copies will be mailed directly from the press, which should shave a couple of days in getting the copies to you. We are excited about the whole arrangement, and hope you concur.

Corrections to Nov/Dec 80 Issue

In our lead story on page 17 we implied that the Model III has a built-in cassette. Not so, it should have said that the cassette *port* is built-in.

In the "Let There Be Light!" article, on page 53, figure 1 indicates a connection between the 9 volt supply and the three resistors. This is no connection, the 9 volts are applied only to pin 7 of the 741 Op Amp.

Table 2 on page 89 listed a whole string of 74L5 integrated circuits. These should all be 74LS series IC's.

Technical Editor Lost

Jim Crocker, who was with us for slightly over a year, left us recently for greener pastures. We wish him well in his new endeavors. View from the Top of the Stack, which Jim authored, will continue along the same lines.

New Foreign Subscription Rate

Effective the first of January 1981, the subscription rate for all foreign subscribers is increased to \$30.00 per year. The increased size and weight of the Journal account for this. Also, foreign subscriptions are now for one year only.

Model I Discontinued!

Word has it that the Model I will be discontinued by the time you read this. Just what that means is uncertain. Will Radio Shack continue to sell software for the Model I? What about those 200 plus thousand owners? What do they do? 80-U.S. will continue to support the Model I with programs and hardware articles. You simply can't expect that many people to dump a working system and move up to something else.

Model III Came and Went

Just before press time we received our 32K Model III with two drives. Further on in this issue we went on

about how well it worked, etc. Sad to say, but the next day it developed a case of disk I/O problems, and is now resting peacefully at the repair center. In spite of that, it's still one neat machine, and you will be hearing more about it as time goes on.

In this Issue

Our feature articles are primarily on the measurement of time. We are happy to announce that the Calendar program works on all three TRS-80 Models (so do a couple of others in this issue).

A relative newcomer to the staff, Spencer Hall in this issue does some very interesting things with Basic Level II subroutines. See for yourself, on page 26.

Files & Foibles is a tutorial on file structures. It starts out this issue with a look at sequential files and file buffers. This is a preliminary step, prior to getting into Random files, which will appear next issue. See page 110.

Pete Carr takes us on a tour through the insides of NEWDOS80 and VTOS 4.0. As promised in the last issue, this is an in-depth look at these two new operating systems.

Every now and then your old editor takes pen in hand and writes. Such is the case on page 92. The Making of a Computer Program is a real life account of just that. Getting a big system program running and debugged is difficult and time consuming. (But darn, it was sure worth the effort!)

How about estimating the man-days required to write a computer program? Terry does a neat job on that with a program to estimate programming on page 78. Try it on something you have written recently - you may be surprised.

Our 21 feature articles this issue make it the biggest ever. It's been a challenge getting it all together, and we hope you enjoy this type of mid-winter madness. As always, tell them you saw it in the JOURNAL, and remember that nice days are made, not had.

Mike

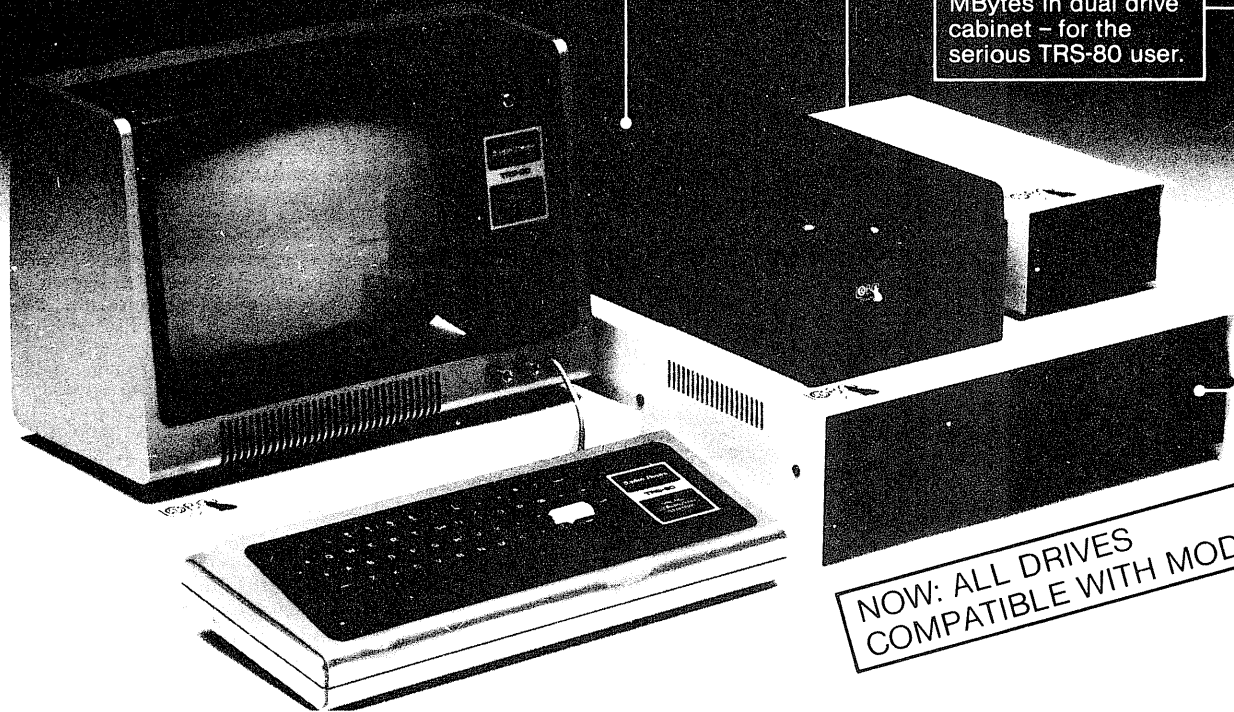
THERE IS A DIFFERENCE IN TRS-80 DISK DRIVES CAPACITY

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...



354 S. Fairview Ave.
Goleta, CA 93117
805/683-1576

"CAN YOU REALLY AFFORD TO PAY LESS?"

Quantity discounts available –
Dealer inquiries invited

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

- | | |
|---|---|
| <input type="checkbox"/> 5 1/4-in. Floppy drive | <input type="checkbox"/> 8-in. Winchester hard disk, 10 Mbyte drive |
| <input type="checkbox"/> 8-in. Floppy drive
Single sided
Double sided | <input type="checkbox"/> 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

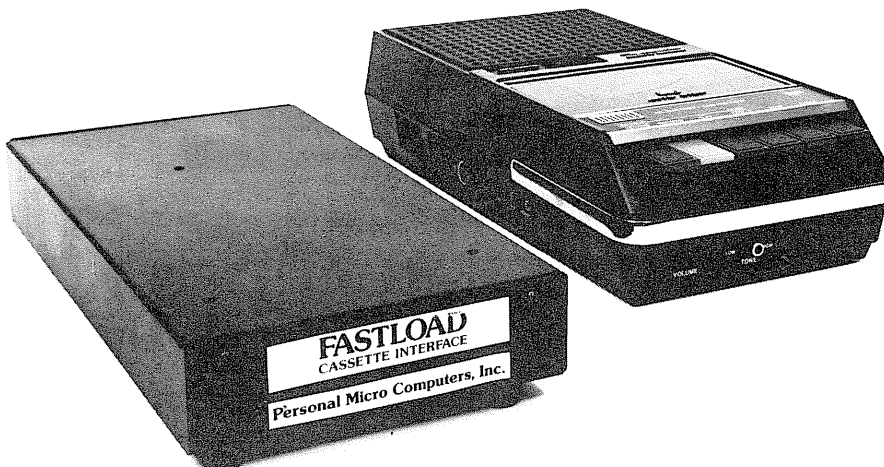
New Products

Horse Handicapping Program

A horse race handicapping system for the TRS-80 and Apple home computers has been announced by the 3G Company. The system also provides a listing which makes it adaptable for use with other computers. The 3G Company gathered and stored data on a high speed, large scale computer from a vast number of races. The computer then analyzed which attributes contribute to a horse's performance in a race, either positively or negatively. Out of twenty-three factors, the computer narrowed it down to ten, and these ten factors formed the basis for the system. The system consists of a guide on how to use the "Daily Racing Form" to obtain the ten factors needed for each horse, a sample form to simplify the data gathering, a cassette which contains a program that computes the odds for the current race, a program listing for use with other computers, and tips on how to use these odds when wagering. The complete cost for the system is \$19.95. This price includes postage and handling and is available mail order from 3G Company, Rt 3, Box 28A, Gaston, OR 97119.

Retaining Wall Program

Another tool for structural engineers has been introduced by DISCO-TECH of Santa Rosa, CA. RETWALL-1, a Retaining Wall design program for the TRS-80 is interactive, which means the designer synthesizes his own knowledge and the computing power of the program to develop the best possible design of either block walls or concrete walls with straight (parallel) or tapered sides. The designer is not locked into a few predetermined solutions. The program computes masonry stresses for concrete block walls and displays the maximum height for a given loading condition when masonry stress limits are exceeded. If the design exceeds maximum stress limits, RETWALL-1 lets the designer exercise the poured concrete option or go to full stresses instead of half stresses and call for special inspection. Trial design results are immediately displayed on the screen and may be printed out if the designer wishes to retain them. The program is priced at \$125, including manual. DISCO-TECH, Morton Technologies Inc., PO Box 11129, Santa Rosa, CA 95406 (707) 523-1600.



Solar Energy System Simulation

A comprehensive software package which teaches about and aids in the design of solar energy systems is now available from Peripherals Plus. The system consists of three programs: Thermal Analysis, Economic Analysis and Builder Program to create climatic data tapes. A major advantage of the Thermal Analysis program is that it allows multiple design options to be examined quickly and easily. The Economic Analysis determines the cost effectiveness of the solar system on a life cycle basis. The Builder program lets the user create a data tape for a specific geographic location. An extensive 80-page manual discusses solar energy simulations in depth and shows many sample runs of all the programs with a complete discussion of the output. Available for the TRS-80 on cassette (CS-3307, \$49.95, requires 16K) or diskette (CS-3802, \$99.95, requires 32K). Plus \$2.00 shipping. Peripherals Plus, 119 Maple Ave., Morristown, NJ 07960 (201) 267-4558.

Catalog Available

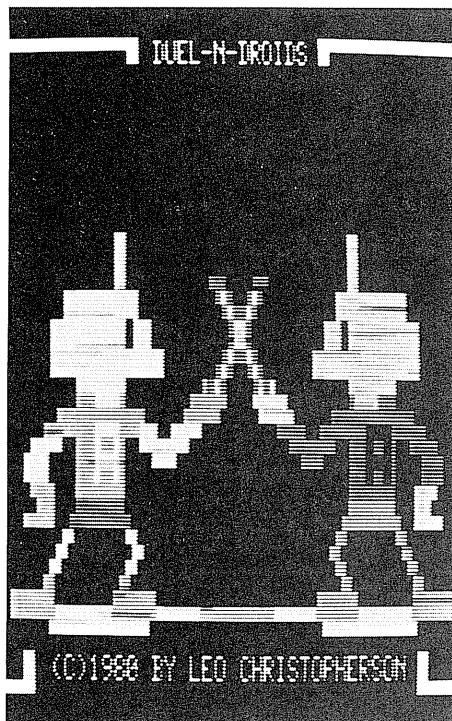
A catalog and specification sheets describing microcomputer peripheral equipment and interfaces are available from Pacific Office Systems, Inc., 918 Industrial Ave., Palo Alto CA (415) 493-7455. Featured are the POS-100 NRZI 9-track Tape Drive Controller/Formatter for S-100 systems; the POS 103/202 Dual Speed Auto Answer Modem; POS 731/735 I/O Selectric Printer Interface; POS TRS-80 Daisy Wheel Printer Interface; POS ASCII 725 IBM Selectric Printer; an office Selectric printer-conversion kit; plus an assortment of refurbished Selectric and Daisy Wheel terminals and other peripherals.

Disk Speed Cassette Loads

Personal Micro Computers, Inc. has announced an innovative device to input prerecorded programs into the TRS-80 Model I, Level II at 16 times normal speed. Standard cassettes can now be loaded at 8000 baud (that's one kilobyte per second) using a modified CTR-41 recorder and the FASTLOAD Cassette Interface. Any cassette program previously saved at normal speed (500 baud) can now be loaded at high speed. For short programs FASTLOAD is faster than disk because of disk start up time and longer programs load in seconds instead of minutes. To use the FASTLOAD, the user has only to initialize with a system command after turning on the TRS-80. Thereafter, the command LOAD, normally reserved for disk programs, can be used under Level II basic and will permit loading of programs at 1 kilobyte/second. The FASTLOAD can also be used under disk basic by calling it with a system command. Price for FASTLOAD Cassette Interface is \$188.00. The modified CTR-41 recorder is \$95.00. Units are available from dealers or by mail order from Personal Micro Computers, Inc., 475 Ellis St., Mountain View, CA 94043.

Duel-N-Droids

Acorn Software Products, Inc. announces the debut of DUEL-N-DROIDS, a new sound and graphics game by Leo Christopherson, for the Model I Level II TRS-80. The program features two androids who square off against each other with swords in both "practice" and "tournament" duels. The player controls his android with four keys, causing it to maintain defense, back off, or attack. Each win moves the rank of the player's android one level up the Duel-N-Droids scale. In the practice duels, the player manually controls his android while the computer controls the other. In the two types of tournament duels, the machine controls both androids, matching the player's against either equally or randomly ranked androids of its own. DUEL-N-DROIDS is priced at \$14.95 on cassette or \$20.95 on diskette. Dealer inquiries should be directed to Acorn Software Products, Inc., 634 North Carolina Ave., SE, Washington, DC 20003 (202)544-4259



Olympic Decathlon

Microsoft Consumer Products announced Olympic Decathlon, an exciting skill game for personal computers based on the Decathlon athletic competition. The program encompasses ten events including the 100 meter dash, long jump, pole vault, discus throw, shot put, 400 meter dash, 110 meter hurdles, 1500 meter run, high jump and javelin throw. Available on either cassette or diskette for the TRS-80 Model I. Retail price is \$24.95. Microsoft Consumer Products, 400 108th Ave., NE, Suite 200, Bellevue, WA 98004 (206) 454-1315

80 EPROM Programmer

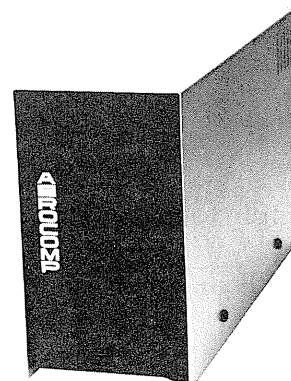
A 14 page booklet is available which provides complete product description of the 2708, 2516, 2716 and 2732 EPROM programmer for the TRS-80. The booklet is also a complete do-it-yourself instruction manual for the home builder. Complete with schematics, parts list (including supplier's addresses), and software listings. The software is human engineered and the menu driven display requires no programming experience. EPROMs may be duplicated or new data entered from cassette system tapes. The 80 EPROMMER will function with or without the expansion interface. Easy construction and low parts cost (under \$50 including power supply and ribbon cable) make the \$9.95 postpaid booklet a valuable investment. Software on cassette tape and parts kits are also offered. If the booklet is bought for product description, its purchase price can be applied to the \$129.00 purchase price of the completed and tested unit. The booklet "80 EPROMMER" is available from Graves Manufacture and Service, PO Box 306, Lake Bluff, IL 60044

Compiler for Level II

ACCEL2 is a new compiler for Level II TRS-80 Basic + Disk extensions, developed in Britain by Southern Software and now being marketed in the United States and Canada by Allen Gelder Software. It is a true doubly-optimising compiler producing extremely compact machine code translation of selected Disk Basic statements and functions in Integer, Single and Double Precision and String variable types. ACCEL2 has six diagnostic messages and a set of local/global compilation options to increase compatibility with subject programs and to control output code growth. It's compile-time routines are self relocating and occupy 5120 bytes while the run-time component takes only 1024 bytes, making the powerful compilation process available to even 16K non-disk machines. Output can be saved to disk or tape. The run-time routines can be included with the derived code without royalty fees being required for further sale. ACCEL2 is supplied on cassette with a booklet of instructions and examples, and sells for \$88.95 from Allen Gelder Software, Box 11721 Main Post Office, San Francisco, CA 94101

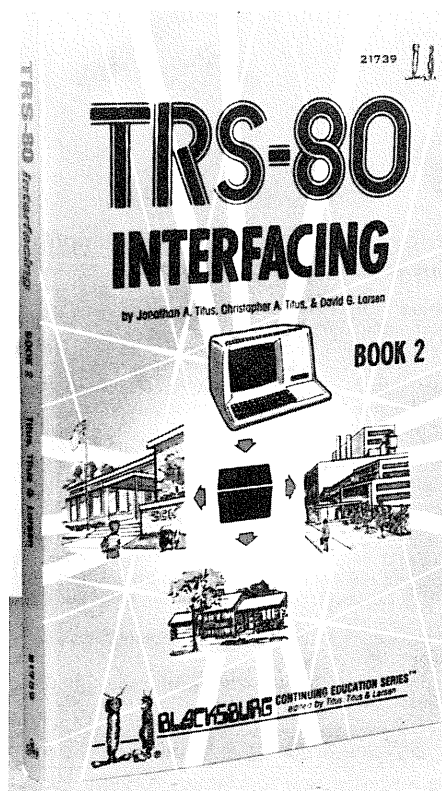
Disk Drives

AEROCOMP, Inc of Dallas, TX has introduced a new line of disk drives incorporating the best features in mini-floppy drives on the market. The MPI bare drive was selected as the heart of the system. AEROCOMP disk drives incorporate a highly stable, solid state, dual power supply for long service life. The single-headed models (40-1 & 80-1) are "flippy", a very important feature which allows the use of both sides of a diskette by flipping it over - cuts media cost by 50%. All drives are fully assembled and 100% tested at the factory, and are complete with attractive silver sheet-metal enclosures. For complete information and pricing contact AEROCOMP, Inc., PO Box 24829, Dallas, TX 75224 (214) 337-4346



TRS-80 Interfacing

The Blacksburg Group has introduced Book 2 in the series TRS-80 Interfacing. Written by J A Titus, C A Titus and D G Larsen, this 254 page book (No 21739) is available for \$9.95 plus \$1.00 shipping from Group Technology Ltd., PO Box 87, Check, VA 24072 (703) 651-3153



Computer Forms

New England Business Service Inc has announced the establishment of NEBS Computer Forms as a division. Their aim is satisfying the needs of the small computer user. The company produces pre-printed forms for computer use. New phone numbers are (617) 448-6167 for the general offices and 1-800-225-9550 toll free for orders and customer inquiries. (MA residents call 1-800-922-8560)

Double Density Software

Micro Systems Software, Inc. now has double density software available for the TRS-80 Model I that is equipped with the Percom Doublert+. First is a disk editor called "Disk Zap 2.3". This editor will work either single or double density diskettes. It is track and sector oriented, and offers total access to all parts of the disk. It has the ability to format and backup diskettes as well as editing them. Second is a new double density DOS. DOSPLUS 3.1D is similar in most respects to single density operating systems, but offers the increased disk storage of double density. Disk Zap is \$19.95, and DOSPLUS 3.1D is \$99.95. Micro Systems Software Inc., 5846 Funston St., Hollywood, FL 33023 (305) 983-3390

Federal Funding Handbook

Radio Shack has announced the latest addition to their Educational Resource Series, a Federal Funding Guide and Proposal Development Handbook for Educators. The handbook, written by Dr Frank Jackson, is a resource guide for educators that explains how to locate external funding and how to write proposals. Some special features of the handbook include identification of the major sources of Federal funding and listing of publications for monitoring Federal funding sources. It also outlines the essential elements of a proposal, offers suggestions for continuing and increasing external funding and contains an appendix of educational sources of funding information. The appendix has a listing with addresses of all state education agencies to contact for assistance in obtaining funding information. The handbook is available from participating Radio Shack stores and dealers and Radio Shack Computer Centers, nationwide. Price is \$2.50

Computer Home Banking

Electronic home banking became a reality in Knoxville, TN. For the first time, consumers are able to use the services of their local bank with a computer at home. The program is expected to be available nationwide in 1981.

The "Express Information" bank-at-home is a joint venture of United American Service Corporation (UASC; Radio Shack, and CompuServe, a subsidiary of H & R Block. The United American Bank in Knoxville was selected as the first bank to use and market the service to its customers.

For an estimated price of \$15 to \$25 a month, 400 of the bank's customers will gain services of Radio Shack's new TRS-80 Color Computer, including a standard keyboard which plugs into the customer's own television set and telephone. Customers will have access to a comprehensive news and financial advisory service, be able to pay most of their bills, receive current information on their checking accounts, use a sophisticated bookkeeping service, and apply for loans. This opens a new dimension in convenience banking.

A commitment to in-home banking was expressed by Jake F Butcher, Chairman, President and Chief Executive Officer of United American Bank:

"We are delighted to be chosen as the first bank in the country to offer these computer services. Our desire is to offer our customers the most sophisticated service possible. The technology is there, our reports indicate that customer demand is there and we are ready to be the first bank to offer what they have been asking for -- convenience banking without leaving home."

MODEL III DISK DRIVES AVAILABLE NOW!!

- Super Disk Controller. Allows you to read **any** soft sector diskette. **Single or Double Density!**
- Reads 5¼" and 8" Diskettes!
- No additional hardware required to read 8" diskettes.
- Enhanced storage capability of up to 800 kilobytes per drive with purchase of the 80 or 160 track drives.
- Convert existing Model I software to Model III format*
- Complete package includes Drive, Power Supply, Cables, Controller Board, Brackets and DOS.

40 Track Drive - \$775.00

80 Track Drive - \$895.00

160 Track Drive - \$995.00

* Some programs may require patching to operate.

Dealer Inquiries Invited

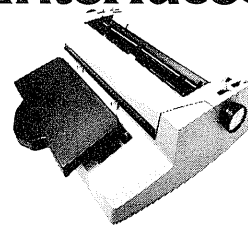
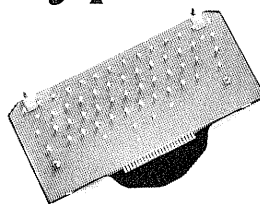
Model III TRS-80 is a product of the Tandy Corp.

**Watch for our 10 megabyte Model III-H
Winchester Drive System!**

Call or write for further information

**Micro Mainframes
714 Alhambra Blvd.
Sacramento, CA 95814
(916) 447-7048**

At last...the Typewriter Interface!



Turn your electric typewriter into a low cost, high quality hard copy printer. 1 Year Warranty

Dynatyper—the patented* RDI—I/O Pak is fast becoming the industry standard for typewriter output. Why? Because:

1. It takes 2 minutes to initially install and 5 seconds to remove or replace.
2. You *do not* have to modify your typewriter. All factory warranties and maintenance agreements on your typewriter will be honored.
3. You can use it with *all* powered carriage return typewriters that have U.S. keyboard. Our Model I works with all *non* Selectrics and our Model II works with Selectrics. Conversion between models takes 2 minutes and the kit (26 plungers) is available for a nominal charge.
4. You don't have to lug around a bulky printer when you travel. If there is a typewriter at your destination, you can install the light (3 lbs.) I/O Pak in just 2 minutes.
5. Same interface for TRS-80, Apple and GPIB. Centronics and Pet compatible interfaces are available in third quarter 1980. Electric pencil available.
6. Delivery: Stock to two weeks. Price: \$499. for the complete system, FOB Rochester, Domestic.

Over 1000 in operation today. VISA and MasterCard accepted. Call Ken Yanicky at 716-385-4336.

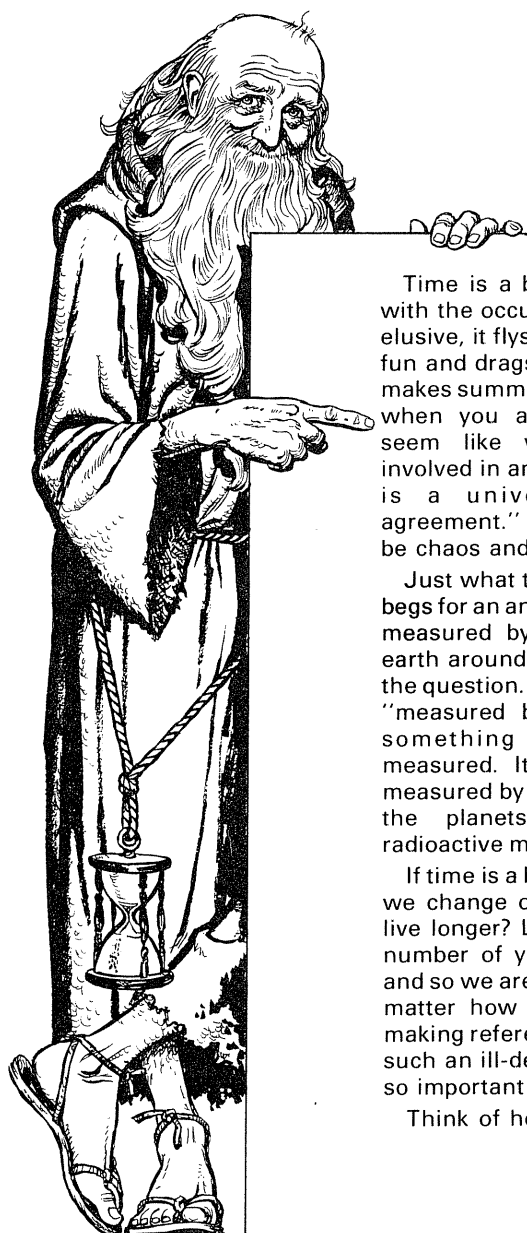
*Patent Pending

ROCHESTER DATA

3100 Monroe Avenue, Rochester, New York 14618

incorporated

TIME



Time is a basic concept that deals with the occurrence of events. Time is elusive, it flies by when you are having fun and drags when you are bored. It makes summer seem like a whole year when you are young, and months seem like weeks when you are involved in anything interesting. Time is a universal "gentleman's agreement." Without it there would be chaos and disorder.

Just what time is, is a question that begs for an answer. Saying that time is measured by the movement of the earth around the sun doesn't answer the question. The very fact that we say "measured by" indicates that it is something *e/se* that is being measured. It is time that is being measured by clocks, the movement of the planets and the decay of radioactive material, but what is time?

If time is a basic concept, then could we change our concept and actually live longer? Longer than what, some number of years? Years imply time, and so we are caught up in a circle, no matter how you think of it you are making reference to time. It is odd that such an ill-defined concept should be so important in our lives.

Think of how many computers are

bought because they are faster than another. Racing exists because one horse or car may be faster than another. New airplanes are built and sold on the premise that they can get you from here to there faster than a previous model.

We take time for granted. Once each year we get together to raise hell and drink champagne and celebrate the passing of yet another year. We still don't know what time is.

80-U.S. is not terribly hot on 'theme issues,' the kind that Wayne, David and the others do so well. But in this issue we have a theme of sorts. The theme is not on time, but on the measurement and conversion of time. On the following pages you will find a calendar program (with Anatomy by Dr. Bahn), a day of the week routine, a days between dates routine and, as part of another article (A Functional Subroutine Library) you will find a routine for conversion to Julian dates.

Although none of these routines will tell you what time really is, they may help you measure the concept of time in the only reality we know.

May the year 1981 be as rewarding to you as the last year was to us!

The Editors

A Calendar Maker

We work with calendars every day, but amazingly, few of us have really looked at how the calendar works. They have been around since the dawn of recorded history, and so we have come to accept them without question.

Quick, was 1900 a leap year? Will 2000 be a leap year? The answers are no and yes (but you knew that anyway, didn't you!). The calendar we commonly use is called the Gregorian Calendar, named after Pope Gregory XIII who decreed that the day after 4 October 1582 would be 15 October to account for 10 days of error in the Julian calendar.

The Julian calendar had been authorized by Julius Caesar in 46 BC and worked much the same as the Gregorian. It had 365 day years with a 366th day thrown in every 4 years (called an intercalary day). This was to account for the fact that the true year was $365\frac{1}{4}$ days long.

Sounds good so far, but in 730 AD an Anglo-Saxon monk known as Bede, announced that the Julian year was 11 minutes and 14 seconds too long. This meant that the 365½ day year was longer than a true solar year by 0.01 days. So, the Julian calendar used too many leap years, thereby gradually losing with respect to the true year.

To correct for that, Pope Gregory made three out of every centesimal (divisible by 100) years into regular years while the fourth was to be a leap year. So 1600 was a leap year, but 1700, 1800 and 1900 were not. The next centesimal year, 2000, will by this reckoning be a leap year.

The calendar program here is based on the Gregorian calendar and includes corrections for leap years as well as every 100 and 400 years. The information on calendars was taken from the World Almanac. The formula for the day of the week was adapted from the book *Some Common Basic Programs* by Poole and Borchers. The calendars were tested using a random selection of years and comparing a few days with a perpetual calendar from the 1979 World Almanac. No errors were noted in the testing.

Using the subroutines in the program, it is possible to build other programs to give Julian calendars, Julian dates, days between dates, etc. Even if you just want a calendar display for your room or office, you can have fun with this one.

```

110 REM ***** ANDY'S CALENDAR MAKER *****
120 REM BY TERRY R. DETTMANN *****
130 REM *****
140 REM *****
150 CLEAR2*MEM/3:DEFINTA-Z:DIMA$(15)
160 HDR$=STRING$(E2,E1)+STRING$(20,32)+"ANDY'S CALENDARS"
170 D$="
180   +CHR$(13)+STRING$(E2,E1)+CHR$(13)
190   1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8
200   + "202122232425262728293031
210   1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8
220   FM$=" %":
230   HW$=" %":
240   SUN MON TUE WED THU FRI SAT"
250 REM 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8
260 HS$=" %"
270 REM --- MASTER INPUT ---
280 CLS:PRINTHDR$
290 INPUT"FOR WHAT YEAR DO YOU WANT THE CALENDAR";Y
300 REM --- PRINT THE CALENDAR ---
310 GOSUB520:REM --- PRINT ANDY
320 FORMB=11012STEP2:
330 MC=MB:YC=Y:GOSUB360:W1=W:
340 MC=MB+1:YC=Y:GOSUB360:W2=W:
350 MC=MB:GOSUB490:M1$=MY$:
360 MC=MB+1:GOSUB490:M2$=MY$:
370 LPRINT" ":LPRINTUSINGH$:M1$:Y:M2$:Y:
380 PRINT@896,USINGH$:M1$:Y:
390 PRINT@950,USINGH$:M2$:Y:
400 LPRINTH$;HWS
410
420
430
440
450
460
470
480
490
500
510
520
530
540
550
560
570
580
590
600
610
620
630
640
650
660
670
680
690
700
710
720
730
740
750
760
770
780
790
800
810
820
830
840
850
860
870
880
890
900
910
920
930
940
950
960
970
980
990

```

```

230 MC=MB:GOSUB420:D1=DM:MC=MB+1:GOSUB420:D2=DM
240 FOR I=1 TO 6
250 LPRINT " ";
260 FOR J=1 TO 7:E1=((I-1)*7+J+6-W1)*2-1
270 IF ((I-1)*7+J)>D1+W1 THEN DY$=" " ELSE DY$=MID$(D$,E1,2)
280 LPRINT USING FM$;DY$;:NEXT J
290 LPRINT " ";
300 FOR J=1 TO 7:E2=((I-1)*7+J+6-W2)*2-1
310 IF ((I-1)*7+J)>D2+W2 THEN DY$=" " ELSE DY$=MID$(D$,E2,2)
320 LPRINT USING FM$;DY$;:NEXT J
330 LPRINT " ":NEXT I
340 NEXT MB
350 GOTO 100
360 REM ----- DAY OF THE WEEK -----
370 IF MC=2 THEN 380 ELSE MC=MC+12:YC=YC-1
380 W=1+2*MC+INT(.5*(MC+1))+YC+INT(YC/4)-INT(YC/100)+INT(YC/400)
390 W=W-INT(W/7)*7
400 W=W+6:W=W-INT(W/7)*7
410 RETURN
420 REM ----- NUMBER OF DAYS IN THE MONTH -----
430 IF MC=2 THEN 450 ELSE LPM=0
440 IF (Y-INT(Y/4)*4)=0 THEN IF ((Y-INT(Y/100)*100)=0) AND ((Y-INT(Y/400))>0) THEN LPM=0 ELSE LPM=1
450 M$="312831303130313130313031"
460 DM=VAL(MID$(M$,2*MC-1,2))
470 IF MC=2 THEN DM=DM+LPM
480 RETURN
490 REM ----- MONTH OF THE YEAR -----
495 REM 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8
500 MN$="JANUARY FEBRUARY MARCH APRIL MAY
      + "JUNE JULY AUGUST SEPTEMBER OCTOBER "
      + "NOVEMBER DECEMBER "
510 MY$=MID$(MN$, (MC-1)*9+1, 9):RETURN
520 REM ----- PRINT ANDY -----
530 FOR Z=1 TO 3:LPRINT CHR$(138):NEXT Z
540 GOSUB 620
550 FOR I=1 TO 14
560 FOR K=1 TO 5:LPRINT TAB(16);
570 FOR J=1 TO 12:J$=MID$(A$(I),J,1)
580 IF J$="0" THEN LPRINT B$;:GOTO 600
590 LPRINT A$;
600 NEXT J:LPRINT CHR$(13):NEXT K:NEXT I
610 RETURN
620 REM ----- DATA FOR PRINTING THE FIGURE -----
630 A$="80US":B$=" ":A$(1)="000010010000"
640 A$(2)=A$(1):A$(3)="000111110000"
650 A$(4)="001101100":A$(5)="001110011100"
660 A$(6)="000011110000":A$(7)="001111111100"
670 A$(8)="01101110110":A$(9)="110011110110"
680 A$(10)="01101110110":A$(11)="001110010011"
690 A$(12)="000010010000":A$(13)=A$(12)
700 A$(14)="011110011110":RETURN

```

This program requires a printer.

The calendar program works without modification on Model I, Level II and up, and on the Model III 16K Level II and up. Be especially careful to get the spacing correct in lines 70, 80, 90 and 500.

This program will also work on the Model II with slight modification. Model II users should change the percent signs in lines 80 and 90 to backward slashes (use Control 9 to get the backward slash). Also, Model II users should change the two PRINT@'s in line 210 to PRINT@ 1700 and 1780.

JANUARY							FEBRUARY							1981						
SUN	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT
4	5	6	7	8	9	10	1	2	3	4	5	6	7	1	2	3	4	5	6	7
11	12	13	14	15	16	17	8	9	10	11	12	13	14	15	16	17	18	19	20	21
18	19	20	21	22	23	24	15	16	17	18	19	20	21	22	23	24	25	26	27	28
25	26	27	28	29	30	31	22	23	24	25	26	27	28	29	30					

MARCH							APRIL							1981						
SUN	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT
1	2	3	4	5	6	7	5	6	7	8	9	10	11	1	2	3	4	5	6	7
8	9	10	11	12	13	14	12	13	14	15	16	17	18	8	9	10	11	12	13	14
15	16	17	18	19	20	21	19	20	21	22	23	24	25	15	16	17	18	19	20	21
22	23	24	25	26	27	28	26	27	28	29	30			22	23	24	25	26	27	28
29	30	31					23	24	25	26	27	28	29	29	30					

MAY							JUNE							1981						
SUN	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT
3	4	5	6	7	8	9	1	2	3	4	5	6	7	1	2	3	4	5	6	7
10	11	12	13	14	15	16	8	9	10	11	12	13	14	8	9	10	11	12	13	14
17	18	19	20	21	22	23	15	16	17	18	19	20	21	15	16	17	18	19	20	21
24	25	26	27	28	29	30	22	23	24	25	26	27	28	22	23	24	25	26	27	28
31							29	30						29	30					

JULY							AUGUST							1981						
SUN	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT
5	6	7	8	9	10	11	3	4	5	6	7	8	9	3	4	5	6	7	8	9
12	13	14	15	16	17	18	10	11	12	13	14	15	16	10	11	12	13	14	15	16
19	20	21	22	23	24	25	17	18	19	20	21	22	23	17	18	19	20	21	22	23
26	27	28	29	30	31		24	25	26	27	28	29	30	24	25	26	27	28	29	30

SEPTEMBER							OCTOBER							1981						
SUN	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT
6	7	8	9	10	11	12	4	5	6	7	8	9	10	4	5	6	7	8	9	10
13	14	15	16	17	18	19	11	12	13	14	15	16	17	11	12	13	14	15	16	17
20	21	22	23	24	25	26	18	19	20	21	22	23	24	18	19	20	21	22	23	24
27	28	29	30				25	26	27	28	29	30	31	25	26	27	28	29	30	31

NOVEMBER							DECEMBER							1981						
SUN	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT
1	2	3	4	5	6	7	6	7	8	9	10	11	12	6	7	8	9	10	11	12
8	9	10	11	12	13	14	13	14	15	16	17	18	19	13	14	15	16	17	18	19
15	16	17	18	19	20	21	20	21	22	23	24	25	26	20	21	22	23	24	25	26
22	23	24	25	26	27	28	27	28	29	30				27	28	29	30			
29	30																			

Anatomy of the Program

(A Calendar Maker)

R C Bahn

1. INTRODUCTION

This program is designed for printer output: the "Android" figure is first printed, this is followed by a calendar of a designated year.

The program is written in seven modules. Each module has essentially one entrance and one exit. There are no GOTO statements which direct the program flow to another module. Module 1 is the initialization module and extends from line 10 to line 90. Module 2 is the mainline program and extends from line 130 to line 350. Module 3 extends from line 360 to line 410. Module 4 is a subroutine to determine the number of days of the month and extends from line 420 to line 480. Module 5 is a subroutine which determines the month of the year and extends from line 490 to line 510. Module 6 is a subroutine to print the "Android" and extends from line 520 to line 610. Module 7 is a subroutine to generate the string variables necessary to print the "Android" and extends from line 620 to line 700.

The program uses several types of BASIC statements. If you are not already familiar with BASIC, review the use of the following statements or symbols: CLS, REM, CLEAR, DIM, DEFINT, IF, THEN, ELSE, GOSUB, RETURN, FOR, STEP, NEXT, CHR\$, MID\$, STRING\$, VAL, =, +, >, <, INT, LPRINT, PRINT@, PRINT USING, INPUT. The more complex computer concepts which are important in this program include the FOR-NEXT loops in lines 150-350 and lines 520-600; subroutines in lines 360-410, 420-480, 490-510, 520-610, and 620-700; concatenation of string variables illustrated in line 60 and 500; unpacking of string variables illustrated in lines 500 and 510; subscripting of string variables illustrated in lines 630-700; and the use of logical IF statements illustrated in lines 270, 310,

370, 430, 440, 470, 580.

Finally, in problems concerned with integers such as dates in a month, the ordered number of a month in the year, or the year itself, divisibility by certain numbers such as 4 in leap years, and 7 as in days of week provides crucial information. For example, commonly leap years are exactly divisible by 4. Exact divisibility by four (a leap year) could be determined in a computer program by making $Y = \text{INT}(\text{YEAR}/4)$ and then testing whether $\text{YEAR} = 4 * Y$. If there had been a remainder in the division, the equality test would have been false. If the remainder of the division were zero, the equality test would be true and YEAR would be in fact a leap year. This type of logic is employed in lines 380, 390, 400, 440.

Two related calendar problems remain for your solution. Given two dates: determine the elapsed time in years, months, weeks, days. Given a date, determine the day of the week. Both of these problems have significant practical applications in business.

II STRING VARIABLES

A\$	Graphics block "80US" defined in line 630 and used in line 590.
A\$(I)	Subscripted variable containing binary (0,1) data for graphics output.
B\$	Blank graphics block defined in line 630 and used in line 580.
D\$	Packed list of the possible dates in a month defined in line 70 and used in lines 270 and 310.
FM\$	Format of LPRINT USING statement. Defined line 80 and used in lines 280 and 320.
H\$	Format of LPRINT@, USING statement. Defined in line 90 and used in lines 200 and 210.
HDR\$	The output header defined in line 60 and used in line 110.

HW\$	List of days of the week. Defined in line 80 and used in line 220.		
J\$	Temporary storage of A\$(I). Defined in line 570 and used in line 580.	180-190	MB in line 170 for right hand month. This adjusts for STEP 2 in line 150.
M1\$	Temporary storage of left hand month. Defined in line 180 and used in lines 200 and 210.	200	Find month of year for left hand and right hand months of calendar output.
M2\$	Temporary storage of right hand month. Defined in line 180 and used in lines 200 and 210.	210 220 230	LPRINT left hand and right hand month and year.
MN\$	Packed list of months defined in line 500 and used in line 510.	240	Print on video screen months and year.
MY\$	Name of month. Generated in line 510 and returned to the mainline program. Transferred to M1\$ and M2\$ following GOSUB 490 in lines 180 and 190 respectively.	250 260 270	LPRINT the days of the week.

III NUMERICAL VARIABLES

A	Used in line 50 in DEFINT statement.		
D1	Day of month defined in line 230 and used in line 270	280	Find number of days in month for right and left hand calendar display.
D2	Day of month defined in line 230 and used in line 310.	290-320	Set up I loop for maximum of six lines of output per month.
DM	Day of month defined in line 460 and 470, returned to mainline program and used in line 230.	330:1	Print left margin offset.
E1	Indexing variable defined in line 260 and used in line 270.	330:2	Set up J loop for seven days in a week in left hand display.
E2	Indexing variable defined in line 300 and used in line 310.	340 350	Compute date and extract from D\$ with the MID\$ statement. Note leading blanks in D\$ defined in line 70.
I,J,K	Loop indices used in lines 240, 260, and 300 respectively.	360-410	Print date in left hand calendar display. Note use of semicolon to indicate continuous printing on line.
LP	Indexing variable used in lines 430, 440 and 470.	370 360-400	Repeat process outlined in lines 260-280 for right hand calendar display.
MB	Outermost loop index of mainline program used in line 150. End of loop occurs on line 340.	420	Terminate printing with blanks. Note absence of semicolon. Next printed characters will occur on a new line.
MC	Indexing variable used in line 160-190, 230, 370, 380, 430, 460, 470 and 510.	430 440 450	Terminate I loop.
W	Ordered number of day of week, defined in lines 380-400, returned to the mainline program and used in lines 160 and 170.	460 470 490-510 500	Terminate MB loop.
W1	Temporary storage of left hand day of week, defined in line 160 and used in lines 260 and 270.	510	Return to Master Input. To escape from this program press the BREAK key.
W2	Temporary storage of right hand day of week, defined in line 170 and used in lines 300 and 310.	520-610 530	Module #3, "day of the week" subroutine.
Y	Input variable for year, defined in line 120 and used in lines 160, 170, 200, 210, 440.	540 550	Adjust MC for months later than February.
YC	Temporary storage of year, defined in lines 160 and 170 and used in lines 370 and 380.	560	Compute W, day of week for Gregorian calendar.
Z	Used in line 50 in DEFINT statement.	570	Module #4, "number of days in the month" subroutine.

IV LINE BY LINE COMMENTARY

10-90	Module #1; initialization.		
10-40	Graphic program header.		
50:1	Clear memory and save 2/3 of memory for string variables.	570	Go to line 450 if the month is not February.
50:2	Define all variables (A-Z) as integers.		Set leap year Flag (LP) to zero or one.
50:3	Reserve space for A\$(I).		Define M\$, the packed number of days in successive months.
60	Concatenate (add) string variables to form the output header.	580-590	Find correct number of days within M\$.
70	Define the possible dates in the month.		If the month is February, adjust for leap year.
80-90	Define output formats (FM\$,H\$) and define days of week (HW\$).		Module #5, "month of the year" subroutine.
100-350	Module #2, mainline program.		Define MN\$, a concatenated string variable containing names of months of the year.
100-130	Clear screen, print screen header and prompt for keyboard input of year.	600	Extract from MN\$ the appropriate name of the month by use of MID\$ statement.
140	PRINT Android	620-700	Module #6, "Print Android" subroutine.
150-340	PRINT calendar.	630	Initialize printer, move paper ahead 3 lines.
150	Set up loop (MB) to start at 1, end at 12 with interval jumps of 2. This loop will actually end when MB=11.	640-700	CHR\$ (138) is a graphics character.
160-170	Find day of week for left hand and right hand months of calendar output. Note increment of		Define A\$, B\$, and A\$(I). Note A\$ and A\$(I) are different variables.

A Day of the Week/Days Between Dates Program

This program works "as is" on Models I, II, and III.

Here is a short program that will give you the total number of days between any two dates (within limits) or the day of the week for a given date.

The total days calculator subroutine was inspired by a routine in the Texas Instruments SR-56 calculator handbook. The day of the week subroutine was an original creation by the author, who wishes to remain unknown.

```

10 REM * * * * *
20 REM * * * DAYS BETWEEN DATES/DAY OF WEEK * * *
30 REM * * * * *
40 CLEAR100
50 CLS:REM * * * * * MENU/SELECTION * * * * *
60 PRINT"          DAYS BETWEEN DATES AND DAY OF WEEK"
70 PRINT"      ( VALID BETWEEN 1 JAN 1801 THROUGH 31 DEC 2099 )"
80 PRINTSTRING$(60,61):REM * DRAWS EQUAL SIGN LINE *
90 PRINT"ENTER 1 FOR DAYS BETWEEN DATES"
100 PRINT"ENTER 2 FOR DAY OF WEEK"
110 PRINT"ENTER 3 TO QUIT"
120 PRINTSTRING$(60,61)
130 PRINT:INPUT"YOUR CHOICE":X
140 IFX=(0ORX=)4THENGOTO130
150 ONXGOTO170,260,640
160 REM * * * * * DAYS BETWEEN DATES SECTION * * * * *
170 INPUT"ENTER FIRST DATE (MM,DD,YYYY)":M,D,Y
180 GOSUB530:GOSUB330:K=D1
190 INPUT"ENTER SECOND DATE (MM,DD,YYYY)":M,D,Y
200 GOSUB530:GOSUB330
210 T=D1-K:T=ABS(T):IFT<>1THEN220ELSE230
220 PRINT"THERE ARE (<T>) DAYS BETWEEN THESE TWO DATES.":GOTO
    240
230 PRINT"THERE IS (1) DAY BETWEEN THESE TWO DATES"
240 PRINT:PRINT"PRESS (ENTER) TO CONTINUE":INPUTX:GOTO50
250 REM * * * * * DAY OF WEEK SECTION * * * * *
260 M=01:D=01:Y=1801:GOSUB330:K=D1
270 INPUT"DAY OF WEEK FOR WHICH DATE? (MM,DD,YYYY)":M,D,Y
280 GOSUB530:M2=M:D2=D:Y2=Y:GOSUB330:T=D1-K
290 Q=T/7:W=INT(Q):Q=Q-W:Q=Q*100:Q=INT(Q):Q=Q/7:Q=INT(Q):GOSUB4
    40
300 PRINT"FOR THE DATE GIVEN, THE DAY OF THE WEEK IS - ":B$
310 INPUT"PRESS (ENTER) TO RETURN TO MENU":GOTO50
320 END
330 REM * * * * * TOTAL DAYS CALCULATOR SUBROUTINE * * * * *
340 D1=(Y*365)+D:M=L
350 IFM<=2THEN390
360 L=(L*.4)+2.3
370 L=INT(L)
380 D1=D1-L:Y=Y+1
390 M=((M*31)+(Y-1)/4)
400 M=INT(M):D1=D1+M
410 IFY=1900THEND1=D1+1
420 RETURN
430 END
440 REM * * * * * DAY OF WEEK SUBROUTINE * * * * *
450 IFQ=0THENB$="WEDNESDAY"
460 IFQ=2THENB$="THURSDAY"
470 IFQ=4THENB$="FRIDAY"
480 IFQ=6THENB$="SATURDAY"
490 IFQ=8THENB$="SUNDAY"
500 IFQ=10THENB$="MONDAY"
510 IFQ=12THENB$="TUESDAY"
520 RETURN
530 REM * * * * * UNREAL DATE TRAP * * * * *
540 IFM<=0 OR M=13THENCLS:PRINT"UNREAL MONTH!":M:D;Y:GOTO630
550 IFD<=0 OR D>31 THEN CLS:PRINT"BAD DAY!":M:D;Y:GOTO630
560 IFD=31AND(M=4ORM=6ORM=9ORM=11)THENCLS:PRINT"THERE ARE NOT
    THAT MANY DAYS IN THAT MONTH!":GOTO630
570 IFY<1801 OR Y=2100 THEN CLS:PRINT"YEAR OUT OF RANGE":M:D;Y:
    GOTO630
580 IFM=2ANDD=30THENCLS:PRINT"FEBRUARY DOES NOT
    HAVE THAT MANY DAYS!":GOTO630
590 IFM=2ANDD=29ANDY=1900THENCLS:PRINT"NO FEBRUARY 29TH IN 1900
    !":GOTO630
600 IFM=2 AND D=29THEN Z=Y/4
610 IFZ-INT(Z)<>0THEN CLS:PRINT"THERE IS NO FEB 29TH IN THAT YE
    AR!":GOTO630
620 RETURN
630 FORX=1TO2000:NEXT:GOTO50
640 CLS:PRINT"BYE FOR NOW":STOP

```


A Functional Subroutine Library

David R Pepple
Bexley, OH

If like many TRS-80 programmers you are beginning to notice you need the same piece of code in a variety of programs, building a functional subroutine library may be your salvation.

When I first received my Level II TRS-80 I hurriedly began constructing programs. After a short period of time I came to realize that frequently I was doing many of the same tasks over and over. I developed some standard subroutines to do many of these things, but to really get any use out of them I had to know at the beginning of a program that I would need them. This is due to the fact that CLOAD in Level II clears memory of any existing code when loading a new program. Then I ran across a short note on how to merge two programs using the CLOAD command in the July 1979 issue of Radio Shack's *Microcomputer Newsletter*¹. This simple six step process provides the basis for really beginning to get the use out of subroutines no matter when you realize the need for them.

If we combine this helpful hint with some coding conventions and procedural guidelines, we stand a good chance of substantially reducing the keying required to get new programs developed. This technique will also reduce the testing time necessary to become operational.

In establishing any library of subroutines it is imperative that coding and documentation standards be developed to avoid later frustration and disappointment. Some guidelines I use are:

1. All subroutine line numbers start at 30000 or higher. When developing a program I keep line numbers below 29000. This guarantees that I will always be able to merge any subroutines into my program since they always have line numbers greater than the receiving program.

2. All subroutines begin with a remark statement specifying the function.

3. All subroutines are reached by a GOSUB command and are designed to be closed. This means that they have only one entry and one exit point and that all variables are initialized and controlled within the routine.

4. A standard sheet of information about each subroutine is created when it is stored on the library tape. These sheets are used as a reference when deciding to incorporate routines into a program.

To give a more concrete example of how such a library of subroutines could be constructed I have developed the program which follows this article. This program was written based on an article in *Computerworld* by R A Cornish². The article described a series of assembly language routines for translating Gregorian dates to Julian dates and translating Julian dates to the day of the week.

I have written the routines in Level II Basic and built a simple mainline program which prompts for a date, validates it, calls the Julian date subroutine and the day of the week subroutine, and then prints out the day of the week and the Julian date.

To begin constructing your own subroutine library follow these simple steps:

1. Enter and test the program which follows. After it is operational, save the entire program.

2. Remove the mainline logic of the program which is not to be part of your subroutine library by deleting the lines up to but not including 30000.

3. Now save the remaining code on a new tape which will be your subroutine library. Since it is necessary to use both of the subroutines to obtain day of the week, I suggest that both be saved together, and named "DATE".

4. Write up a subroutine description sheet as shown in Figure 1, and file it with the tape.

You now have the first subroutine for use in future programs. Other subroutines you may wish to develop include a vertical histogram, a large character generator, a linear regression routine and an X-Y coordinate graph. As your library gets larger these subroutines will become the building blocks of your future programs.

Just in case you missed the July 1979 issue of Radio Shack's *Microcomputer Newsletter* the six steps to merge your subroutines with a program already in memory are:

1. Make sure the program to be merged (the one on cassette) has line numbers that are larger than the line numbers of the program located in memory.

2. Look at the contents of locations 16633 and 16634 using PRINT PEEK(16633),PEEK(16634). Write down the numbers you see there.

3. If the contents of 16633 is 2 or greater, execute the following statements: POKE 16548,PEEK(16633)-2 and POKE 16549,PEEK(16634). Then go to step 5.

4. If the contents of 16633 is 0 or 1, execute the following statements: POKE 16548,PEEK(16633)+254 and POKE 16549,PEEK(16634)-1. Then go to step 5.

5. CLOAD the program from cassette. Then execute the statements POKE 16548,233 and POKE 16549,66

6. LIST, RUN or CSAVE the merged program.¹

Here's hoping that future articles are going to begin using such routines in their programs so we can reduce some of the drudgery associated with every new program.

¹ "How to Merge Two Programs Using the CLOAD Command", *Microcomputer Newsletter*; July 1979; page 2

² "TGIF Algorithms", Cornish, R A *Computerworld*; June 11, 1979; In Depth/29 - In Depth/32

SUBROUTINE LIBRARY SPECIFICATION SHEET

SUBROUTINE NAME DATE CONVERSION

DATE CREATED 12/31/79

CLOAD NAME DATE

TAPE NUMBER 20

TAPE LOCATION 010

BEGINNING LINE NUMBER 30000

ENDING LINE NUMBER 40170

PASSIVE MEMORY SIZE 250

PURPOSE OF SUBROUTINE This routine is designed to convert Month, Day, Year into Number of Days, Year by entry at 30000. A second subroutine will determine the day of the week based on the previous output by entry at 40000.

RESERVED VARIABLE NAMES

LABEL	PURPOSE	DESCRIPTION
JULIAN DATE		
MO	INPUT	Month (01 - 12)
DY	INPUT	Day (01 - 31)
YR	INPUT	Last two digits of Year (00-99)
DAYS	OUTPUT	Result of Julian Date conv.
P	WORK	
Q	WORK	
R	WORK	
B	WORK	
DAY OF THE WEEK		
DAYS	INPUT	Use for calculation of day of the week
YR	INPUT	Use for calculation of day of the week
DW	OUTPUT	Subscript for the day of the week
DY\$()	OUTPUT	Description for day of the week
Z1	WORK	
R1	WORK	
Q2	WORK	
R2	WORK	
Q3	WORK	
R3	WORK	
FD	WORK	
Q4	WORK	
R4	WORK	
FI	WORK	

Figure 1

```

10 REM * * * * *
20 REM * BASIC DATE CONVERSION ROUTINES BY DAVID R PEPPE *
30 REM * * * * *
40 REM - - - - - MAINLINE TEST LOGIC - - - - -
50 CLS:PRINTTAB(20);"DATE CONVERSION ROUTINES"
60 PRINT"THIS PROGRAM IS USED TO DEMONSTRATE THE DATE CONVERSI
  N"
70 PRINT"ROUTINES. THESE ROUTINES ARE DESIGNED TO TAKE A DATE"
80 PRINT"ENTERED IN MONTH, DAY, YEAR FORMAT AND CONVERT IT TO"
90 PRINT"A JULIAN DATE AND THEN DETERMINE THE DAY OF THE WEEK"
100 PRINT"THAT IT REPRESENTS":PRINT
110 PRINT"AFTER USING THIS PROGRAM, YOU CAN DELETE LINES 10 TO
120 PRINT"220, THEN STORE THE DATE ROUTINES ON DISK OR TAPE. TH
  IS"
130 PRINT"WILL ALLOW YOU TO INCORPORATE THESE ROUTINES IN"
140 PRINT"OTHER PROGRAMS EASILY, USING THE TECHNIQUES DISCUSSED
  "
150 PRINT"IN THE ACCOMPANYING ARTICLE":PRINT
160 INPUT"PRESS (ENTER) WHEN READY";A$:CLS
170 INPUT"MONTH, DAY, YEAR (LAST TWO DIGITS)";MO,DY,YR
180 IFMO<0 OR MO>12 OR DY<1 OR DY>31 OR YR<00 OR YR>99 THEN PRI
  NT"INVALID DATE - PLEASE RE-ENTER":GOTO170
190 GOSUB 30000
200 GOSUB 40000
210 PRINT DY$(DW),MO;"/";DY;"/";YR, DAYS;" DAY OF YEAR";YR
220 GOTO170
30000 REM * * * * *
30001 REM   CALCULATE JULIAN DATE SUBROUTINE
30002 REM * * * * *
30010 P=(MO-1)*30
30020 Q=INT((MO-1)/2)
30030 R=(MO-1)-(Q*2)
30040 B=Q+R+P
30050 IF Q>3 AND R=0 THEN B=B+1
30060 IF Q=0 THEN DAYS=B+DY:GOTO 30100
30070 IFQ<>0 AND (YR+1900)/4=INT((YR+1900)/4) THEN DAYS=B-1+DY:
  GOTO 30100
30080 IFQ<>0 AND (YR+1900)/4<>INT((YR+1900)/4) THEN DAYS=B-2+DY
  :GOTO 30100
30090 PRINT" ERROR - DAYS CANNOT BE COMPUTED"
30100 RETURN
40000 REM * * * * *
40001 REM CALCULATE THE DAY OF THE WEEK SUBROUTINE
40002 REM * * * * *
40010 IF FT=1 GOTO 40050
40020 FOR X = 0 TO 6:READ DY$(X):NEXT
40030 DATA SUNDAY, MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATU
  RDAY
40040 FT=1
40050 Z1=INT((1900+YR)/28)
40060 R1=(1900+YR)-(Z1*28)
40070 IF R1=0 THEN R1=28
40080 Q2=INT((R1-1)/4)
40090 R2=(R1-1)-(Q2*4)
40100 Q3=INT((Q2*5)/7)
40110 R3=(Q2*5)-(Q3*7)
40120 IF R2+R3-7=0 THEN FD=R2+R3-7
40130 IF R2+R3-7<0 THEN FD=R2+R3
40140 Q4=INT(DAYS/7)
40150 R4=DAYS-(Q4*7)
40160 IF R4=0 THEN R4=7
40170 R4=R4-1
40180 DW=(FD+R4)-7
40190 IFDW<0 THEN DW=FD+R4
40200 RETURN

```

.....
 ● **ACCEL2:** Compiler for TRS-80 Disk BASIC.
 ● Compiles selected subset to Z80 machine code in
 ● all four variable types, compact 1K run-time com-
 ● ponent controls interpreter to streamline all other
 ● statements and functions. Technique minimises
 ● code expansion without impairing huge speedups
 ● for true double optimisation. Six diagnostic mes-
 ● sages. Local/Global options increase compatibility
 ● with subject programs. Output save to Disk,
 ● instructions for self-contained SYSTEM tape.
 ● Professionals note: No royalties on the derived
 ● code! ACCEL2 brings your BASIC programs alive.
 ● It's like having a 100 mhz clock! **\$88.95**

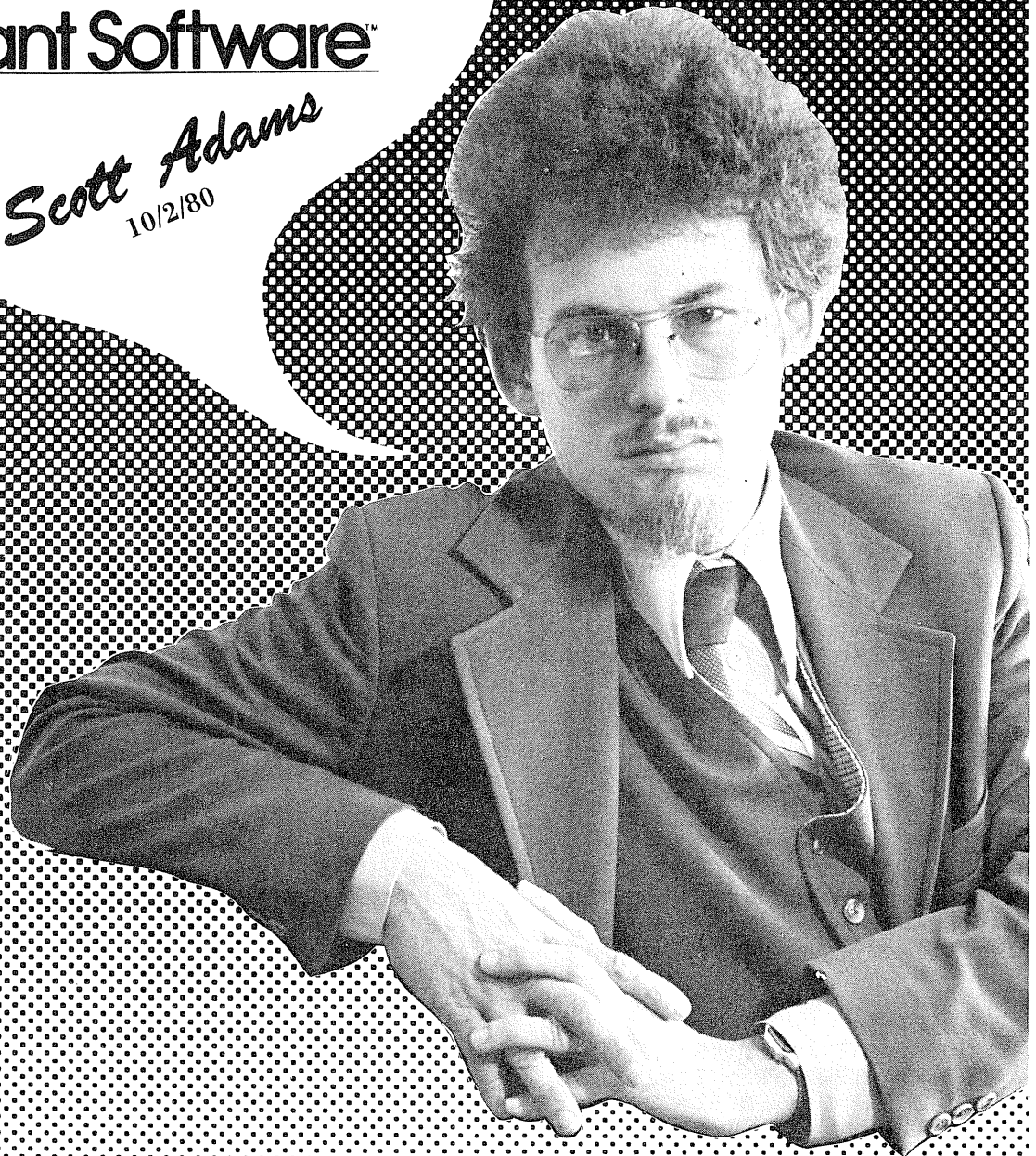
 ● Developed by Southern Software in England,
 ● now available in US from
 ●  **ALLEN GELDER SOFTWARE**
 ● Box 11721 Main Post Office
 ● San Francisco, CA 94101

 ● TRS-80 tm Radio Shack/Tandy Corp

In all my dealings with software
houses the one company
which has always impressed me with its high
quality service has been

Instant Software™

Scott Adams
10/2/80



SCOTT ADAMS,
President of
Adventure International
and author of the
"Adventure" Series.

Instant Software™

PETERBOROUGH, NEW HAMPSHIRE 03458 603-9247296

Instant Software™

Now Sells The Adventure Series*

YOU CAN CALL 1-800-258-5473 TO ORDER THESE NEW INSTANT SOFTWARE OFFERINGS. OR STOP BY ONE OF OUR 300 DEALERS THROUGHOUT THE U.S.A. AND THE WORLD, FOR PERSONALIZED SERVICE.

***AND
MORE**

ADVENTURE!

MORE*

- 1) **ADVENTURELAND** - You wander through an enchanted world trying to recover the 13 lost treasures. You'll encounter wild animals, magical beings, and many other perils and puzzles.
TRS-80 Tape Order No.5501R
- 2) **PIRATE'S ADVENTURE** - "Yo ho ho and a bottle of rum..." You'll meet up with the pirate and his daffy bird along with many strange sights as you attempt to go from your London flat to Treasure Island.
TRS-80 Tape Order No.5505R
- 3) **MISSION IMPOSSIBLE ADVENTURE**
Good morning, your mission is to...and so it starts. Will you be able to complete your mission in time? Or is the world's first automated nuclear reactor doomed?
TRS-80 Tape Order No.5507R
- 4) **VOODOO CASTLE** - Count Cristo has had a fiendish curse put on him by his enemies. There he lies, with you his only hope.
TRS-80 Tape Order No.5508R
- 5) **THE COUNT** - You wake up in a large brass bed in a castle somewhere in Transylvania. Who are you, what are you doing here, and WHY did the postman deliver a bottle of blood?
TRS-80 Tape Order No.5511R
- 6) **STRANGE ODYSSEY** - Marooned at the edge of the galaxy, you've stumbled on the ruins of an ancient alien civilization complete with fabulous treasures and unearthly technologies.
TRS-80 Tape Order No.5512R
- 7) **MYSTERY FUN HOUSE** - Can you find your way completely through the strangest Fun House in existence.
TRS-80 Tape Order No.5513R
- 8) **PYRAMID OF DOOM** - An Egyptian Treasure Hunt leads you into the dark recesses of a recently uncovered Pyramid.
TRS-80 Tape Order No.5516R
- 9) **GHOST TOWN** - Explore a deserted western mining town in search of 13 treasures from rattlesnakes to runaway horses, this Adventure's got them all!
TRS-80 Tape Order No.5517R

ALL 3 PAKS
TRS-80 32K
Apple 48K

3 PAKS
Apple
No.5503AD
TRS-80
No.5504RD

3 PAKS
Apple
No.5509AD
TRS-80
No.5510RD

3 PAKS
Apple
No.5514AD
TRS-80
No.5515RD

STAR TREK 3.5: Get those Klingon's! The newest, most sophisticated Star Trek version by the Grand Master - Lance Micklus. The top program of it's kind available.

TRS-80 Mod.1 L.II 16K
Order No.5518RD \$19.95 Disk
Order No.5519R \$14.95 Tape

SLAG: War gaming at it's best. Real time graphics combined with long range planning make this an exciting, fascinating game.

TRS-80 Mod.1 L.II 16K
Order No.5520R \$14.95 Tape

ASTEROID: The real time, high resolution graphics game that's a smash hit at Arcades all over the world. Three levels of difficulty. Save your quarters.

Apple 2 Disk
Order No.5521AD \$19.95

KID VENTURES: #1 Little Red Riding Hood. Allows your child to interact with the story, learning as they go. Designed for readers and non readers alike. Includes sound and play along cassette tape.

TRS-80 Mod. 1 16K
Order No.5522R \$14.95 Tape to Disk

GALACTIC EMPIRE: Good strategy space war game. You as commander of Galactica's Imperial forces, must capture and hold the 20 inhabited worlds of the Galactic System.

TRS-80 Mod. 1 16K
Order No.5523R \$14.95 Tape
Order No.5524RD \$19.95 Disk

GALACTIC TRILOGY: Special all three games of the Trilogy - Galactic Empire, Galactic trader, and Galactic Revolution all on one disk at a special savings.

TRS-80 Mod. 1 16K
Order No.5525RD \$39.95

★ **INTERACTIVE FICTION:** The computer sets the scene with a fictional situation. Then you become a character in the story. When its your turn to speak, you type in your response. The resulting dialogue and even the plot will depend on what you say.

SIX MICRO STORIES: An introduction to interactive fiction. Involves the reader in a variety of situations from being a spy to a pilot in a doomed 747 and more.

Order No.5526RD \$14.95 Disk TRS-80 Mod. 1 16K

LOCAL CALL FOR DEATH: A detective story considerably more challenging them the above program.

Order No.5527RD \$19.95 Disk TRS-80 Mod. 1 16K

TWO HEADS OF THE COIN: Psychological Mystery set in the London of Sherlock Holmes. Most challenging of all. Will tax your observational and imaginative skills.

Order No.5528RD \$19.95 Disk TRS-80 Mod. 1 16K

Prices: All Tapes 16K . . . \$14.95
All 3 Paks Disk . . \$39.95

DEALERS:

Instant Software is offering you SUBSTANTIAL discounts when ordering these top selling programs. Just call toll-free 1-800-532-5474, to place your order. WE SHIP RIGHT AWAY!!! Call us, if you need any further information.

We Guarantee It!



Instant Software™

PETERBOROUGH, NEW HAMPSHIRE 03458 603-924-7296

9 Z-Subroutines

Spencer Hall

```
0 CLEAR 500: GOTO 10
1 Z=(64-LEN(ZT$))/2:PRINTTAB(Z)ZT$:RETURN
2 Z=960+(64-LEN(ZB$))/2:PRINT@Z,ZB$::GOSUB6:
  RETURN
3 PRINTSTRING$(64,ZC$)::RETURN
4 PRINTSTRING$(64,CHR$(ZG$)):RETURN
5 PRINT@64*(ZL-1),::RETURN
6 PRINT@64*ZP,::FORZ=1 TO(14-ZP):PRINTCHR$(255):
  NEXTZ:PRINT@64*ZP,::RETURN
7 FORZ=1TO345*ZS:NEXTZ:RETURN
8 PRINTTAB(18)"TO PROCEED TOUCH SPACE BAR"
9 Z$=INKEY$:IF Z$=""THEN 9 ELSE RETURN
```

Whether you are a bewildered beginner in Basic or an old China hand in assembly language, I promise that the nine Level II Basic statements above will delight your heart. I call them "Z-SUB's" for a variety of reasons which I will go into later. For now, let's just say that the letter "Z" suggests the lightning speed with which they do some very useful and exciting things on the screen.

These are things which the bewildered beginner in Basic...who has just mastered CLS and PRINT...doesn't expect to do for a long time.

Here is the magic which each one performs:

1. Places the text of any line you write in the exact center of the current line.
2. Places any text in the exact center of the bottom line of the screen...where it will remain until it is written over or erased with CLS.
3. Displays a full 64 character line of any character you choose. This is useful when you want to create a line of hyphens or possibly asterisks to use as decorative patterns on the screen.
4. Displays a full 64 character line of any ASCII graphics symbol you choose. Look at page C/6 of your level II Manual for a catalog of all the patterns you have available. You simply tell Z-SUB 4 which one you want...by the three digit number shown on page C/6. It's very simple,

as we will explain in a moment.

5. Places the next line of text (PRINT statement) you write on the line of your choice (1 thru 16, of course).

6. (My favorite) Erases everything on the screen below a line of your choice and starts your next PRINT statement at the top of the newly erased space.

7. "Freezes" the screen and prints the message: "TO PROCEED PRESS (SPACE) BAR."

7. "Freezes" the screen for any number of seconds you choose.

8. "Freezes" the screen and prints the message: "TO PROCEED PRESS (SPACE) BAR."

9. "Freezes" the screen without printing the above message.

The best part of all this is that you only need to write these statements once. Copy them now and record them on tape. Whenever you set out to write a program in the future simply load them into memory first and then proceed with your program beginning at the traditional statement 10 starting point. When your program is RUN, statement 0 jumps over these one line zingers and begins execution where you want it to. Then, whenever you want to call forth this power, you can do so by simply writing a statement six characters long! There is one small catch to this claim, of course. You must furnish either a text to be centered, a line to which to do to,

a character to repeat or the time in seconds for which you wish to "freeze" the screen, etc. If anything in the above puzzles you, Figure 1 will tell you exactly how to do these things.

If you are already impressed by the power of these one line subroutines, you haven't seen anything yet. We have just begun to describe their power. Now let's look at the full potential of each.

Before we do, however, here's a word about how the Z-SUB's got their name. The old vaudeville and burlesque comedians who later became the radio comics (and still later the standup comedians in night clubs) had a repertoire of short lines that were complete jokes in themselves. They could be slipped into a routine "ad lib" when things were dragging. Several, delivered one after another, were a sure way of thawing out a cold audience. The profession called these little life savers "one line zingers."

So I gave these invaluable pieces of programming the same name. It's a name suggestive of their value, and also a reminder of the fact that they are subroutines complete in one statement.

This name also suggested to me that the various arguments they require should all involve the letter "Z." This guarantees that there will never be any confusion with a variable you might name in your program...if, that is, you stay away from the "Z." That's an easy rule to remember....but be sure to remember it!

SUB #1: CENTER TEXT - This is especially good for titles. Use it right after CLS in a one line statement thus:

```
xx CLS:GOSUB 1
```

Where, you may ask, is the ZT\$="(text)" statement which makes this line work? The answer is that it is back at the beginning of your program. This is the only line you are going to want to center in this particular program, but from time to time you are going to want to head each one with this title. The line shown above...and nothing more...will do it for you, with or without the CLS statement as you wish. Pretty elegant, yes? Of course, you can change what is centered at any time simply by rewriting the ZT\$= statement before saying GOSUB 1.

SUB #2: CENTER TEXT AT BOTTOM OF SCREEN - Nearly everything we said above goes for this one too. When

you are writing a teaching program and have a multiple choice question on the screen for the reader to answer, it's a quick and painless way of reminding the viewer that he should "ENTER THE NUMBER OF THE CORRECT ANSWER."

SUB #3: Use the hyphen (minus sign) to draw a line under a displayed array...columns of numbers, that is...to indicate that the column sums will be printed next. Also asterisks make a pleasing divider for a screen display. Interesting decorative effects also result from using the "greater than" or the "less than" symbols.

SUB #4: The ASCII graphics symbols on page C/6 of the Level II Manual consist of lighted blocks in a pattern two blocks wide by three blocks high. When making a line of these blocks it is wise to remember that the top four blocks print in that part of a screen line normally occupied by printed characters and that the lower pair of blocks occupy the normally dark screen space between two lines of characters. Use symbol 132 to "paint" a solid line instantaneously across the screen. This gives the best separation from lines of print above and below. Many other symbols give disappointingly similar shaped lines. I have not tried them all, but suggest numbers 137, 146, 153, 155 and 183 for noticeably different effects.

SUB #5: Change a line or fill in a blank line on an otherwise full screen without disturbing the other text. Use it with SUB #1 to center a title on a blank screen without the need to think through the required PRINT@ position number. If you remember that there are sixteen lines on the screen it isn't hard to decide that a one line title can be placed by giving SUB #5 line number 8. Use this also with numbers

2 or 3 to lay in a line of something anywhere you want it.

SUB #6: Keep a running header such as a title and/or a set of instructions on the screen while almost anything happens below it. Once you have fed it the last line number you want protected from erasure, GOSUB 6 wipes out everything leaving your header intact. The next text you print will appear below the protected header. Count out the lines you want protected...running the partial program if necessary to see them.

SUB #7: When you want the viewer to read your message carefully...as, for example, profound truths or complex instructions...print a few words or a line at a time and stop the screen to allow the viewer time to think. The statement, GOSUB 7 is all it takes. As a rule of thumb, assuming your viewer is reasonably literate, allow one second per half a line of text. Plan ahead and set ZS=1 or 2 at the start of your program and you'll never have to redefine ZS again.

SUB #9: Tell your viewer at the start of the program that the screen can be advanced after a "freeze" by touching the space bar. Then use this SUB without further comment. There's another, perhaps more important use however. Get those dramatic action keys like "I," "D," "H" and others in the Edit Mode where things happen as soon as the viewer touches the character. Again much more elegant than INPUT with its meaningless question mark.

This SUB captures the key the viewer has touched and defines it as Z\$. Follow with a series of IF statements such as:

```
xx IF Z$="X" GOTO 3500
xx IF Z$="Y" GOTO 4500
xx IF Z$="N" END ELSE PRINT
```

"YOU HIT A WRONG KEY"

```
xx GOTO.....
```

The last statement above would take the program back to where the viewer gets another chance to touch a key.

There they are. Write them once and record them not once but several times in rapid succession. Keep this tape, specially marked, where you can run it in at the start of every new program. Multiple recordings make it unnecessary to rewind between uses. There is always another recording ready to go where the last one ended yesterday. Whenever possible define all the Z-variables you intend to use at the start of your program and then the six easy keystrokes do all the work every time you need them. (You don't even need a space between GOSUB and the number... if you're as lazy as I am.)

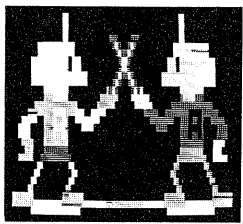
Here are a few stray thoughts in conclusion. You may want to think up some SUBS of your own. I'd like to hear about some new spectaculars that readers come up with. It might be wise not to move step numbers around, however, because you will soon memorize the effects and the numbers which produce them. The grouping shown here helps do that. 1 and 2 center text. 3 and 4 print 64 character lines. 5 and 6 have to do with chosen line numbers. The timer stands alone under the traditional mystery number, 7. Numbers 8 and 9 are a team. If, when your program is complete, you haven't used some of the SUBS you may want to delete them just for appearances sake. The easy way, of course, is to key in its number and touch "ENTER". ●

Figure 1

SUB #	Purpose	Symbol	Meaning
1	Center text on line	ZT\$	Text to be centered
2	Center text on bottom line	ZB\$	Text to be centered
3	Print a line of characters	ZC\$	Character to use
4	Print a line of graphics	ZG	ASCII graphic number
5	Start next text on a chosen line	ZL	Number of chosen line
6	Protect text above chosen line	ZP	Number of last line to be protected
7	Freeze screen for a number of seconds	ZS	Time of freeze in seconds
8	Freeze screen until space bar is touched and print message saying so	--	(None required)
9	Freeze screen until any key is hit	--	(None required)

(Note: #9 captures key which is hit and defines it as Z\$. Use Z\$ in tests for choice of branching.)

THE PROGRAM STORE



new
DUEL
«N»
DROIDS

By Leo Christopherson from Acorn
Your 'droid has already learned NIM, so now it's time to teach it how to wield a laser sword! Leo Christopherson, author of "Android NIM," "Dancing Demon" and other animations, has developed a new type of animation and high-quality sound in his latest work.

Your 'droid starts out as a lowly clown. You teach it how to use a laser sword by controlling its movements. After training it to be a "Grand Master," you enter the tournament against the program's skilled 'droid! Entertainment for all ages.

Protected Tape...\$14.95
Protected Disk...\$20.95

DEATH-MAZE
5000



from Med Systems

A new breed of adventuring! Venture through a graphically represented 3-D maze, with halls that could lead to infinity. Step through the doors or drop into the pits. Will you encounter monsters and mayhem, or will you be treated to useful objects and information? Will you ever get out alive?

You may never find your way out of Deathmaze 5000, but you'll keep trying!

16K TRS-80, 32K APPLE II...\$12.95



HAIL
to the CHIEF

By P. Brasher & R. Vance from Sensational Software

How would you run a political campaign for the highest office in the land? Would you be elected? Find out with this campaign strategy simulation developed by political scientists. Choose (and perhaps change) your positions on major issues as you conduct your campaign, all the while keeping an eye on the weekly polls.

TRS-80 32K Cassette, 48K Disk
Apple II & Apple II+ 48K Disk
Atari 400 32K Cassette, Atari 800 40K Disk
.....\$24.95

TRS-80 Level II 16K
unless otherwise noted



new
BASKETBALL
Dribble Dribble

By John Allen from Acorn

New machine language action game, with sound, from the author of the acclaimed "PINBALL!"

You have to be fast to keep up with the action as you try to outscore your opponent in five minutes of one-on-one basketball. Compete against a friend or your computer.

Steal the ball, duck around your opponent and slant toward the basket for a lay up! The graphics are based on a 3-dimensional depiction of a basketball court, and ball dribbling sounds add to the realism. It's all there but the cheers -- so real you'll wonder how the ball keeps from coming through the screen of your TRS-80! Dribble, Dribble!

Protected Tape...\$14.95
Protected Disk...\$20.95

GALACTIC TRILOGY

By Douglas Carlston

Take control of the Galactica as you navigate through an uncharted 3-dimensional universe. In "Galactic Empire," you attempt to unify a universe that is randomly created each time you play.

"Galactic Trader" pits your bartering skills against those of the other inhabitants as you try to accumulate riches and power. But watch out for the assassins and the energy cartel -- they're out to getcha!

Diplomacy and 'deviousness play equal parts in "Galactic Revolution." It's a game that combines tactics, social manipulation and Machiavellian ruthlessness. For more intrigue, this game allows more than one player. Sound effects.

Choose any game at \$14.95 for TRS-80 16K on tape, \$24.95 for Apple II 48K Disk.

To control the entire universe, get all three!

JET FIGHTER PILOT

from Instant Software

Launch one of several realistic jet fighters from an airport, or catapult from the deck of an aircraft carrier. Incredibly realistic simulation, right down to maintenance problems.

You will not only learn about the dynamics of flight, you'll discover the complex operation of modern military jet aircraft as you sit back and try to keep up with the constantly changing instrument panel display. Challenging and informative.

Cassette...\$14.95

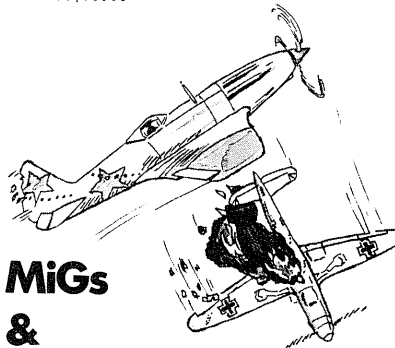
EDAS
Editor/Assembler

By Roy Soltoff from MISOSYS

With EDAS, you are no longer tied to memory limitations while writing in assembly language. Now you can assemble directly from text stored on disk. Branching lets you test your program, then return directly to EDAS. Great for editing and debugging.

Other features include: global editing, upper/lower case support, block moves, plus availability of DOS commands within EDAS. It's the Editor/Assembler designed with the programmer in mind!

Disk...\$79.00



MiGs
&
Messerschmitts

"It is the summer of 1941 and the Blitzkrieg is smashing into the heart of Russia..."

This is how your instructions begin when you become the fighter squadron leader in "MiGs & Messerschmitts", one of four exciting new Discovery Air Combat Simulations.

These World War II re-enactments are historically accurate -- they challenge you to learn the tactics used by the actual combatants! Written in machine language for fast response.

MiGs and Messerschmitts
RAF: The Battle of Britain
Jagdstaffel
Winged Samurai

For TRS-80, Apple II, PET -- 16K...\$19.95

ACCEL & ACCEL II

From Allen Gelder Software

Imported from England, a compiler for TRS-80 Level II Basic (ACCEL) and Disk Basic (ACCEL II). ACCEL lets you compile the integer portion of your Basic programs to fast, efficient Z-80 machine code. ACCEL II compiles floating-point arithmetic as well, and supports Disk Basic.

Both allow a significant improvement in run-time -- up to 3000% faster in some cases -- and improved program security!

ACCEL.....\$44.95
ACCEL II....\$89.95

Visit Our New Store: W. Bell Plaza - 6600 Security Blvd - Baltimore, MD



TO ORDER CALL TOLL FREE 800 424-2738

For information
Call (202) 337-4691

THE PROGRAM STORE

4200 Wisconsin Avenue NW, Dept. KR6 Box 9609
Washington, D.C. 20016

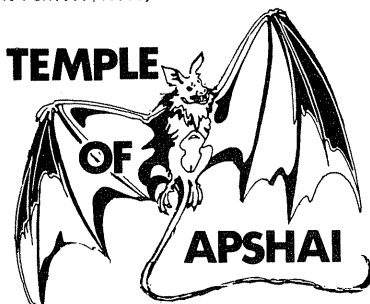
28 80-U.S. JOURNAL Jan/Feb 1981

MAIL ORDERS: Send check or M.O. for total purchase price, plus \$1.00 postage & handling. D.C. residents, add 5% tax. Charge card customers: include all embossed information on card.

POKER PETE

By David Gubser from Quality
Practice up for your weekend poker game
against animated Pete. He shuffles and deals,
then plays five card draw against you. Pete
will bluff, raise call or fold. But watch out --
Pete's got a gun!

New version with keyboard or optional light
pen input....\$11.95
(Light Pen....\$19.95)



From Automated Simulations
The first of the DunjonQuest series, and still
one of the most popular. In exploring over
200 rooms in the magical labyrinth, you will
encounter more than 30 kinds of fearsome
monsters guarding over 70 treasures. Some of
the treasures will help you in your quest, but
you must still watch out for the many mon-
sters and traps that spring out from the walls
and shadows.

The "Book of Lore" fills in the background
and describes the appearance of the temple as
you go. Test your mettle against the servants
of evil and the infamous Innkeeper as you
play this real-time fantasy simulation. A real
challenge, even for serious gamers.

\$24.95 (includes Book of Lore)

PACKER

From Cottage Software
The ultimate editing tool for BASIC program
lines. There are five commands which allow
easier reading of the program and more effi-
cient execution by the computer.

Specify "PACK" and the program will com-
press text into multiple statement lines up to
the maximum length you specify. This really
speeds up storage, load, and execution time.
It can reduce the memory requirement by as
much as 33% while saving disk or tape space,
too.

The "UNPACK" command breaks multiple
statement lines into single statements with full
spacing for easy reading and editing.

"SHORT" deletes any unnecessary words
(e.g. LET, etc.), and removes all REMarks.

Also included are two handy utilities: "MOVE"
lets you relocate program lines within your
program, and "RENUMB" allows program
renumbering.

Save time, memory and storage space -- order
PACKER today! Versions for 16K, 32K and
48K supplied on two cassettes.

\$29.95

SBT Structured BASIC Translator

By Gene Bellinger from Acorn
Speed up program development and docu-
mentation with structured programming. You
can write your programs using:
CALLS, PROCEDURES, CASECALLS,
IF-THEN-ELSE, WHILE and UNTIL. Once
written, SBT will quickly translate the
structured code into an efficient BASIC
program. The program is fast (a 20K BASIC
program in less than 4 minutes) and compact.
Once you try structured programming, you
may never go back to BASIC.

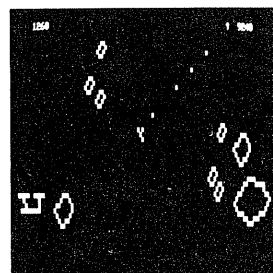
32K TRS-80, Single Disk...\$29.95

ATERM

By Tom Stibolt from Acorn
The complete ASCII terminal program, with
features you want and need: true full-
duplex, compatible with Radio Shack's
RS-232 and Lynx, supports all 128 ASCII
characters including lowercase (if keyboard
has been modified for it), and BELL sound on
AUX line from the computer.

You can set baud rate (on RS-232), parity,
word length and stop bits from the keyboard,
even while receiving. Lineprinter output is
buffered in memory to allow the use of slow
printers to be used without nulls. ATERM is
completely compatible with Radio Shack's
Communications Package.

Tape...\$19.95



**SUPER
NOVA**

By Bill Hague from Big Five
Asteroids surround your ship. You must shoot
the asteroids, as well as any alien spaceships.
Written in fast machine code, this game is
GREAT!

You may encounter five different kinds of
alien ships, including the very deadly flag-
ship. You shoot from your ship's position,
rotate it, use your thrusters to move -- if you
are overwhelmed, you can even get away to
hyperspace. Fast and exciting.

Tape...\$14.95

SPACE WAR

By Device Oriented Games from Acorn
A two-player, real-time action game that lets
each player control a spaceship with rotate,
thrust, fire, and hyperspace. Five game
options (including gravity) and three playing
speeds. In fast machine language.

Tape...\$9.95

DISK*MOD

By Roy Soltoff from Misosys
This machine language program modifies your
copy of Radio Shack's EDITOR/ASSEMBLER
for use on your disk operating system. You
can load and save both text and assembled
object code to disk. And unlike the NEWDOS+
version, you can read the disk directory, kill
files, and determine both space used and
available without exiting EDTASM.

Other capabilities include: Block moves for
relocating sections of text. Global change,
which permits changing a label, for example,
throughout the text. Pagination lets you list
your program neatly on 8-1/2 X 11 pages. In
addition, high memory can be reserved to
allow for machine language routines such as
printer drivers.

DISK*MOD allows lowercase input, branching
to any address, and a functional [CLEAR]
key. It causes the symbol table to be
alpha-sorted and to be output 5-across, and
improves the format of "DEFM". Get all these
features and more, plus corrections to errors
in the Radio Shack program -- upgrade your
EDITOR/ASSEMBLER with DISK*MOD today.

Tape...\$19.95

DDT Disk Drive Timer

GRAPHIC DISPLAY OF MOTOR SPEED			
DRIVE NO.:	0	RPM RANGE:	10
EACH MARK REPRESENTS	0.17 RPM.		
(SLOW)	(CORRECT)	(FAST)	
295.00	296.67	298.33	300
		301.67	303.33
		305.00	

from Disco-Tech
Analyze and adjust your disk drive motor
speed with a real-time graphic display. Man-
ual details use for Radio Shack, Shugart,
MPI, Pertec and Vista drives, and DDT can be
used with any drive. All you need is DDT,
two screwdrivers and five minutes.

Disk....\$19.95

SARGON II

By Dan & Kathe Spraklen from Hayden
Acclaimed the best of the microcomputer chess
playing programs. SARGON II came in third
in the 9th North American Computer Chess
Championship, playing against much bigger
machines! You haven't really played chess
against your computer until you try SARGON
II.

Tape...\$29.95 Disk...\$34.95

PROGRAMS UNLIMITED...

...if you don't see the program you'd
like, give us a call -- we probably have it!

NEWDOS+.....99.95	VTOS*.....99.95
NEWDOS/80*....149.95	RADEX 10*.....99.95
Level III BASIC.49.95	CCA Data Mgr*....75.95
MMS FORTH /Man*.79.95	SYSTEM SAVERS....14.95
KeyEdit*.....18.95	Tiny PASCAL*....50.00
Acorn w/Sound (each):	Tape 14.95 Disk 20.95
Pinball	Pigskin StarTrek Ting-Tong
Invaders From Space	Alien Invasion
Language Teacher (Fr, Span, Ital.)*	19.95 ea.
OSBORNE & ASSOC. Business Systems*:	25.00 ea.
General Ledger	Accounts Payable
Accounts Receivable	Payroll w/o cost acct.

* Disk

THE PROGRAM STORE • Dept KR6 Box 9606 • 4200 Wisconsin Ave, NW • Washington, D.C. 20016

Item	Price	Postage \$1.00	name	
		Total	addr	
		<input type="checkbox"/> CHECK <input type="checkbox"/> VISA	city	state zip
		<input type="checkbox"/> MASTERCARD MC Bank #	Card #	Exp



Bob Liddil

Copyright 1981 Bob Liddil

Here's Captain 80, Software Super Hero, strapping his seat belt for a flight into the unknown reaches of outer space. I have all the necessary items required for space travel but, like all good astronauts I must go down the checklist.

Space suit. Check

Flashlight, suitable for use in vacuum. Check.

Canteen containing one quart of Kool-Aid. Check.

Book: "One Thousand Ways to Say I Want My Mommie." Check.

Teddy Bear. Check.

TRS-80 and one copy of Super Nova. Check.

Hotline to the Deity in charge of space men at war. Check.

Prepare to blast off. Check...No... Wait a minute...

Mommieeeeeeeee.

SUPERNOVA, from Big Five Software is a fine, fun, program. Its graphics work is smartly done, competent and realistically true to its "drop in a quarter" arcade cousin. The opening display is a dazzling starburst that stretches TRS-80 graphics to its limits. The game body display is multi-movemental, the little meteors scurry hither and yon, dutifully fragmenting when touched by the high intensity laser fire laid down by your ship.

There are aliens for the more energetic targetiers. One of them shoots back with the most realistic lightning bolts I've ever seen. Wait a minute. He's shooting at me? Captain 80 calling New Chicago. Get me out of here!!

"The shark burps as he munches the rest of the boat"

That's better. Here's Captain 80 enjoying the tranquil sea breezes of Salachi Bay. I'm aboard a medium size fishing boat searching for a Great White Shark. **THE TERROR OF SALACHI BAY** from Lakefront Software or **SHARK ATTACK** as it is billed by Adventure International and Soft Sector Marketing (its two principal distributors), is a primitive dot to dot search program not unlike the old Hunt The Hurtle. But there the similarities end. The movements are INKEYed arrows for ease of operation and the interaction messages are entertaining. But don't try passing this one on to your fifteen year old. It is designed specifically for the little ones.

We are leisurely floating on Salachi Bay. There is a munching sound coming from the back of the boat. OH NO! IT'S THE SHARK! I dive overboard. While the shark is dining on our \$40,000 fishing boat, I swim ashore. There is a cave up ahead and a house beyond it. From behind, I hear the shark burp as he snacks on the last of the boat and glances my way. DARN! Just when I thought it was safe to go back to my computer....

Now where am I? The note inside my mailbox says: "Welcome to Zork." **ZORK?** Oh yes, the Underground Empire, Personal Software's smash new disk Adventure. Good, maybe I can get some gold or something.

I went into the house through the front. It was dark and my computer warned me of the presence of Grues. Grues have sharp teeth and consider adventurers to be a delicacy, says my Zorkian encyclopedia. Quickly I typed **PRAY FOR EARLY SUNRISE**, followed by **GO WEST AS FAST AS YOU CAN BOGGIE**. Since Zork accepts multi-image commands, it executed my orders gracefully. Safe. Now for a little exploration.

ZORK has undertaken the task of providing the best of traditional adventuring combined with the kind of imaginative descriptions and lavish vocabulary normally reserved for big memory systems. It succeeds greatly.

Down in the caverns of The Underground Empire, and armed with an elven sword of no small magical ability, I discovered a dam, holding back a large lake. Being the curious knave that I am, I entered a control room full of colored buttons. I love button pushing. I have died from button pushing so many times that I am enshrined in the Button Presser's Hall of Fame. Naturally, I began pressing buttons, pulling levers...

I drown in a swift rush of water and am reincarnated into the upper Zorkian forest. Everything has been stolen from me. But I shall press on.

There is a helmet on my head, a control stick in my hand, and more dials and buttons in front of me than I have ever seen before. I wonder what's happening.

"F-15 niner four five," says a voice in my ear, "you are clear for launch in seven seconds - good luck."

I know where I am. I am in Instant Software's *JET FIGHTER PILOT*, part of their new Flying Circus game disk package. I am in *big* trouble. Of all the air flight simulations available, this one is the toughest even for a grizzled

veteran of computer combat.

The launch sequence consists of trimming movable flaps, setting brakes, spinning turbines, ignition, fuel, throttle, afterburner, stick pitch, read thrust, LAUNCH!

Once airborne you have full F-15 armament capabilities with which to attack the enemy, radar, launch computer and...hold on, I smell smoke.

The *AIR TRAFFIC CONTROLLER* from Creative Computing's Sensational Software tenses as he hears my terse mayday.

"Control, this is F-15 Niner Four Five, reporting fire and engine failure, altitude, three five zero, speed seven hundred, heading two four niner. I'm over the city, I'll stay with the plane until we're over the open water."

"Roger Niner Four Five, we have you on radar, don't stay with her too long. Good luck."

I set the controls for level flight. The plane is beginning to fall apart. Ah, there's the ocean. Cut throttle, set flaps, drop speed brake, lower wheels, pray, push ejection button. Hmmm, that's how I got into this in the first place.

The plane arcs downward into the ocean and I swing peacefully from The Software Exchange *PARACHUTE* program sinking softly toward a new piece of software, I can't make out the title. Wait a minute! My luck must be changing. I'm falling into *INTERLUDE*! Now where is that checklist?

The preceding exercise in fancy represents just a few of the hundreds, nay, thousands of programs that are available to enrich your TRS-80. They are possible because you, the software consumer, are rejecting the philosophy of swapping pirated copies. There is a trader in Massachusetts, who includes on his "have" list, the complete library of Greg Hassett Adventures. When the pirates stoop so low as to steal education fund money from a fourteen year old, then it is time we all put a stop to them once and for all by saying NO to their advertisements, and initializing a campaign of letter writing whenever we get their addresses to let them know how we feel about them. Anyone wishing to reply to this radical opinion, which is mine, may write to Captain 80, Box 66, Peterborough, NH

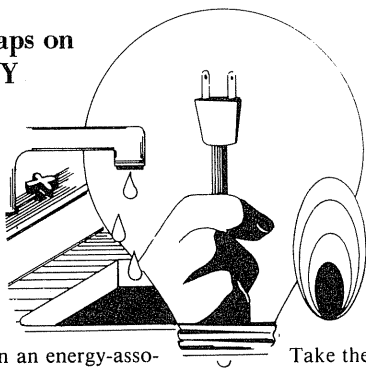
These Penny-Pinching Programs Help You Beat High Energy Costs!

Designed for Home and Commercial Use

Here's your chance to put the wraps on staggering energy costs. **ENERGY AUDIT** helps you inspect any home, analyze your findings and make fuel-conservation investments that result in big savings.

1. Energy Audit

Whether you're a home owner or involved in an energy-associated business, this is a "must" program in accurately determining heat loss. This program creates a computerized model of any dwelling. It will describe what's needed, where it's needed and the estimated costs to make a building an energy miser. Requirements: TRS-80, Level II, 16K; Expansion unit with at least 16K for disk-based version; 1 mini-disk drive; a compatible DOS and an optional printer. Order No. 0052 RD, disk, \$75.00 or 0089R, cassette, \$49.95.



ENERGY CONSUMPTION lets you monitor and manage fuel use for maximum cost efficiency. This combination will mean more dollars in your pocket in these uncertain times.

2. Energy Consumption

Take the guesswork out of energy consumption and conservation. This program keeps utility bills for a 5-year period. Records precise amounts used and costs for natural gas, water and electricity. Keep track of energy costs, examine seasonal fluctuations and evaluate conservation efforts you've undertaken. The soaring cost of fuel requires careful energy management. This program can make you a tight-fisted professional! Order No. 0132R \$9.95.

Instant Software™

PETERBOROUGH, N.H. 03458

THEY'RE EASY TO ORDER. . .

- See Your Instant Software Dealer, or
- Call Toll-Free 1-800-258-5473.

HOMES for the TRS-80

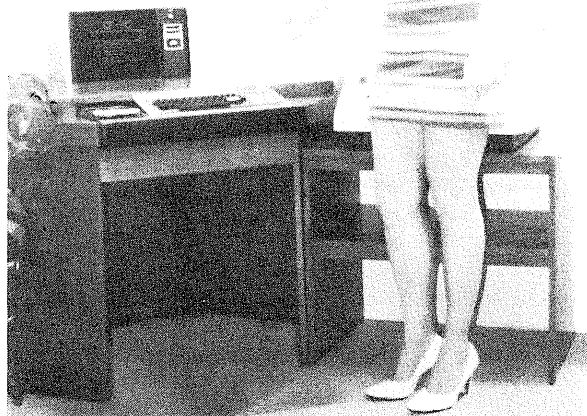
Featuring

High Quality, Commercial,
and Economy Models

Computer Consoles from \$129.95
Printer Stands from \$39.95

Complete TRS-80 Business
System, built-in.

Special Educator Series



**Custom furniture for the
TRS-80 office or home decor.**

AVS is a manufacturer, dealer, and consultant who specializes in products for the TRS-80 and their uses. You will note we do not handle multiple brands of each type of product. When we decide to handle a product, we try to analyze all the manufactures of that product. Then we select the manufacture we feel provides the greatest features, performance, and reliability relative to the cost and needs for the TRS-80. Though many times there may be more than one manufacture that meets our requirements, we prefer to only handle the one, and give our customers maximum support in the use of that product relative to the TRS-80.

AVS Custom Computer Furniture: Home for the TRS-80 takes the orphan modules of the TRS-80 and turns them into one homogenous unit. Our economy, commercial, and high quality consoles all house the keyboard, interface, cassette, and monitor. In addition, we have a special series for the schools. Our lines consist of computer consoles, line printer stands, storage hutches, carrels, and booths.

VISTA Disk Drives: The Vista drives are preferred by many system specialists, including ourselves, for their speed and reliability. They are available in both 40 and 77 tracks, providing memory in excess of 197K per disk. In addition, a module is available that allows the TRS-80 to run the Vista drives at double density, thus, providing more than 394K of memory per disk.

EXATRON Stringy Floppy: The stringy floppy is the ideal mass storage system for most home, school, and some small business systems, because of its simplicity of operation, high reliability, low cost, and speed. Unlike disk, it does not require the expansion interface nor the 11K of computer memory, to perform its many similar functions.

BASE II Line Printer: This printer performs many functions not even available in printers three times its price. Besides having a vertical density of 144 half dots to the inch, you can program your own character fonts. In addition, we have a special modification that allows the printer to run without an expansion interface.

AVS Green Thing: Our green screen works on both models I and II of the TRS-80. It not only performs the same tasks, but costs one-half to one-third of its competition.

Miscellaneous Accessories: For our customer convenience, we provide a series of AC outlet strips, line filters, and cooling fans.

AVS

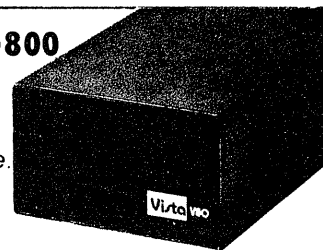
**AUDIO-VIDEO
SYSTEMS**

**2485 AUTUMNVALE AVE.
SAN JOSE, CA. 95132
PH. 408-946-1265**

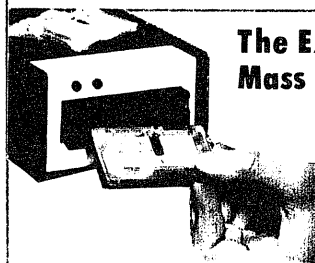
PRODUCTS for the TRS-80

The VISTA V-80 & V-800 Disk Drive System

- 40 & 77 Track Units
- 12 msec Access Time.
- High Reliability.
- Double Density.
- 394K per Diskette.



List \$395.00



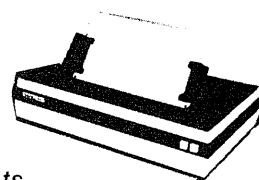
The EXATRON Stringy Floppy Mass Storage System

- Access 15 x Cassette.
- Simplicity of Operation.
- Direct Keyboard Cont.
- Uses only 1K Memory.
- 60K per Waffer.

List \$249.50

The BASE II Graphic Line Printer

- Tractor/Friction Feed.
- 5 Character Densities.
- Programable Aux. Fonts.
- 2K Buffer
- 72V x 99H Dot Res.



List \$699.00

The AVS Green Screen Thing SC-80

- Improved Contrast.
- Reduces Eye Fatigue.
- Enhances Legibility.
- Decreases Glare.
- System Looks Prof



List \$7.95

(Add \$100 for shipping & handling, and California residents add 6% sales tax)

Miscellaneous System ACCESSORIES

- AC 3 Wire, Multi-Outlets.
- AC Line Filters
- Cooling Fans



OMNI-KEY: The Utility for Mere Humans

Mere humans. Sounds insulting, doesn't it? But the fact is, our computers tower over us in one principal virtue. Patience. They can await input for days on end without becoming bored. They can digest DATA statement after endless DATA statement and not once complain of the tedium. They endure our most serious blunders with aplomb. And we humans? We curse the monotony of program entry, mutter at our clumsiness with EDIT, and rail at Tandy for their %&!% inadequate keyboard. *Aargh!* Computers are supposed to relieve this tiresome aggravation, not intensify it! Why doesn't somebody do something?

We have. We wrote OMNI-KEY. And if you had OMNI-KEY, your programming would not only be less tiresome, but more productive. How? Well, when was the last time these little annoyances got under *your* skin?

KKey BBounce. OMNI-KEY eliminates it.

Repetitive Keying of the Same Character. OMNI-KEY has autorepeat. Hold *any* key down, and it repeats about eight per second.

Typing Out Common Keywords. OMNI-KEY lets you assign BASIC keywords to the SHIFT-letter keys. Type SHIFT-P, for example, and you get PEEK(), or whatever *you've* made that key represent.

Repetitive Keying of Similar Phrases. Have you typed "DATA x, y, z ENTER" one too many times? OMNI-KEY's macro key types the repetitive stuff with a single keystroke. You just fill in the blanks. In fact, it's possible to enter hundreds of DATA statements in a row without typing line numbers, "DATA", or the commas! OMNI-KEY's unique macro pause and macro repeat make it possible. And you program the macro key any time and any way (up to 80 characters) you see fit.

The "What's on the right of the cursor?" EDIT Mode. Come on. You don't *need* to put up with this half-blindness when editing a program. If the statement is listed on the screen (even a multi-liner), OMNI-KEY lets you edit it *in place and in full view* with its movable cursor. And you don't need any fancy commands to do

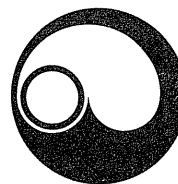
it, either. To insert characters, just type them -- the lines will shift to accommodate them. Deletions are even easier -- just hit the CLEAR key. Need to move a statement? Just edit the line number! It's that easy.

Separate Drivers for Lower-case, Printers, Video Display, etc., etc. OMNI-KEY has its own lower-case driver and shifting built in. But the real beauty of the beast is what you can add to it. If you can use the Editor/Assembler, you can write your own OMNI-KEY modules. OMNI-KEY has a configuration mode which reads your specially-assembled SYSTEM tapes and merges them with the OMNI-KEY functions into *one single load module*. OMNI-KEY, in its standard and customized forms is equally at home with Level II or Disk BASIC, and you don't even have to set MEMORY SIZE to use it! Just enter BASIC, LOAD or CLOAD OMNI-KEY and RUN. It activates itself, reserves its own memory, and waits in the background until needed. Pretty simple.

Simplicity. That's the power of OMNI-KEY. It's simple, it's easy to live with, and it lets you, the programmer, do what you do best. Program. Without the tedium, without the aggravation, and, best of all, without spending a lot of bucks. At only \$23, OMNI-KEY has got to be the best deal going! And if you're a mere human, that's something to think about.

OMNI-KEY cassette for TRS-80 Model I, Level II and Disk BASIC, instructions, postpaid to any U. S., Canadian, or Mexican address. Others are F. O. B. Port Townsend. VISA and Mastercharge are welcome. Dealer inquiries are invited.

\$23



**DISCOVERY
BAY
SOFTWARE
CO.**

P.O. Box 464 Port Townsend, WA 98368

Machine Language Utility Routines for the BASIC Programmer

Jay G McKinney
Seattle, WA

Some time ago I wanted to sort a list of names and save the alphabetized list on a tape cassette. Sorting with a Basic program was slow. And, recording and reading back the sorted list with PRINT#-1 and INPUT#-1 statements was slower yet.

To solve this problem, I wrote three machine language routines and then imbedded them in Basic strings so they could be used as part of any program. The three routines are:

STRING SORT ROUTINE - re-orders the elements of a string array into ascending alphanumeric order.

TAPE WRITE ROUTINE - writes out an entire string as a single record.

TAPE READ ROUTINE - reads back records written by the tape write routine.

All three routines are designed to process a single dimension string array. Except for the amount of storage space available, there is no limit to the size of this array. The two demonstration programs accompanying this article give the machine codes for the routines and show how to use them. The routines are written for the Model I TRS-80 with Level II Basic.

String Sort Routine

The string sort routine utilized the commonly known "bubble sort" technique. Sorting of the array is accomplished by exchanging the "pointers" that point to the individual string elements. If the array contains one or more null (empty) strings, they will be placed at the end of the list. The routine will sort 200 random length (1 to 25 characters), random data strings in approximately 15 seconds, or 400 strings in about 72 seconds.

Tape Write Routine

The tape write routine writes out an entire string array in one operation. Prior to writing any data it computes the total number of bytes (characters) in all of the string elements. It then writes out this number followed by the byte count for each string. When this is done, it writes out a special "check byte" followed by all of the strings followed by another special "check byte". The check bytes are used when reading the data back in. The routine will write out two hundred, 30 characters strings, in approximately 100 seconds, or about nine times faster than PRINT#-1 would.

Tape Read Routine

The tape read routine is designed to prevent the user from clobbering his/her program. It does this by making two checks before reading the data. It checks to make sure there is enough storage space (by using a CLEAR n statement), and it checks to see if the size of the array (into which the data is being read) is the same as when it was written out. In addition, unlike the other two routines, the tape read routine returns a value to the Basic program when it is finished executing. This value is referred to as the Return Code, and has the following meaning:

Return Code 0 - Indicates everything went OK, no errors occurred.

Return Code 1 - Indicates that either the dimension of the array when the data was written out is not the same as the array into which it is being read, or it means that incorrect or garbage data was read from the tape.

Return Code 2 - Indicates insufficient string storage space.

If a return code of 1 occurs, the array into which the data is being read may contain all, some or none of the expected data. If the return code is 2, the entire array will be empty (contain null strings).

(Note to machine language programmers: Before reading the tape, the tape read routine executes the Level II ROM instructions which compress the string storage area. This makes

the maximum number of consecutive memory locations available which in turn, allows all of the strings to be read as if they were a single block of data.)

Imbedding a Machine Language Routine in a Basic String

The basic procedure for imbedding a machine language routine in a Basic string is as follows:

1. Provide a "dummy" string, large enough to hold the machine language routine. Example:

ZS\$="123456789123456789 etc"

2. Include the decimal values of the machine codes that comprise the routine in one or more DATA statements. The machine codes must not include 0 or 34.

DATA 227,10,43,70,etc

3. Write a routine to read the machine codes and store them in the dummy string:

```
10000 L=VARPTR(ZS$)
10010 J=PEEK(L+1) + 256 * PEEK(L+2)
10020 FOR I = J TO J+122
10030 READ K:POKEI,K
10040 NEXT I
```

Once the machine codes have been stored in the dummy string, the DATA statements and the statements to read and store the machine codes may be deleted from the program.

Executing a Machine Language Routine

To execute a machine language routine, first tell Basic where it is with statements like this:

```
K=VARPTR(ZS$)
POKE 16526,PEEK(K+1:POKE 16527,PEEK(K+2)
```

Where ZS\$ is the string that contains the machine codes for the routine. Then execute:

```
X=VARPTR(A$(0))
Y=USR(X)
```

Where X is the value needed by the routine and Y is the value which it returns (if it returns one). In our example, which is based on the string sort routine, X represents the memory address of the first string in array A\$. Y represents a dummy variable since the sort routine doesn't return any value.

Sort Demonstration Program

This program reads the machine codes that comprise the string sort routine and stores them in the dummy string ZS\$. Next, it generates string B\$ which contains 255 random data characters. It then uses this string to fill array A\$ with 201 random length, random data strings. These strings are displayed as the array is being filled. The strings in array A\$ are then sorted by executing the string sort routine. The sorted strings are then displayed to show that they have indeed been sorted correctly. A message is then displayed asking the user to delete the data statements and the statements used to read and store the machine codes, and then to re-execute the program.

Tape Write/Read Demo Program

This program reads the machine codes for the tape write and read routines and stores them in two dummy strings ZW\$ and ZR\$. Next, it fills array A\$ with 51 random length strings and then records this array on tape by executing the tape write function. It then executes the tape read function which reads the data back into array B\$. If an error is detected while reading the tape, it instructs the user to rewind the tape and try again. If no errors are detected, it verifies that the data read back is the same as when it was written out by comparing the contents of the two arrays A\$ and B\$.

```

100 ' * * * * PROGRAM TO DEMONSTRATE MACHINE LANGUAGE SORT
200 ' * * * * BY JAY MCKINNEY, 27 FEB 1980
300 CLEAR 8000: DEFINT A-Z: DIM A$(200): S=0
400 '
500 ' * * * * DUMMY STRING TO HOLD SORT ROUTINE
600 Z$="123456789*123456789*123456789*123456789*123456789*12345
6789*123456789*123456789*123456789*123456789*123456789*12345
6789*123456789"
700 '
800 ' * * * * WARN USER TO SAVE PROGRAM
900 CLS: S=1: PRINT CHR$(23): PRINT "WARNING": PRINT "WARNING"
1000 PRINT "CSAVE PROGRAM BEFORE ATTEMPTING TO EXECUTE IT": GOSUB
620
1100 '
1200 ' * * * * READ AND STORE SORT ROUTINE
1300 CLS: PRINT "PROGRAM IS NOW READING & STORING MACHINE LANGUAGE
E"
1400 PRINT "CODES FOR THE SORT ROUTINE"
1500 NT=0: L=VARPTR(Z$): J=PEEK(L+1)+256*PEEK(L+2)
1600 FOR I=J TO J+122: READ K: POKE I,K: NT=NT+K: NEXT
1700 '
1800 ' * * * * VERIFY DATA STATEMENTS WERE ENTERED CORRECTLY
1900 IF NT=16736 GOTO 2300
2000 PRINT "DATA STATEMENTS CONTAIN ONE OR MORE ERRORS"
2100 PRINT "CORRECT STATEMENTS AND RE-RUN PROGRAM": STOP
2200 '
2300 CLS: PRINT "SORT ROUTINE HAS BEEN SUCCESSFULLY STORED"
2400 PRINT "IN STRING Z$": GOSUB 620
2500 '
2600 ' * * * * MACHINE CODES FOR SORT ROUTINE
2700 DATA 227,217,205,127,10,43,70,43,78,11,197,229,203,131,197,
229,253
2800 DATA 225,229,253,78,3,253,70,4,197,221,225,253,78,6,253
,70,7
2900 DATA 197,253,70,2,253,78,5,253,225,4,12,203,211,221,35,253,
35,5,40
3000 DATA 48,13,32,6,203,83,32,41,24,12,203,147,253,125,221,
190,255
3100 DATA 40,230,48,27,253,225,253,229,6,3,213,253,85,2,253,102,
5,235
3200 DATA 253,114,2,253,116,5,253,35,16,239,209,203,195,225,193,
35,35
3300 DATA 35,11,120,177,32,157,225,193,203,67,32,147,217,227,201
,201
3400 '
3500 ' * * * * GENERATE & DISPLAY 201 RANDOM LENGTH STRINGS
3600 CLS: PRINT "PROGRAM WILL NOW GENERATE AND LIST 201 RANDOM LE
NGTH,"
3700 PRINT "RANDOM DATA STRINGS": GOSUB 620
3800 FOR I=1 TO 255: B$=B$+CHR$(RND(33)+57): NEXT
3900 FOR I=0 TO 200: N=RND(25): P=RND(256-N)
4000 A$(I)=MID$(B$,P,N): PRINT A$(I): GOSUB 650: NEXT
4100 '
4200 ' * * * * SORT CONTENTS OF STRING ARRAY A$
4300 PRINT "PROGRAM WILL NOW SORT THESE STRINGS": GOSUB 620
4400 '
4500 ' * * * * POKE SORT ROUTINE ADDRESS
4600 K=VARPTR(Z$): POKE 16526,PEEK(K+1): POKE 16527,PEEK(K+2)
4700 '
4800 ' * * * * EXECUTE SORT ROUTINE
4900 PRINT "PROGRAM IS NOW SORTING - - - SORTING - - - SORTING"
5000 Y=USR(VARPTR(A$(0)))
5100 PRINT: PRINT "SORT HAS FINISHED"
5200 '
5300 ' * * * * DISPLAY SORTED STRINGS
5400 PRINT "PRESS ENTER TO DISPLAY SORTED STRINGS": INPUT Z$
5500 FOR I=0 TO 200: PRINT A$(I): GOSUB 650: NEXT
5600 '
5700 IF S=0 GOTO 5900
5800 PRINT: PRINT "NOW DELETE STATEMENTS 80 THRU 330 AND RE-RUN PR
OGRAM": STOP
5900 PRINT "END OF SORT DEMONSTRATION": GOSUB 620: GOTO 300
6000 '
6100 ' * * * * SUBROUTINE TO WAIT FOR ENTER BUTTON
6200 PRINT: INPUT "PRESS ENTER TO CONTINUE": Z$: CLS: RETURN
6300 '
6400 ' * * * * SUBROUTINE TO SLOW THINGS DOWN
6500 FOR Q=0 TO 10: R=R: NEXT Q: RETURN
6600 END

```

These programs are for Model I, Level II, 16K


```

140 CLS
150 PRINT "PROGRAM IS NOW READING & STORING THE MACHINE LANGUAGE
ROUTINES"
160 NT=0:L=VARPTR(ZW$):J=PEEK(L+1)+256*PEEK(L+2)
170 FOR I=J TO J+107: READ K: POKE I,K: NT=NT+K: NEXT
180
190
200
210 IF NT<12400 GOSUB 1020
220 PRINT "TAPE WRITE ROUTINE HAS BEEN SUCCESSFULLY STORED"
230 PRINT "IN STRING ZW$": GOSUB 990
240
250
260
270
280
290
300 IF NT<21785 GOSUB 1020
310 PRINT "TAPE READ ROUTINE HAS BEEN SUCCESSFULLY STORED"
320 PRINT "IN STRING ZR$": GOSUB 990
330
340
350
360
370
380
390
400
410
420
430
440
450
460
470
480
490
500
510
520
530
540
CLS
** * * * GENERATE AND DISPLAY 51 RANDOM LENGTH STRINGS
FOR I=0 TO 50: N=RND(60): A$(I)=STRING$(N,CHR$(RND(42)+48)):
PRINT A$(I): NEXT
** * * * TELL USER TO GET TAPE RECORDER READY
PRINT "INSERT TAPE AND PLACE RECORDER IN WRITE MODE": GOSUB 9
90
PRINT "PROGRAM IS NOW WRITING TO TAPE"
610
620
630
640 K=VARPTR(ZW$): POKE 16526,PEEK(K+1): POKE 16527,PEEK(K+2)
650
660
670 Y=USR(VARPTR(A$(0)))
680 PRINT "TAPE WRITE IS COMPLETE"
690 PRINT "REWIND TAPE AND PLACE RECORDER IN READ MODE": GOSUB 99
700
710
720 K=VARPTR(ZR$): POKE 16526,PEEK(K+1): POKE 16527,PEEK(K+2)
730 PRINT "PROGRAM IS NOW READING DATA BACK FROM TAPE"
740
750
760 Y=USR(VARPTR(B$(0)))
770 PRINT "READ FROM TAPE IS COMPLETE"
780
790
800 PRINT
810 RETURN CODE = "Y: IF Y=0 GOTO 830
820 PRINT "BAD DATA READ FROM TAPE"
830 GOSUB 990
840
850
860 PRINT "PROGRAM IS NOW VERIFYING THAT DATA WRITTEN OUT"
870 PRINT "IS THE SAME AS THE DATA READ BACK"
880 FOR I=0 TO 50: IF A$(I)=B$(I) GOTO 910
890 PRINT "NON-VERIFY OCCURRED": PRINT "A-STRING = ": PRINT A$(I)
900 PRINT "B-STRING = ": PRINT B$(I): GOSUB 990
910 NEXT
920
930 PRINT "END OF DATA VERIFICATION PROCESS": IF S=0 GOTO 960
940 PRINT "PRINT"NOW DELETE STATEMENTS 100 THRU 500 AND RE-RUN
PROGRAM": STOP
950
960 PRINT "END OF DEMONSTRATION": GOSUB 990: GOTO 530
970
980
990 PRINT: INPUT "PRESS ENTER TO CONTINUE":Z$: CLS: RETURN
1000
1010
1020 PRINT "DATA STATEMENTS CONTAIN ONE OR MORE ERRORS"
1030 PRINT "CORRECT STATEMENTS AND RERUN PROGRAM": STOP
1040 END

```

```

540 PRINT "THE FOLLOWING STRINGS WILL BE WRITTEN OUT": GOSUB 990

```


ONE CALL! SHOPPING FOR THE TRS-80*
ALL! YOUR HARD/SOFTWARE NEEDS NOW!

PARTIAL SOFTWARE LISTING

CYLON RAIDER	7.95c	BASIC COMPILER	195.95	SYSTEM SAVERS	14.95c	LOWER CASE DRIVER	9.95c
ANIMATED HANGMAN	9.95c	AIDS III (MODEL I)	69.95d	TING TONG	9.95c	REPEAT AFTER ME	9.95c
SPACE BATTLES	14.95b	AIDS III (MODEL II)	99.95d	DISK UTILITY	19.95c	CUBES	9.95c
LEVEL IV GRAPHICS	14.95b	CALCS	24.95d	PIGSKIN	19.95b	BSF	24.95c
TYPING I AND II	18.95c	CRAS	49.95c	BANDITO	9.95c	REMODEL + PROLOAD	34.95c
INVADERS	14.95c	LINE PRINTER	24.95c	PINBALL	14.95b	REMODEL	24.95c
RAMEDIT	34.95d	BASIC TOOLKIT	19.80	ATERN	19.95c	DOSORT	34.95c
STAR TREK 3.4	14.95c	LIBRARY 100	49.95c	DUEL'N DROIDS	14.95b	COMPROC	19.95c
DOGSTAR	9.95c	ELEC' PENCIL 150.00d	100.00c	SUPER SCRIPT	24.95d	INFINITE BASIC	49.95c
PERSONAL FINANCE	9.95b	ANDROID NIM	9.95c	QUAD	14.95c	INFINITE BUSINESS	29.95c
COMPUTER YAHTZEE	7.95c	AIR RAID	9.95c	T-SHORT	9.95c	COPSYS	14.95c
MEAN CHECKERS	19.95b	BARRICADE	9.95c	T-SHORT PLUS	19.95c	DSM MODEL I	75.00d
LEVEL I RELOCATED	15.00c	RSM-2	26.95d	WORDO	14.95c	DSM MODEL II	150.00d
FINAL APPROACH	7.95c	RSM-2D	29.95d	STOCK MARKET	15.95c	SPIDER MT & LOST GOLD	24.95c
KVP	14.95c	RSM RELOCATOR	9.95c	WHEREAMI	10.95c	SPIDER MT ADVENTURE	24.95d
KVP-232	24.95c	DCV-1 2.2	9.95c	MMSFORTH	79.95d	DEATH DREADNAUGHT	14.95c
RENUMBER	7.95c	RSM II	39.95d	TIME SHARE	89.95d	TRIPLE ADVENTURE	35.00d
XREF	29.95c	TRS FORMATTER	14.95	BOMOKU	14.95c	IRV	25.00b
RENUMX	24.95c	PENMOD	19.95c	DDT (DISK DR TIMER)	19.95d	LOST SHIP	14.95c
OPERA	9.95c	SARGON	19.95c	MICROSOFT ADVENTURE	29.95d	CONEY ISLAND ARCADE	39.95d
CHALLENGE	9.95c	SARGON II	29.95b	EDITOR ASSEMBLER	29.95c	LOST DUTCHMAN'S GOLD	14.95c
ANDROMEDA STRAIN	9.95c	GRIDIRON	12.95b	LEVEL III BASIC	49.95d	SPACE WAR	9.95c
SUPER ADD	24.95c	BATTER-UP	10.95c	DECATHLON ADV	24.95b	AUTO/QEDIT	14.95c
SUPER MULT	24.95c	POKER PETE	11.95c	ST-80 II	79.95d	MORSE CODE	14.95c
INVADERS +	19.95b	FASTGAMMON	19.95b	ST-80 III	150.00d	KEY EDIT	19.95c
ENCHANTED ISLAND	14.95c	LOWBALL POKER	11.95c	MUMATH	74.95d	WIN 21	29.95c
MYSTERY MANSION	14.95c	SKETCH-80	11.95c	PENCIL PAL	35.00c	NEWDOS 35	49.95d
MICRO-OPOLY	7.95c	STARFLEET ORION	19.95b	GRAPHICS INTERPRETER	7.95c	NEWDOS 35+	99.95d
MATCHBOOK FOOTBALL	9.95c	INVASION ORION	19.95b	BATTLE GRID	7.95c	NEWDOS 40	59.95d
THESIS I	25.00b	TEMPLE OF APSHAI	24.95b	V.I.C.	19.90c	NEWDOS 40+	110.00d
BOSS 2.2	29.95b	DATESTONES OF RYN	14.95b	OTHELLO	12.95d	NEWDOS 77+	110.00d
RX	24.95c	MORLOC'S TOWER	14.95b	GUNNERS	12.95d	NEWDOS-80	149.95d
DISK DIRECTORY	14.95d	RESCUE REIGEL	24.95b	CRUSHMAN	12.95d	CP/M	149.95
MAIL LISTER 2	79.95c	HELLFIRE WARRIOR	24.95b	CRIBBAGE	12.95d	FORTRAN	95.00
INVENTORY II	149.00d	MISOSYS DISASSEMBLER	19.95c	TERROR AT SELACHI BAY	7.95c	MACRO ASSEMBLER	95.00

NOTE: c=CASSETTE d=DISKETTE b=BOTH (add \$5.00 for disk version)

PARTIAL HARDWARE LISTING

DISK DRIVES READY TO RUN.....	PRINTERS.....	CABLES (2DR \$25) (4DR \$35) (PRINTER/EI \$35) (PRINTER/KEYBOARD \$52)
MPI 40TR SINGLE \$339	EPSON MX-80 DRIVER INC'D \$645	
MPI 40TR DUAL (IN 1 CASE) \$639	MICROLINE 80 \$595	***** WE PAY THE SHIPPING ON DRIVE AND PRINTER ORDERS *****
MPI 40 DBL HEAD (2 IN 1) \$469	CENTRONICS 737 (LP IV) \$850	
MPI 40 DUAL DBL HEAD \$899	CENTRONICS 779 (LP I) \$500	
MPI 80TR SINGLE \$499	ANADIX DP8000 \$600	ASK ABOUT USED 35 & 40TR DRIVES, EI'S, 16K LII'S, ETC.
MPI 80TR DUAL (IN 1 CASE) \$939	* INDICATES USED RE-CONDITIONED	AND, OUR 10 DAY "NO QUESTIONS" RETURN POLICY ON USED EQUIPMENT

* * * SPECIALS * * *

ULTRADOS* \$89.95
BOSS 2.2 DISK \$29.95
BOSS 2.2 CASS \$24.95

* \$119.95 AFTER LIMITED INTRO PERIOD

LEVEL IV PRODUCTS, INC.

32238 Schoolcraft Road, Suite F4 • Livonia, MI 48154
313-525-6200 Outside Michigan call 1-800-521-3305

Please add \$2.50 for shipping and handling
\$1.50 C.O.D.

Level IV Products Catalog
NEW - SEND \$2 FOR YOUR COPY
REFUNDABLE ON FIRST ORDER

Dealers Orders Welcome



INSIDE

VTOS 4.0 & NEWDOS/80

Pete Carr
Port Orange, FL

No doubt you have heard of both NEWDOS/80 and VTOS 4.0 by now. They are new disk operating systems (DOS) for the Model I TRS-80. NEWDOS/80 is from Apparat Inc., and VTOS 4.0 is from Virtual Technology.

NEWDOS/80 and VTOS 4.0 come with various programs (mostly utilities) on the disk which takes up all available disk space. You will need to make a copy of this disk and save the original as a master. You do not use the original disk after that since accidents may happen requiring you to make more clean backup copies. After making a copy, you should kill or purge any of the utilities which you will not normally be using. This is to make space on the disk for your programs.

The programs supplied on NEWDOS/80 are:

1. SUPERZAP - inspect/change disk or main memory.
2. DISASSEM - disassemble Z80 code.
3. LMOFFSET - move module to a new location.
4. DIRCHECK - inspect and list directory.
5. EDTASM - disk editor/assembler.
6. LEVEL I - level I running in Level II.
7. LVIDSKSL - save and load Level I programs on disk.
8. CHAINTST - sample chain file build program.
9. LCDVR - lower case driver.
10. ASPOOL - automatic spooler.

VTOS 4.0 comes with:

1. PATCH allows patching of errors or making modifications on the disk.
2. KSM/DVR - keystroke multiplication driver.
3. RS232/DVR - RS232 communication driver.
4. PRDVR - printer driver.
5. VTCOMM/CMD communications package.
6. KSR/CMD - keyboard send receive terminal.
7. BASIC/OVN - renumber program.
8. BASIC/OVX - reference utility.

There are numerous other files on VTOS but they are mostly patches for programs that need to be modified to operate correctly with VTOS 4.0 or examples of certain features.

The first obvious observation is that compared to TRSDOS 2.1, things have come a long way. TRSDOS 2.1 had many errors and lacked many features which consumers wanted. A market for a better DOS was just sitting there, waiting to be capitalized upon. (Radio Shack now supplies TRSDOS 2.3, which is very good).

The first company to improve upon and fix the many errors in TRSDOS 2.1 was a company called Apparat, Inc. Their system was called NEWDOS 2.1 and it quickly became the standard system for serious disk TRS-80 users. It was, and still is, a great system. Randy Cook, who wrote TRSDOS, also offered a DOS called VTOS. It had many features, but was not nearly as popular with the consumers as was NEWDOS 2.1.

I recently purchased both NEWDOS/80 and VTOS 4.0. During the last few weeks I have used and tested both systems and will offer some comparisons for those of you who are thinking about buying one of these systems. I will go into quite a bit of detail, but both systems offer so much that I will only hit on what I think are the major points. Perhaps there are things you think should have been mentioned, but it would take a whole book to talk about every possible feature. Some of these comparisons will be my opinion. Included also are comparison figures between the two systems that show the speed differences. I used a stopwatch, and tried to be fair by keeping everything as equal as possible.

NEWDOS/80 and VTOS 4.0 allow you to use any size disk drive from 35 to 80 tracks. These drives may be mixed together in any configuration. They both have a "purge" command which allow you to kill files and programs from your disk much more quickly and easily than previously possible. They both offer commands which allow you to choose which files or programs you wish to copy to another disk and then proceed to copy them without any further assistance from you. This is a great time and finger saver. They both offer user definable system options that were previously inaccessible to the user. One of these system options for NEWDOS/80 allows you to disable password protection so that you have access to programs even if they are password protected.

"The first obvious observation is that compared to TRSDOS 2.1, things have come a long way."

You are furnished with the option to make the disk operate in the "run only" mode. This disables the break key and the user never sees any DOS command, cannot list the program, or accidentally clear the screen. This is a nice safety feature for people using the computer who don't know anything about it except how to enter data, and could prevent them from accidentally bombing the program.

A favorite feature of mine in NEWDOS/80 is the DFG MINIDOS function. By striking the DFG keys at the same time, you are put into a MINIDOS mode. This allows you to do disk commands such as DIR, KILL, FREE, etc., even while in the middle of a program. By typing MDRET you are returned right back to the place in the program where you left off. Very nice, especially with Scripsit! The only thing I don't like about this procedure is that in most

programs the screen is cleared (graphics and all) which makes it hard to continue with your program.

Lately I have been using a great utility along with NEWDOS/80 called "BOSS" (from Level IV Products). While using a program, if you want to check the value of certain variables, BOSS will move the screen to high memory, issue a CLS, then print these variables on the screen for you to check. You are then returned to the exact place you left off in your program, with the screen (graphics and all) restored exactly as it was. This is just one of the many features of BOSS. It would be nice if the DFG MINIDOS command used this screen restoration procedure.

When doing a DIR (looking at the disk directory) you don't have to enter DIR :O(I,A). To see all invisible files and their size, just enter DIR O I A. Both ways work, but the shorthand method is much easier to use, a timesaver in NEWDOS/80. I also like the way NEWDOS/80 stops the screen from scrolling when doing a DIR. You just press ENTER to continue. VTOS 4.0 lacks this feature.

BACKUP does not exist in NEWDOS/80. Instead, it provides the COPY command. It uses codes such as [COPY 0=35 1=40 00/00/00 NFMT CBF CFWO], (NFMT=no formatting is to be done, CBF=check file with operator before copying). This command will copy the programs of your choice from drive 0 (which is 35 tracks), to drive 1 (which is already formatted for 40 tracks).

VTOS 4.0 also allows you to choose which programs you want to copy from one disk to another. You can specify BACKUP :0 to :1 (MPW="PASSWORD," SYS=System, INV=Invisible, VIS=Visible). Depending on what options you choose the programs of your choice will be copied to drive 1.

NEWDOS/80 CHAIN is a great feature. How many times have you wished to be able to automatically, from the moment you turned on your disk drive, see how much space is available on the disk, see what programs are on the disk, load a machine language utility program (such as a sort utility), go to Basic and set memory size and then run the program? All this without your having to touch or enter a thing from the keyboard.

VTOS 4.0 offers a chaining feature similar to NEWDOS/80. It operates much slower, but adds even more versatility. During VTOS's chaining you can make the interface give off a buzzing sound or make the screen flash on and off to catch someone's attention. This is very creative, and I like it.

VTOS 4.0 TYPE AHEAD feature allows you to enter keystrokes before you are normally allowed to. Suppose you are in DOS and want to go into Basic and run a program called MAIL. Just type BASIC [ENTER] and while the computer is loading Basic, go ahead and type [RUN "MAIL"] and press [ENTER]. As soon as the computer gets Basic loaded it will go ahead and run the mail program automatically. It does this by storing whatever you type on the keyboard into memory, and then executes the command the next available chance it has.

The VTOS 4.0 KSM program allows you to define the keys on the keyboard to mean whatever you want them to mean. You can define the keys Q, W, E, as DIR :0, DIR :1, DIR :2, and then by just pressing the control key (which is the [CLEAR] key) and the key of your choice, the computer will automatically execute that command with that one keystroke. This works in Basic with commands like STRING\$, GOSUB, etc.

VTOS 4.0 has a graphics packer which is used by pressing [CLEAR] and certain keys. It then prints graphic characters on the screen. This is similar to the utilities you see advertised for the TRS-80. You can define these characters as strings, tie them together and use them in your program by using the Basic command: PRINT X\$.

NEWDOS/80 and VTOS 4.0 both come with lower case drivers built-in, and if you have a lowercase hardware mod, you can use these drivers with either Electric Pencil or Scripsit.

VTOS 4.0's ROUTE, SPOOL and LINK commands allow versatility not available from any other DOS's I know of. Suppose you have a program that LPRINTS to a printer a block of data and you want to have it printed on the video screen instead. Normally, you would have to change all LPRINT commands back to PRINT commands. By using the ROUTE command (ROUTE *PR TO *DO), every LPRINT used in your program would automatically be routed to your screen. This command also works in reverse so you can also route anything that normally goes to the video to the printer.

VTOS 4.0 SPOOLER is a great timesaver when running a program which requires many printouts. It keeps you from having to wait until the printer stops printing before you can continue to enter data through the keyboard. It does this by intercepting the data which is directed to the printer and stores it in memory or in a disk file. It then returns to the program so you can enter more data. While you continue to enter data through the keyboard, the spooler automatically sends the information it had put into memory to the printer. Another example for use of the spooler is that you can LLIST a program to the printer while the computer is running another program. This is super! NEWDOS/80 has a spooler program too, but it is harder and more cumbersome to use. I like the VTOS 4.0 spooler better.

"I will probably end up using both, because both have features that lend themselves to certain programming situations."

The VTOS 4.0 LINK command allows you to link two devices together. By using this you can make the computer send whatever data goes to the video screen go to another device (such as the printer) at the same time. VTOS 4.0 has other commands, such as FILTER, but gives no example of what it does.

Both NEWDOS/80 and VTOS 4.0 allow you to set your disk drives to the fastest motor stepping speed they can handle. Most early Radio Shack drives could only step in 40 milliseconds. Most new drives will step in 20 milliseconds or faster. VTOS accomplishes this by the SYSTEM command: SYSTEM (DRIVE=0, STEP=x), where x is a choice of four stepping times from 40 milliseconds to about 5 milliseconds.

This is a good time to explain a feature of VTOS called

SYSGEN. After you set the disk drive motor stepping rate and other options, such as the blinking cursor, typeahead, lowercase and your definable keys, you type "SYSTEM (SYSGEN)," and your system configuration is saved to disk. The next time you power up the computer, your custom configuration will be loaded with all the system options you previously defined.

Setting the stepping time in NEWDOS/80 is not quite as easy as with VTOS. You are supplied with a "ZAP" which is information that, in conjunction with a NEWDOS/80 program called SUPERZAP, allows you to change disk system information directly from the keyboard. It isn't hard to do, and after you have set your disks to the fastest stepping rate they can handle, will usually find no need to do it again.

By using SUPERZAP and "ZAPS," which are periodically sent out from Apparat, you will be able to fix any newly found errors or change programs which are not compatible with NEWDOS/80. This is a great way to fix bugs without having to get a new version or send your disk back for update.

After receiving the systems I found that they both lacked features which the advertisements intimated they had. A case in point was NEWDOS/80's claim of variable record length up to 4095 bytes long. That sounds great, doesn't it? The Model II computer uses VAR length records and it saves a lot of time, frustration and confusion when setting up random files. It makes the fielding statement much easier to use, etc. Well, NEWDOS/80 does have VAR records, but it is nothing like I thought it would be. It only offers it if you are using one of their new and very different disk I/O procedures. These new procedures are completely different from TRSDOS, NEWDOS 2.1 or VTOS, and you have to start from scratch and learn a whole new way to write disk I/O routines.

Great! Just when I was starting to understand how to use all the standard disk I/O commands I would have to learn a completely new set of commands and procedures if I were to take advantage of NEWDOS/80's new disk I/O file features. From talking to other people I was not the only one who had this misunderstanding. Granted, NEWDOS/80 still supports the old standard disk file commands, but they don't have the variable record length feature.

I think making things easier to use and more understandable to us non-computer-technicians should be on or near the top of everyone's list. One of the computer's main selling points is that it is supposed to take away some of the drudgery, frustration and extra work which people have to go through. Shouldn't this apply to programming the computer as well?

In any case, if you have enough patience to learn how to use the new disk file commands, they look to be many times more versatile and powerful than the standard TRSDOS or VTOS disk file features. Instead of just two file types (sequential and random) there are 5 new types. They are called MF, MU, FI and FF files. I'm sure it is not as hard as it looks and would be well worth the time it takes to learn them.

VTOS 4.0 does offer standard variable length records, but does not show you how to incorporate them into your programs. The manual is very skimpy and lacks much information needed to really take advantage of this

system. So far, no one I've talked to can help with this quandry.

VTOS 4.0 advertisements claimed "FASTER" - improved loading times up to 1400%." That sure is a big improvement in speed, isn't it? But an improvement compared to what? If anything, VTOS 4.0 is one of the slowest DOS's I have ever used. If it is 1400% faster, they must have compared it to a previous version of VTOS, because my comparison test showed that NEWDOS/80 is much faster than VTOS 4.0.

One of the first things I did with VTOS 4.0 was enter the date after I booted the system up. VTOS didn't display the date using the standard "09/01/80" format but displayed it in the format "Tuesday, September 1, 1980." Now that really looks great. Somebody took the time and effort to make the date easier to read. But wait! September 1st, 1980 is a MONDAY, not a TUESDAY. What a great introduction to VTOS 4.0! Well, I called about this, but no one knew what I was talking about. I explained to my retailer and he came up with the same date error I did.

I started to wonder what else was wrong with this system that slipped past the author before putting it up for sale. I could just see myself entering the 700th entry into my General Ledger and the system coming back and telling me that "27 entries are now in the system, press enter to continue." Well, this date problem was fixed in a few days, but instead of fixing it right, they just took the day off of the display so that it would read: 'September 1, 1980.' If that's not a shortcut around a problem I don't know what is. I don't want to seem picky, but isn't it the little things that make up the whole?

I know it takes a lot of time to write a DOS system, and anyone can make mistakes, but every stereo system I've bought worked without error. If it was defective it was taken back and exchanged for one that worked. They didn't just take one of the speakers out and say, "just use it with one speaker."

Well, back to the story. Further use of VTOS 4.0 found more errors and omissions. One feature I was excited about was the PR/DVR utility which would allow you to set custom parameters for your printer. Here is the example from the small VTOS manual:

LINES=x -- to establish the maximum number of lines per page (such as 6 for mailing labels).

PAGE=x -- to establish the physical page's line count.

CHARS=x -- to establish the maximum number of characters which will fit on line.

INDENT=x -- to set the indentation level to be used for lines which exceed x characters in length.

Immediately I pictured a nice program listing that didn't run across page boundaries, looked real neat and was easy to read. Great! I'd been looking for something like this and here it was included with the system. It didn't work! Every time I tried to use the page command it came back with an error. I again checked on this with my retailer, but he hadn't ever used this feature of VTOS 4.0. Finally, after calling around the country, I got an answer from the main VTOS distributor in Dallas, Texas. He admitted indeed that the page command didn't work, but they would try to fix it in the future.

NEWDOS/80 didn't come with any obvious error or omissions that I could see. But there had been some, because they included a ZAP sheet (information for modifying the disk) which they explained how to use and

what each ZAP was for. At least there was no waiting or calling on the telephone.

OK - you are probably thinking by now that I don't like either of these systems. The truth is that they are both fantastic. They both offer advanced and versatile features that are very innovative and show themselves to be the work of very knowledgeable and creative authors. Matter of fact, they add features that even the more powerful TRS-80 Model II doesn't have! They are both so good I am having a very hard time choosing between them as to which system will eventually be my main operating system. I will probably end up using both, because both have features that lend themselves to certain programming situations.

I like to think of NEWDOS/80 as lean and mean, with features that make programming a pleasure. VTOS 4.0 is very flashy, and features like ROUTE and SPOOLER make it very operator oriented. There is no doubt that when doing disk commands, such as DIR and FREE, that NEWDOS/80 runs circles around VTOS 4.0. When it comes to writing Basic programs, NEWDOS/80 really shines with its ability to move lines from one location to another with the option of keeping your original line or deleting it. It also has a reference and renumber command that is much faster than the similar VTOS commands. NEWDOS/80's Basic shorthand such as E=Edit, L=List, A=Auto, and being able to step through the listing of a program one line at a time is much easier to use and faster than VTOS 4.0.

Which System is Best?

I like both very much. VTOS 4.0 has more features and is overall more versatile, but is much slower. I am leaning toward NEWDOS/80 because of its speed and ease of use in programming, but it is too early to choose yet. If I had a choice, I would love to see the best features of both combined in one system. This dream system would be NEWDOS/80 with features like Route, Spool, Definable keys, Blinking cursor, Graphics packer, Standard variable length records and SYSGEN from VTOS 4.0.

NEWDOS/80 and VTOS 4.0 are still very new and both have allowed room for expanding and adding new features as time goes by. The people who sell VTOS 4.0 will surely fix and modify some of the problems VTOS 4.0 has, probably by the time you read this report. In all fairness, VTOS 4.0 is the newer of the two, and will probably have a few growing pains. ●

Stop Watch Test

All of these tests were done using the same motor stepping rates for both systems. Also the save, load and renumber was done on the same program. The start of the clock was at the moment the ENTER key was pressed.

	VTOS 4.0	NEWDOS/80
From boot to ready	8 sec	3 sec
DIR :0	7 sec	2 sec
Free	6 sec	2 sec
Load Basic	8 sec	3 sec
CMD "DIR :0"	9 sec	3 sec
Load a program	5 sec	2 sec
Save a program	8 sec	3 sec
Renumber a program	14 sec	2 sec
35 track backup	1 min 40 sec	1 min 20 sec

How to make **VERSAFILE** *more versatile*

*Radio Shack's new Versafile is good....
these additions can make it better.*

Mike Zielinski
Rohnert Park, CA

One of the big thrills I wanted with a home computer was the ability to put everything I could think of on files. Everything I owned, did, knew - *everything!!*

After spending hours writing a different program for each file type I encountered, and trying some of the programs in the computer magazines, I found most of them to be fairly structured, requiring rather rigid statements.

So, I decided to give *VERSAFILE* from Radio Shack a try...

What caught my eye was that it was supposed to be very unstructured. I received my copy in about a week and was pleasantly surprised. It seems to be well written and bug free. The documentation is fine and I can't think of any unanswered questions. Like all software to really understand it you have to use it.

There is one small (?) error that I almost overlooked. The description requirements indicate that a 16K disk system will do the trick but I don't think this is true. I have a 32K system and did not worry about memory space until I wondered if all my additions would fit a 16K system. When I checked the size of the unaltered program it was 5937, but the maximum user available memory in a 16K system is 5528. Check me out, but I think you will need a 32K system for the unaltered program, and I know you will if you add all my extras.

Now to begin. *VERSAFILE* is written in BASIC to run on a disk system with up to 4 drives. The purpose of the program is to create disk files in which to store and retrieve statements. The core of the program is data lines 10000-10060 (see figure 1.) that contain 45 key words. The first 8 of these key words make up the disk file names accessed by the program. The next 37 are called "unnecessary words" and are removed from all questions.

Let me elaborate on key words. When entering a statement the only key word important is one of the first 8, the key words that create and name the disk files. When asking a question the term "key word" takes a different meaning. First the question is tested for a file key word. (Yup, one of the first 8.)

Then the total 45 key words in the data lines (10000-10060) are considered "unnecessary words" and are removed from the question. The first 8 key words are also removed from the question. There is no reason to do a string comparison on a key word that makes up the file name. All the statements in that file had to have that specific key word to get there in the first place. The resultant string is a string of key words to be compared against the statements in the files. Example: if you entered "ON WHICH DISK IS THE PROGRAM SKIRACER?" and one of the first 8 key words is "disk" with the remaining 44 included "on," "which," "is," and "the" then the statements in the file named "DISK" will be compared for the words "program" and "skiracer". Get it?

What makes this file system different is that any type of statement can be stored and retrieved. In retrieving a statement, the program compares key words in the question with key words in statements in the files. It then counts the number of matches in a comparison and displays the top scoring statements.

To create a file or add to it, simply enter a statement that includes a file name key word (one of the first 8) and end it with a period. If you do not use a file name key word the statement will be loaded into the last file (the 8th key word) as the default file.

To search a file for information, ask a question including a file name and end with a question mark. If you do not use a file name the default file will be searched for a key word match and the first statement with enough matches will be

returned. Only a few statements will be returned without a file name whereas with a file name the program will return all the statements with enough key word matches.

The third function would, of course, be kill. There are two ways to kill statements. One way is to locate the exact statement you wish to kill, enter the kill command and re-create the statement. The second is to enter the kill command and enter a statement, here though if the statement is too vague more than one statement may be killed (or none).

With both statements and questions, and with the kill function, if more than one of the first 8 key words is present then the first word in the data line string (10000) will be used for the file name. Example:

If line 10000 reads:

10000 DATA " FISH ", "CHIPS",etc....

Then both of these questions will address the file named "FISH."

WHO SELLS THE BEST FISH AND CHIPS?

WHO SELLS THE BEST CHIPS AND FISH?

The niceties of the program include global search, multiple kill command, listing one or all files, output to line printer, and an interrupt key.

Now to make a good program better. I love to program but I still buy many programs. There are many available that for your time and effort you just can't beat. All the programs that I buy I study to learn other programming techniques. Then I just can't resist customizing them to my whims.

Here are some modifications that I have made to this program. Some are major, a few minor, and one maybe.

I have NEWDOS so the first change was to make it run on that system. I found the only changes necessary are lines 2100 and 19900 (but DO NOT change line 3000). Change the "O" in OPEN "O", 1,B1 to "E" like this: OPEN "E", 1,B1.

This is the only change I had to make for NEWDOS and my program runs fine. The Poke statements in lines 2114 and 20000 appear harmless and seem to be flag storage locations to accomodate the repeated RUN 110 statements. This is the maybe, depending on whether or not you have NEWDOS.

Here are the minor changes:

The first question asked by the program is which disk drive is to be used. Well, I've only got one drive so I use the same one every time. Again two attacks on a problem. One REM or remove the line asking the question and set D\$ equal to the drive you will always be using. Like this:

REM or remove line 60 and add line 65.

65 D\$="O" (or 1,2, or 3)

If you will usually, but not always be using the same drive try this change. Do not REM or remove line 60 and add:

57 D\$=""

65 IF D\$="" THEN D\$="O" (or 1,2, or 3)

This sets D\$ to null. If you just press ENTER (the null string) D\$ is then set to the drive you have preselected.

If you REM or remove the question regarding which disk drive to be used (line 60), then also remove the CLS at the beginning of line 70.

If you're a real stickler toss this line in:

75 PRINT:PRINT "THE DRIVE TO BE USED IS --->";D\$

These next changes may seem petty, but here they are. I changed the stars in line 1660.

from: 1660 PRINT STRING\$(63,42) : RUN 110

to: 1660 PRINT STRING\$(63,61) : RUN 110

because they started to hurt my eyes. I also pulled line 55 since it was no longer true.

Here are the major modifications. They aren't really modifications but actually just additions.

The first one displays the 8 words that make up the file names. I kept forgetting the file names and would have to BREAK and LIST line 10000 to remember. This addition lets me see the file names while in the RUN mode. Add:

330 IF A\$=" " GOTO 7000

and add this block:

7000 PRINT:PRINT "FILE NAMES NOW IN USE----->";

7010 FOR MX=1 TO 4 : READ EX\$,EY\$

7020 PRINT EX\$,EY\$: PRINT TAB (32); : NEXT MX

7030 RUN 110

These next two extras require this line so let's start with it. Add:

150 C1=CHR\$(34):C2=CHR\$(95)+" - ":AL=STRING(15,45)

This addition is necessary because I have static ram chips in my brain...they always need refreshing. I'm too lazy to pull the book out everytime I forget a command or syntax arrangement. This table puts the information at my finger tips while running the program. Add:

320 IF A\$=" " GOTO 9000

And this block: (lines 9000-9130)

Comments by the author of VERSAFILE

Mike's article is very interesting and contains much useful information and some useful changes. I would like to caution users of VERSAFILE against using the program on NEWDOS, as the PEEKs and POKEs that Mike feels are harmless are actually direct manipulations of the DCB in TRSDOS (to simulate the OPEN "E" function) and can cause havoc with NEWDOS. So please use the program under TRSDOS 2.3 for which it was designed.

The other changes which Mike presents should cause no harm and may increase the ease of use for certain users. I am very pleased that VERSAFILE has been so well accepted and is now generating such constructive criticism and interest.

There are a few changes you can make to VERSAFILE to extend it's capacity and utility. As Mike stated in his article, the program does require a 32K machine, but a substantial amount of memory remains unused. I strongly suggest that the following changes be made to the program.

1. Change the "CLEAR 8000" statement in line 110 to "CLEAR 13000".

2. Change the "DEFSNG L,X,Y,Z" statement in line 120 to "DEFINT L,X,Y,Z".

3. Change the "DIM E(80),Y(80),A(25)" in line 120 to "DIM E(200),Y(200),A(25),K(25)".

The following changes are for Model II VERSAFILE:

1. Change the "DEFSNG L,X,Y,Z" in line 120 to "DEFINT L,X,Y,Z".

2. Add to the end of line 120 - ",K(30)".

3. This change is the most important - - change the last statement in line 420 from "RUN 110" to "GOTO 300".

I would like to thank 80-U.S. for the opportunity to comment on this article, and hope my comments prove useful to the users of VERSAFILE. There have been many reviews and articles on my products in many magazines, and this is the first time I have been solicited to comment and assist in the technical editing of one. This policy at 80-U.S. should be applauded, as it is a valiant effort to provide the fairest and technically most accurate information to readers.

Bill Schroeder.


```

9000 CLS:PRINTTAB(24)"C O M M A N D S"
      :PRINT AL;AL;AL;AL
9010 PRINT". - MUST END STATEMENTS TO BE FILED.
9020 PRINT"? - MUST END QUERIES TO BE SEARCHED BY.
9030 PRINT">>>> NOTE MANDATORY SPACE IN THE NEXT 4
      COMMANDS. <<<<<"
9040 PRINT"*"C2"GLOBAL SEARCH QUERIES.
      ("C1"*"CHR$(95)"?"C1" WILL LIST ALL FILES.)"
9050 PRINT"MK"C2"MULTIPLE KILL WITHIN ONE FILE. ENTER
      "C1"DONE"C1" TO END."
9060 PRINT"K"C2"SINGLE KILL FUNCTION. (BOTH MAY KILL
      MORE THAN ONE LINE!!)"
9070 PRINT"P"C2"LINE PRINTS RESULTS OF SEARCH."
9080 PRINT C1"FILE NAME"C1" - WITH "C1"?" WILL LIST
      ENTIRE FILE."
9090 PRINT"*# - LIST "C1"FILE BLOCKS"C1" WITH DISK
      LOCATION & "C1"MERGE FILE NAME"C1"."
9100 PRINT"# - WILL LIST FILE NAMES NOW IN USE."
9110 PRINT"@ - WILL INTERRUPT SEARCH OR LISTING AND
      RETURN."
9120 PRINT"<ENTER> - WILL DISPLAY THIS COMMAND TABLE.
9130 RUN 110

```

Do not add the line feeds and spaces as I have done. This is for ease of reading only. Whenever you simply press the enter key this command table is displayed. Figure 2 shows a screen dump of this command and gives further insight into VERSAFILE.

I've been saving the best addition for last. By writing small data line "programs" in BASIC and saving them with the ASCII File option (SAVE"COMPFILE/ASC",A) these "programs" can be merged to set up 45 new key words. (You realize that the /ASC is not mandatory, I use it to help identify files on my disks). Here are a few definitions before I go on.

"ASC merge file" - is the block of data lines (10000 as far as 12999, by user choice) that is being saved as a BASIC file with the "A" option.

"Disk name" - is the name of the disk on which the 8 files and ASC merge file is located.

"Merge file name" - is the name of the ASC merge file.

"File word block" - is the 8 (or 45) words themselves.

Here's how it works. The MERGE command gives you the ability to have one program that uses different data, be it string or numeric, without having to save the whole program on disk more than once. This is a powerful command that can be used on more applications than just the one described here.

With VERSAFILE we can go mild or wild, or in between. Let's start with mild. As you can see from figure 1., IS, ARE, WAS, etc. make rather a vague file word block. I think Radio Shack was just trying to get us started. They make excellent unnecessary words. Anyways, what you do is write a program in BASIC that is just data line 10000 naming 8 files. Like this:

```
10000 DATA " MEMORY "," MODE "," DISK ",.....etc.
```

Then save it with a merge file name using the ASCII (A)

```

8000 GOSUB 8800 : NT=0
8010 CLS : MT=0
8020 PRINT AL"PRESS "C1"M"C1" TO MERGE A FILE
      BLOCK "AL
8030 PRINT" DISK NAME      MERGE FILE NAME
      FILE NAMES IN BLOCK" : PRINT

```

option. Remember to have exactly 8 and use spaces. As in the fish and chips example be sure to rank them to your own preference. There is no need to save and merge the 37 unnecessary words if you are happy with them. I took Radio Shack's first 8 and filled them into line 10060, removing the XX's and deleting "S" and "I". But if you want, you can change the unnecessary words to fit each file word block's needs. Then you would be saving from 10000 to 10060.

Now we need a slick way to keep track of these ASC merge files, and make our swaps (MERGEs). This is the only hard part - where you have to do some work. You must create data lines in the program that will stay with the program to recall and use the ASC merge files. So for each new merge file you create, you have to add a data line between 12000 and 12999, and add line 13000. To work with this extra they must be in the format shown. (Even if you are changing all 45 words, only the first 8 should be entered in the 12000-12999 data lines).

```
12xxx DATA Disk name, Merge file name, "file name-1:
file name - 2: file name-3:.....file name-8"
```

Example:

```
12000 DATA VERSA SIDE 1, COMPFILE/ASC,
"MEMORY:MODE:
```

```
DISK:ADDRESS:FILES:BASIC:CALL:TRS-80"
```

Don't forget this line:

```
13000 DATA, END,END,END
```

(Again don't add the line feed and spaces as I have done.) Note also, the quotes and use of colons with the file word block.

Here are the additions to the program that make it all work. Add:

```
325 IF A$=" " * " GOTO 8000
```

And this block: (lines 8000-8910)

```

8040 READ E1,E2,E3
8050 IF E1="END" AND E2="END" AND E3="END" MD=1
      : GOTO 8140
8060 NT=TN+1 : FOR MZ=1 TO 7 : M1=INSTR(E3,":")
8070 E(MZ)=LEFT$(E3,M1-1) : M2=LEN(E3)
      : E3=RIGHT$(E3,M2-M1)
8080 NEXT MZ : E(8)=E3
8090 PRINT NT" "E1 TAB(20) E2 TAB(36);
8100 FOR MZ=1 TO 3 : FOR MX=1 TO 3 : M7=MX+(3*(MZ-1))
8110 PRINT E(M7); : IF (M7/3)<>INT(M7/3) PRINT " ";
8120 NEXT MX : PRINT : PRINT TAB(36);
8130 NEXT MZ : PRINT : MT=MT+1 : IF MT<3 GOTO 8040
8140 GOSUB 8900
8150 IF GZ$="M" GOTO 8500 ELSE IF MD=1 RUN 110
8160 IF GZ$<>"@" GOTO 8010 ELSE RUN 110
8500 PRINT @ 964,"WHICH FILE BLOCK, BY NUMBER
      AT LEFT -----> ";
8510 INPUT GZ$ : MU=VAL(GZ$) : IF MU=0 RUN 110
8520 IF MU>NT PRINT @ 949,"ERROR"; : GOTO 8500
8530 RESTORE : GOSUB 8800 : FOR MZ=1 TO MU
      : READ E1,E2,E3 : NEXT MZ
8540 CLS:PRINT @ 261,"INSERT DISK LABELED -----> "E1
8550 PRINT @ 324,"IN DISK DRIVE -----> "
      RIGHT$(DS,1)
8560 PRINT : GOSUB 8900 : PRINT : PRINT
8570 PRINT" TYPE "C1"RUN"C1" AND PRESS <ENTER>
      AFTER THE READY PROMPT."
8580 PRINT : MERGE E2
8800 FOR MZ=1 TO 45 : READ EZ : NEXT MZ : RETURN
8900 PRINT TAB(8)">>>>>>* PRESS ENTER TO CONTINUE *<<<<<<<";
8910 GZ$="" : GZ$=INKEY$ : IF GZ$="" GOTO 8910
      ELSE RETURN

```

Now whenever "*" is entered, the program will display 3 sets of information at a time, (see figure 3) the disk name, merge file name, and the file word block. Then you can either; see 3 more sets by pressing ENTER, merge a file word block (M), or return to the main program. If you choose to merge, you enter the number of the merge file name (see figure 4). The program asks for the disk containing the ASC merge file. After the appropriate disk is mounted, and <ENTER>, then type "RUN" and again <ENTER>. Presto! A new file word block.

You could also have a file word in more than one block giving you access to that file with more than just 7 other files. Sort of cross filing? Be careful though. If the various ASC merge files, with the common key word file, point to files on different disks, you could get all screwed up.

To go wild and crazy (and I said it first in 1973). Suppose when you did a merge you also changed the data lines 12000 to 12999? As if 999 lines isn't enough, or they take too long to get through. By swapping them your "pointers" can point just about endlessly. Want to go crazier? Bury a BASIC program in one of the 12000-12999 lines and pad it with enough data to fill the line to the right size. When you merge that "file word block"....(heh heh heh) and then type RUN? You got it, you're now in an entirely different program with no more VERSAFIL (unless you merge above it), crazy and talk about lazy. As you see, the combinations are endless, which makes computers crazy fun anyway.

Also, if you don't like my control keys, make your own. Change the characters in lines 320-330, change the character in line 1660 to whatever you want, I don't care. Just have a good time with it and let me know if you figure out how to get the damn computer to RUN after MERGE (is there RUN after MERGE?).

This late breaking bulletin just came in: (I just got my printer back from the shop). There is a weak spot in the output to line printer and only enough space to bring you the fixes.

The author has a subroutine to avoid chopping words during a line feed. This works fine on the output to screen, but to line printer, not only no chopped words, no line feed at all!! A real drag if you have long entries and narrow paper.

Change these two lines from:

```

1654 ** LINE PRINT IF IT HAS BEEN REQUESTED
1655 IF P=1 THEN LPRINT" ":LPRINT P$:FOR L=1 TO Z
:IF Y(L)=Y1 THEN LPRINT" "+ E(L):NEXT: ELSE NEXT
1654 IF P=1 THEN FOR L=1 TO Z
      : E(L)=RIGHT$(E(L),LEN(E(L))-4): NEXT L
1655 IF P=1 THEN FOR L=1 TO Z: IF Y(L)=Y1 GOSUB
4500:NEXT: ELSE NEXT

```

Then, take the two lines 4000 and 4010 and make two lines just like the numbered 4500 and 4510. Change the PRINTs to LPRINTs, make sure the internal GOTO is to 4510 (NOT 4010), and pull a few spaces in 4510 to make up for the added "L"s. This line is very close to the maximum of 255 characters so this is important.

```

10000 DATA " IS "," ARE "," WAS "," WERE "," WILL "," HAS ","
        " CAN "," THE "
10010 DATA " AT "," ALL "," IT "," A "," THEN "," OF "
10020 DATA " WHAT "," WHEN "," WHO "," HOW "," WHY "," WHERE "
10030 DATA " PLEASE "," IF "," NOW "," THEY "," TELL "," ME "
10040 DATA " AND "," FOR "," 'S "," MUCH "," COST "," S' "
10050 DATA " IN "," AS "," AN "," ABOUT "," THERE "," S "
10060 DATA " I "," XX "," XX "," XX "," XX "," XX "," XX "

```

Figure 1

C O M M A N D S

```

-----
A PERIOD MUST END STATEMENTS TO BE FILED.
A QUESTION MARK MUST END QUERIES TO BE SEARCHED.
*(SPACE) GLOBAL SEARCH QUERIES.
*(SPACE)? WILL LIST ALL FILES.
MK(SPACE) MULTIPLE KILL WITHIN ONE FILE.
K(SPACE) SINGLE KILL FUNCTION.
P(SPACE) LINE PRINT RESULTS OF SEARCH.
"FILE NAME" WITH "?" WILL LIST ENTIRE FILE
*# LISTS FILE BLOCKS WITH DISK LOCATION & 'MERGE FILE NAME'.
# WILL LIST FILE NAMES NOW IN USE.
@ WILL INTERRUPT SEARCH OR LISTING AND RETURN.
(ENTER) WILL DISPLAY THIS COMMAND TABLE.

```

Figure 2

```

----- PRESS M TO MERGE A FILE BLOCK -----
DISK NAME      MERGE FILE NAME      FILE NAMES IN BLOCK

VERSA SIDE 1  REGFILES/ASC          IS ARE WAS WERE WILL
                                         HAS CAN THE
VERSA SIDE 2  COMPFIL/ASC           MEMORY MODE DISK
                                         ADDRESS FILES BASIC
                                         CALL TRS:80
VERSA SIDE 3  MAINTAIN/ASC          HOUSE COMPROOM GARDEN
                                         GARAGE AQUA VW
                                         PLYMOUTH TOOLS
>>>>>>> * PRESS ENTER TO CONTINUE * <<<<<<<<

```

Figure 3

```

INSERT DISK LABELED -----> VERSA SIDE 1
IN DISK DRIVE -----> 0

```

```

>>>>>> * PRESS ENTER TO CONTINUE * <<<<<<

```

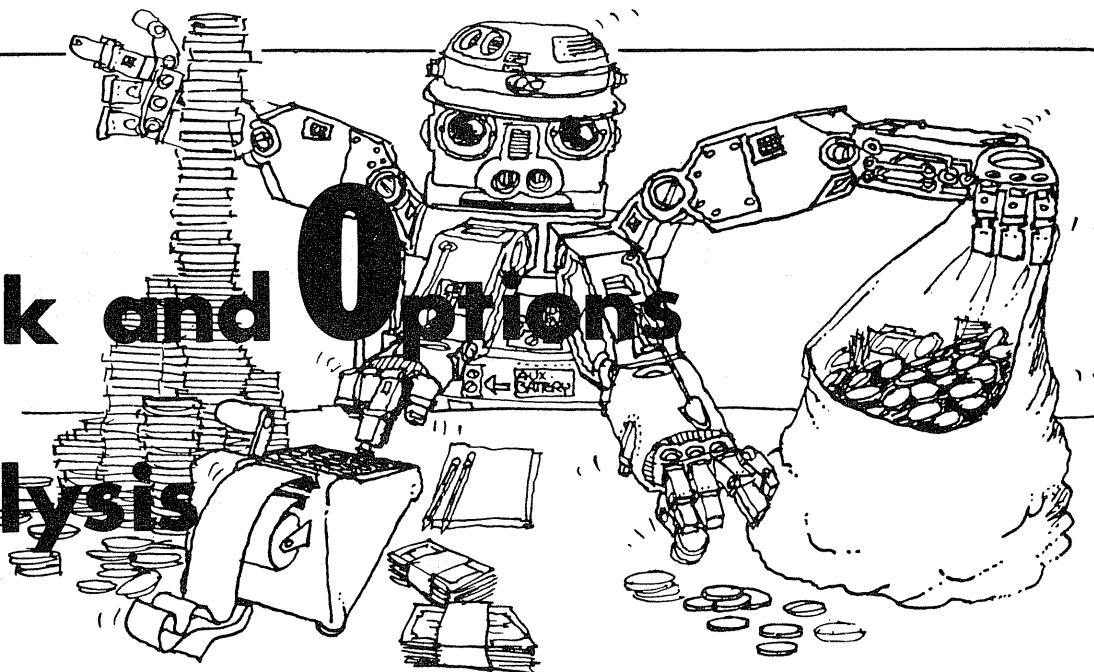
```

TYPE 'RUN' AND PRESS (ENTER) AFTER THE READY PROMPT.

```

Figure 4

Stock and Options Analysis



Keep the data you need to make timely investment decisions at your fingertips with this incredibly powerful investment tool. Considerable effort has gone into methods of tilting the odds in the investment game. Out of this has come the discovery that the strategy of hedging listed options against common stocks can tilt the odds drastically. In fact, it can be *more conservative* and *more consistently profitable* than the simple buying and selling of stock.

The four programs in this package are designed to be used in the real world, and include the effects of commissions, margin interest and dividends, where applicable. Possible investment attitudes, the listed option markets, puts and calls and option strategies are covered in extensive documentation.

The **Option** program presents important indices of both opening and closing call option transactions. The manual includes sample runs illustrating combination strategies with covered and uncovered calls, and covered and uncovered straddles receive detailed treatment.

The **Opgraph** program presents a graph or a table, as the user chooses, of profit from any combination of six basic positions: long or short a stock, long or short a call and long or short a

put. Sample runs are presented which cover hedging with calls, out-of-the-money hedges and in-the-money hedges.

Newprem enables the user to predict the future premiums of an option at whatever time and future stock price the user selects. This method requires the establishment of a data base of historical option premiums in whatever detail the user desires.

Finally, **Portval** enables the user to determine on an item by item basis, the cost, current value per share, total current value and capital gain of a portfolio consisting of long and short stock, and long and short option positions. This program assists the user in keeping a readily available and easily updatable record of his portfolio and, at the same time, assists him in measuring his progress towards financial success.

In order for an investor to continually improve his performance it is necessary for him to refer to past performance; this requires useful records. Finally, he should constantly be evaluating his performances to assure himself he is playing the right game.

The **Stock and Options Trading Analysis** package is available for the 16K TRS-80 Level II on cassette (CS-3306) and disk (CS-3801) for \$99.95. Creative Computing Software should be available at your local computer store. If your favorite retailer does not stock the software you need, have him call our retail marketing department at the number below. Or you can order directly from Creative Computing Software, Dept USLG; P.O. Box 789-M, Morristown, NJ 07960. Visa, MasterCard, or American Express are also welcome. For faster service, call in your bank card order toll free to 800-631-8112. In NJ call 201-540-0445.

TRS-80 Professional Software

**Creative
Computing
Software**

A Pocket COMPUTER

Application



Bill Vick
Plano, TX

Who can't remember the first time their computer came to life and the wonder we all felt thinking that finally we had the power of a main frame monster sitting there? For a few hundred dollars Mr Tandy gave us the equivalent power of a machine that would have cost many thousands a few short years ago. It was a true computer that was not much bigger than a bread box. In many ways, it touched all of our lives.

Well, he's done it again. The new pocket computer from our friends in Fort Worth (via Sharp and Japan) is now available for less than \$250.00, and it will knock your socks off. Here's a Basic interpreter computer you can slip into your pocket or brief case. It's totally self contained and totally portable.

First, it is a programmable computer with a subset of tiny Basic, in addition to having 15 powerful calculator-like functions. Most of the 62 commands or functions available work just like its big brother using Level I Basic. The Basic is a 7K Basic, supported by a 4K operating system. You can abbreviate commands, like entering M. instead of MEM to display the remaining memory. The new commands are PAUSE, which display an instruction for close to a second. USING will format numerical data. BEEP is a programmable sound generator. AREAD reads the contents of the display into a variable. DEBUG single steps through a program. CHAIN will load a program from cassette tape and execute it.

The display is handled with a 24 digit alphanumeric dot matrix liquid crystal. It will scroll and 80 characters per line can be scrolled and displayed. It features ten digit numeric accuracy, with exponential notation to plus or minus the 99th power of ten.

The pocket computer has a capacity of 1,424 steps and has 26 data elements, a 48 step reservable memory for storing frequently used functions, and 1.9K of RAM. It can store a program up to 999 lines long. With a 57 key QWERTY type keyboard, it's compact, but don't let its small size fool you. Eighteen of the keys are definable for functions or other uses. The 4 modes of operation are *RUN*, *DEFine*, *PROgram* and *RESERVE*. The run command allows you to perform program or manual calculations. PRO is the program writing mode. RESERVE

is a reserve or abbreviated format writing mode. DEF performs defined program calculations. Two templates are provided which allow you to identify the functional operation assigned to the reserve keys or defined programs assigned to the definition key.

Because of the permanent memory that retains data in programs with the computer turned off, frequently used programs or data are always recallable. I even have used the reserve keys to keep track of phone numbers. It will hold up to 7 phone numbers, or 8 dates or any message up to 48 characters, in addition to functions like LOG, ABS, SGN, etc., for instant recall and use.

The pocket computer also has 6 error codes, a complete and very well done 122 page manual and optional expansion interface allowing you to save programs or data onto a cassette recorder.

Programming the pocket computer is a breeze. Since I am in sales, I am already dealing with the percentage this years sales are ahead or behind last years, I am also asking myself if 9 orders gives me \$2,713, how much will 13 orders give me. Both simple applications; but in dealing with hundreds of instances of this in any day, a small savings of time becomes very worthwhile. I've included a listing of this simple application to demonstrate how the new commands work. With the pocket computer in the DEF mode you enter RUN. From there on it is self prompting and will lead you to a menu driven routine to call program label A or program label B.

The program has two routines tied together under the define mode. Program A determines the percent of one period versus another. Program B is used in a crude forecast mode. I use a slightly slicker linear regression program for serious forecasting, also on this computer.

The following is an overview of the commands by line number in the program which follows:

Line 1 - Clears the system of all data. Zeros variables. A menu prompts for a shift A to run the percent program or a shift B to jump to the forecast program.

Lines 2-3 - Label A is for the beginning of the percent program. The PAUSE command flashes the previous descriptions and the author's name for .85 seconds on the

(Continued on page 50)

YOUR MODEL II CAN HAVE SNAPP!



SNAPP II EXTENDED BASIC

A family of enhancements to the Model II BASIC interpreter. Part of the package originated with the best of APPARAT, INC.'s thoughts in implementing NEWDOS BASIC. The system is written entirely in machine language for SUPER FAST execution. The extensions are fully integrated into Model II BASIC, and require NO user memory, and NO user disk space. The package is made up of the following six modules, each of which may be purchased separately:

XBASIC—Six single key stroke commands to list the first, last, previous, next, or current program line, or to edit the current line. Includes quick way to recover BASIC program following a NEW or system or accidental re-boot. Ten single character abbreviations for frequently used commands: AUTO, CLS, DELETE, EDIT, KILL, LIST, MERGE, NEW, LIST, and SYSTEM. \$40

XREF—A powerful cross-reference facility with output to display and/or printer. Trace a variable through the code. Determine easily if a variable is in use. \$40

XDUMP—Permits the programmer to display and/or print the value of any or all program variables. Identifies the variable type for all variables. Each element of any array is listed separately. \$40

XRENUM—An enhanced program line renumbering facility which allows specification of an upper limit of the block of lines to be renumbered, supports relocation of renumbered blocks of code, and supports duplication of blocks of code. \$40

XFIND—A cross reference facility for key words and character strings, also includes global replacement of keywords. \$40

XCOMPRESS—Compress your BASIC programs to an absolute minimum. Removes extraneous information; merge lines; even deletes statements which could not be executed. Typically saves 30-40% space even for programs with our REM statements! Also results in 7-10% improvement in execution speed. \$40

ENTIRE PACKAGE ONLY \$200.00



DOSFIX

A collection of patches to TRSDOS and BASIC to enhance their usability and function includes our well-known BREAK7E patches to keep the break key from being used accidentally. FREE WITH ANY MODEL II SOFTWARE PACKAGE.



CONVERT

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



SBASIC — Model I and Model II

Program in a high-level, full structured BASIC! The BEST of the BASIC pre-processors. PERFORM named subroutines. CONDITIONAL case structures. WHILE loops. UNTIL loops. And much more. Forget about line numbers. Model II version is compiled, and SUPER FAST. From Ultimate Computer Systems, Model I \$50 Model II \$75



XPRINT

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



BPRINT

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



PPRINT

Updates to The Electric Pencil to support true proportionally spaced printing with the Shacks new letter quality printer, the Daisy Wheel II. Produces copy which looks as if it had been typeset. \$100



ITOLL

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



EXTENDED BUILT IN FUNCTIONS

Now you can give your TRS-80 all the functions you wished BASIC had given you in the first place. These verbs will give you programming abilities that make you look good. Adds the following function verbs: SORT, PEEK, PEEKW, POKE, POKEW, ETIM\$, and XTIM\$. \$50



DIAL

USR 330D Auto Answer/Auto Dial. Direct Connect Modem. 300 baud, originates/answers 103J compatible. When used in conjunction with our DIAL, software is capable of complete origination of communications with remote locations without operator intervention. Special combination price, modem and software. \$430
Software only. \$50



MASTER / SLAVE

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



SPOOLER—Model I and Model II

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

Serial printer support, and DISK SPOOLING support is particularly recommended for word processing applications. \$100
SERIAL PRINTER OPTION \$50
DISK SPOOLING OPTION \$50



HOSTII/TERMII

Allows remote control of a Model II from another Model II, or any ASCII terminal. Our Host system, unlike the one supplied with TRSDOS 2.0, supports accurate screen positioning on the Term station. Without this feature, formatted displays appear on the terminal looking like randomly placed garbage. Requires NO user memory! This system is designed to provide software support to our customer locations without ever leaving the office. \$50



TERMS OF SALE:

Credit card customers, add 3% C.O.D. customers add \$3. Ohio residents add 4 1/2% sales tax. Shipments normally made the same day we receive your order.



OUR GUARANTEE:

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

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

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

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



8160 Corporate Park Dr.
Cincinnati, Ohio 45242

Ohio residents call collect

(513) 891-4496

Call Toll Free

1 - 800 - 543-4628

All products now available to run with TRSDOS 2.0.

Most products will soon be available for the Model III. CALL FOR DETAILS!



TRS-80 BASIC PLUS

ZBASIC, SIMUTEK'S BASIC COMPILER

The following **BASIC PROGRAM**, written on the TRS-80, was compiled using MICROSOFT'S BASIC COMPILER and SIMUTEK'S BASIC COMPILER. *We feel the results speak for themselves!*

```
10 ' SPEED TEST
    SIMUTEK ZBASIC COMPILER VS. MICROSOFT COMPILER
15 CLS:PRINT@, "HIT A KEY WHEN READY TO START TEST";
20 I$=INKEY$: IF I$="" THEN 20 ELSE FOR Z=1 TO 10:
    FOR X=15360 TO 16383: POKE X, 191: PRINT PEEK(X): NEXT X
30 FOR X=0 TO 127: FOR Y=0 TO 47: SET(X, Y): NEXT Y, X
    :FOR X=127 TO 0 STEP -1: FOR Y=47 TO 0 STEP -1: RESET(X, Y)
    :NEXT Y, X: FOR X=1 TO 1000: GOSUB 1000: NEXT X, Z
40 CLS:PRINT "FINISHED WITH PROGRAM TEST": STOP
1000 RETURN
```

BASIC PROGRAM SIZE: 329 BYTES
PROGRAM RUN: 22 Minutes, 37 Seconds

Compilers:	Microsoft	Simutek
Compiled Size:	10057 Bytes	1228 Bytes
Compile Time:	14 Minutes	0.75 Seconds
Program Run:	17 Min. 04 Sec.	1 Min. 46 Sec.
System Req.:	48K 1 Disk	16K LV II or 32-48K Disk
Price:	\$195.00	Tape \$99.00, Disk \$129.00

ZBASIC is an "Interactive Compiler". This means it is resident while you write your basic programs. You may compile your program and run it or save it, without destroying your resident basic program! In fact, jumping back and forth between your compiled program and your basic program is one of its best features!

Simutek's compiler allows saving your "compiled" programs to tape or disk. Programs may then be loaded by use of the system command for tape, or as a /CMD file from DOS. This makes it extremely hard for people to "pirate" your programs.

Best of all, Simutek does not charge royalties on programs you sell that are compiled with ZBASIC! (Microsoft charges 10% or \$200 a year!)

Why use a complicated "Assembler" to write machine language programs when you can write them in ZBASIC?

Some of the basic commands supported by ZBASIC:

FOR	NEXT	STEP	IF	THEN	ELSE	PEEK	ON GOTO
SET	RESET	POINT	CHR\$	RANDOM	RND ()	POKE	ON GOSUB
DATA	READ	RESTORE	END	GOTO	GOSUB	CLS	
INPUT	INKEY\$	LET	STOP	OUT	INP	RETURN	
PRINT	LPRINT	PRINT@	USR	SGN	INT	ABS	
SQR	LEN	ASC	VAL				
INT. MATH +, -, *, /, AND, OR, SQR							

Model I TRS-80 (or PMC-80) Only
ZBASIC Tape Version: 16K Level II TRS-80 \$99.00
ZBASIC Disk Version: 32 or 48K 1 Disk Sys. \$129.00
ZBASIC Manual Only: \$25.00

Credit Card or C.O.D. Call **Toll Free: (800) 528-1149**
or send check or money order to

SIMUTEK
COMPUTER PRODUCTS™

P.O. Box 13687 Tucson, AZ 85732 (602) 886-5880
(C.O.D. Available \$3.00 Extra)
TRS-80 is a TM of Radio Shack, a Tandy Corp.

display and then jumps to the next line.

Lines 4-5 - Prompts and assigns a keyboard response into the variables L, T.

Line 6 - Percent algorithm.

Line 7 - Subroutine for print formatting of data with a print using command.

Line 8 - Prints the percentage differences.

Line 9 - Turns off the print using command.

Lines 10-12 - Cumulative totals and cumulative difference.

Line 13 - Gives audio BEEP to signify end of routine.

Line 14 - Turns on the print using statement.

Line 15 - Prints total cumulative difference.

Line 16 - Turns off print using statement.

Line 17 - Runs program again by jumping to line 4.

Each program step was entered in the abbreviated format to save memory. It took less than 5 minutes to write the program, and the time savings is many hours over the course of a months time.

Personally, I am very impressed with the new pocket computer. It needs printing capabilities which I'm sure are just around the corner. In fact I saw a recent article in an electronics magazine about Sharp's new printing calculator which looks suspiciously like our pocket computer with a thermal printer added. I haven't seen any of the Tandy program tapes, however, there is a whole world of tiny Basic programs out there that are just waiting to be put into the pocket. It has limitations, but every day I'm finding new uses and applications for it. It is much more than a toy, and I would recommend it to anyone needing a small, portable computer.

```
1 :CLEAR:PRINT "<A>=%<B>=FORECAST"
2 : "A":PAUSE "PERCENT"
3 :PAUSE "BY BILL VICK 8/29/80"
4 :INPUT "LAST PERIOD";L:A=A+L
5 :INPUT "THIS PERIOD";T:H=H+T
6 :P=((T-L)/T)
7 :GOSUB 500
8 :PRINT "5 DIFF=";P
9 :GOSUB 500
10 :PRINT "TOTAL LAST=";A
11 :PRINT "TOTAL THIS=";H
12 :E=((H-A)/H)
13 :BEEP 1
14 :GOSUB 400
15 :PRINT "TOTAL DIF=";E
16 :GOSUB 500
17 :GOTO 4
18 : "B":REM FORECAST PROGRAM
19 :PAUSE "AVERAGE FORECAST"
20 :INPUT "SALES",S
21 :INPUT "ORDERS",O
22 :V=S/O
23 :GOSUB 400
24 :INPUT "PLANNED ORDERS",P
25 :F=P*V
26 :PRINT "FORECAST=";F
27 :GOSUB 500:GOTO 20
400 :USING "####.#":RETURN
500 :USING:RETURN
```


LOBO DRIVES INTERNATIONAL ANNOUNCES - L D O S - THE NEW GENERATION OF OPERATING SYSTEMS FOR THE TRS-80 COMPUTERS. NOT ANOTHER CONVERSION OF TRSDOS, BUT AN ENTIRELY NEW SYSTEM!

- LDOS is an operating system that supports 5", 8", single or DOUBLE SIDED, single or DOUBLE DENSITY, up to 80 TRACKS and even HARD DISK (with ZAPPING THIS and PATCHING THAT).
- LDOS can use SERIAL and PARALLEL printers or both, and fully supports I/O with any RS-232 device.
- All I/O is fully "DEVICE INDEPENDENT", allowing full LINKING and ROUTING of all devices!!
- A true SYMBIONIC SPOOLER, using both RAM and DISK, is provided.
- Assign any alpha key to produce a designated string of characters with KEY STROKE MULTIPLY.
- AUTO REPEAT on all keys, just hold the key down.
- Fully buffered TYPE AHEAD (you can't out type PENCIL).
- Functional PURGE to easily clean up cluttered disks.
- WILDCARDS can be used with DIRECTORIES BACKUP, APPEND and COPY.
- BACKUP by CLASS or only files that have been MODIFIED since they were last backed up.
- COPY and APPEND are up to 300% faster.
- Built in LOWER CASE DRIVER.
- BLINKING CURSOR, (you pick the character).
- Set printer PAGE LENGTH, LINES per page and INDENT on wrap around.
- Preprocessing of I/O is possible with the unique "FILTER".
- JKL function to PRINT THE SCREEN with GRAPHICS SUPPORT.
- EXTENDED DEBUGGER with DISK EDITOR.
- BUILD allows creation of ASCII files which contain instructions for job execution by LDOS.
- DO is the method by which a file created by BUILD is executed.
- JOB LOG may be turned on to keep a record of all of LDOS's activity.
- Extended PATCH utility, for easy maintenance of programs, data and system software.
- Full COMMUNICATIONS functions, including UPLOAD and DOWNLOAD to systems like MICRONET.
- Compiled JOB CONTROL LANGUAGE with conditionals, variables and user interaction.
- Full SYSGEN capability to permanently store your hardware environment and system configurations.
- BASIC enhancements (REFERENCE, RENUM, VARIABLE LENGTH RECORDS).
- LDOS is upward COMPATIBLE with TRSDOS 2.3!!
- Totally compatible with RADIO SHACK'S INTERFACE and DRIVES.
- Fully supports the LOBO INTERFACE and LOBO DRIVES for over 5 megabytes of on-line floppy storage!!! or over 10 megabytes with hard disk!!!
- Plus there are many, many other features that other operating systems are just dreaming about. They are here now, in LDOS.

To provide the TRS-80 user with the ABSOLUTE BEST in sophisticated operating systems, LOBO has assembled the top people in the industry to stand behind LDOS and it's users and has provided the user with an extensive manual. Every LDOS user has the support of a full time LDOS support division, with a toll-free 800 number to aid the user in solving any problems he may encounter.

LOBO has spared no effort or expense in bringing the TRS-80 OWNER the BEST OPERATING SYSTEM and SUPPORT ever!!!!



FEATURES..... THE BEST
DOCUMENTATION..... THE BEST
PRICE..... THE BEST
SUPPORT..... THE BEST
COMPATIBILITY..... THE BEST

Dealer
Inquiries
Invited

"Why pay more for less when you can buy the best".

Contact your LOCAL DEALER or the DISTRIBUTOR nearest you for a free brochure describing LDOS and all it's outstanding features.

.....WEST.....EAST.....MIDWEST.....

LOBO Drives Intl.
354 S. Fairview Ave.
Goleta, CA 93117
805/683-1576

MISOSYS
5904 Edgehill Dr.
Alexandria, VA. 22303
703/960-2998

Galactic Software Ltd.
11520 N. Port Washington Rd.
Mequon, WI. 53092
414/241-8030

System/Command

A 32-line Screen with Controlled Scrolling For the Model I TRS-80

The 13th in a series.

Phil Pilgrim

Discovery Bay Software Co.
Port Townsend, Washington

The problem with "soft copy" (i.e., screen data) is that once it's gone, it's gone. And the main culprit is often that ubiquitous scrolling mechanism that pushes your valuable results off the top of the screen like so much chaff. "No respect," you mutter as you stick another INPUT "HIT ENTER TO CONTINUE";K\$ in your program. "There must be a better way!" Well, the program presented here might be what you are looking for. It not only performs the HIT ENTER, etc. automatically, but also remembers the last 16 lines scrolled off the top of the screen and lets you review them at your leisure. It's like having a 32-line screen - almost.

This program links into both the display and keyboard calling sequences. The keyboard part, KEYLNK, looks for a SHIFT-BREAK or a SHIFT-ENTER. Finding the former, it inhibits any further scrolling until you command otherwise. With the latter, it permits scrolling in the usual uncontrolled basis. The display portion, DSPLY, checks for an impending scrolling situation, and if one exists, checks whether the next line can be displayed. Given permission to scroll, it saves the top line of the screen in

BUFFER and calls the regular display routine in ROM. When commanded to review material scrolled off the screen, it backs up one line (i.e., does a reverse scroll). When DSPLY is awaiting permission to scroll, it displays a down-arrow in the lower righthand corner of the screen.

Permission to scroll is controlled by the value of MODE. If MODE=255 (set by SHIFT-ENTER) then scrolling continues uncontrolled. If MODE=0 (set by SHIFT-SPACE) then scrolling stops, the down arrow appears, and DSPLY waits for keyboard input to tell it what to do. Hitting the SPACE bar or the down-arrow key at this point sets MODE to 1, permitting the screen to scroll one line. Hitting ENTER sets MODE to 16, permitting the screen to scroll 16 lines. Hitting the up-arrow key doesn't affect MODE but performs the special reverse scroll function which lets the user review text scrolled off the top.

When reverse scrolling is performed, a counter (COUNT) is incremented to keep track of how many lines have been reviewed. When a maximum of 16 is

reached, further reverse scrolling is inhibited. This is done in the section labelled BACKUP. Assuming it's okay to reverse-scroll, one line is pulled out of the BUFFER and stored temporarily in TBUF. Next, that line (in BUFFER) is replaced with the bottom line on the screen. Following that, the screen is scrolled down, and the line saved in TBUF is restored on the top of the screen. When one or more bottom lines are saved thus, the program restores them during subsequent forward scrolls before new material is displayed.

To use the program, key it as shown into EDTASM, substituting memory size labels and either JP BASIC or JP DOS as appropriate. Assemble it, and make an object file or tape. Load the object code and execute the START block to link it into the display and keyboard calling sequences. Now, whenever you want to gain control of the scroll, just hit SHIFT-SPACE. At the next scrolling opportunity, the down-arrow will appear and the special functions mentioned will be available. To relinquish control, just hit SHIFT-ENTER. Now you've got respect! ●

0ECC	00100 BASIC	EQU	0ECC	;ENTRY ADDRESS FOR BASIC
402D	00110 DOS	EQU	402DH	;ENTRY ADDRESS FOR DOS
7A68	00120 MEM16K	EQU	7A68H	;MEM SZ=31336
BA68	00130 MEM32K	EQU	0BA68H	;MEM SZ=47720
FA68	00140 MEM48K	EQU	0FA68H	;MEM SZ=64104
FA68	00150	ORG	MEM48K	
FA68 218AFA	00160 START	LD	HL, DSPLY	;LINK IN DSPLY
FA6B 221E40	00170	LD	(401EH), HL	;
FA6E 2A1640	00180	LD	HL, (4016H)	;LINK IN KEYLNK
FA71 22A0FB	00190	LD	(KEYLNK+1), HL	;
FA74 219FFB	00200	LD	HL, KEYLNK	;

FA77	221640	00210	LD	(4016H), HL	;	
FA7A	21C0FB	00220	LD	HL, BUFFER	; FILL BUFFER W/ BLANKS	
FA7D	3620	00230	LD	(HL), ' '	;	
FA7F	54	00240	LD	D, H	;	
FA80	5D	00250	LD	E, L	;	
FA81	13	00260	INC	DE	;	
FA82	01FF03	00270	LD	BC, 1023	;	
FA85	EDB0	00280	LDIR		;	
FA87	C3CC06	00290	JP	BASIC	; RETURN TO BASIC	
FA8A	DA5804	00300	JP	C, 0458H	; IF CARRY, JUST INQUIRING	
FA8D	DD7E04	00310	LD	A, (IX+4)	; MSB OF CURSOR POSITION	
FA90	FE3F	00320	CP	3FH	; LAST QUADRANT?	
FA92	202E	00330	JR	NZ, DOIT	; NO: OKAY TO DISPLAY	
FA94	79	00340	LD	A, C	; GET CHARACTER	
FA95	FE20	00350	CP	20H	; DISPLAYABLE?	
FA97	3808	00360	JR	C, CTLCHR	; NO: CONTROL CHARACTER	
FA99	DD7E03	00370	LD	A, (IX+3)	; LSB OF CURSOR POSITION	
FA9C	3C	00380	INC	A	; LAST POSITION ON SCREEN?	
FA9D	2811	00390	JR	Z, SCROLL	; YES: A SCROLL CONDITION	
FA9F	1821	00400	JR	DOIT	; NO: OKAY TO DISPLAY	
FAA1	FE0A	00410	CTLCHR	CP	0AH	; A CARRIAGE RETURN?
FAA3	381D	00420	JR	C, DOIT	; NO: OKAY TO SNED	
FAA5	FE0E	00430	CP	0EH	; MAYBE: CHECK OTHER SIDE	
FAA7	3019	00440	JR	NC, DOIT	; NO: OKAY TO SEND	
FAA9	DD7E03	00450	LD	A, (IX+3)	; LSB OF CURSOR POSITION	
FAAC	FEC0	00460	CP	0C0H	; LAST LINE ON SCREEN?	
FAAE	3812	00470	JR	C, DOIT	; NO: DO CR	
FAB0	3ABCFB	00480	SCROLL	LD	A, (MODE)	; GET MODE BYTE
FAB3	3C	00490	INC	A	; UNCONTROLLED?	
FAB4	2807	00500	JR	Z, SCOK	; YES: OKAY TO SCROLL	
FAB6	3D	00510	DEC	A	; ZERO?	
FAB7	280D	00520	JR	Z, GETKEY	; YES: MUST ASK PERMISSION	
FAB9	3D	00530	DEC	A	; ONE MORE LINE	
FABA	32BCFB	00540	LD	(MODE), A	; SAVE IT AWAY	
FABD	C5	00550	SCOK	PUSH	BC	; SAVE CHARACTER
FABE	CD85FB	00560	CALL	TOPSAV	; SAVE TOP LINE OF SCREEN	
FAC1	C1	00570	POP	BC	; RESTORE CHARACTER	
FAC2	B7	00580	DOIT	OR	A	; RESET CARRY
FAC3	C35804	00590	JP	0458H	; JP TO DISPLAY ROUTINE	
FAC6	CD07FB	00600	GETKEY	CALL	KEYBD	; GET PERMISSION TO SCROLL
FAC9	3ABDFB	00610	CTEST	LD	A, (COUNT)	; BACKED UP ANY?
FACC	B7	00620	OR	A	;	
FACD	28E1	00630	JR	Z, SCROLL	; NO: OKAY	
FACF	3D	00640	DEC	A	; YES: DECREMENT COUNT	
FAD0	32BDFB	00650	LD	(COUNT), A	; AND SAVE	
FAD3	C5	00660	PUSH	BC	; SAVE CHAR	
FAD4	2ABEFB	00670	LD	HL, (SPTR)	; GET SAVED BOTTOM LINE	
FAD7	11C0FF	00680	LD	DE, TBUF	; INTO TBUF	
FADA	014000	00690	LD	BC, 64	; ONE LINE	
FADD	EDB0	00700	LDIR		; MOVE IT	
FADF	CD85FB	00710	CALL	TOPSAV	; SAVE TOP LINE IN BUFFER	
FAE2	11003C	00720	LD	DE, 3C00H	; TOP OF SCREEN	
FAE5	21403C	00730	LD	HL, 3C40H	; NEXT LINE	
FAE8	01C003	00740	LD	BC, 960	; ALL BUT ONE LINE	
FAEB	EDB0	00750	LDIR		; SCROLL UP	
FAED	11C03F	00760	LD	DE, 3FC0H	; BOTTOM LINE	
FAF0	21C0FF	00770	LD	HL, TBUF	; SAVED BOTTOM LINE	
FAF3	0E40	00780	LD	C, 64	; B ALREADY = 0	
FAF5	EDB0	00790	LDIR		; RESTORE BOTTOM LINE	
FAF7	C1	00800	POP	BC	; RESTORE CHAR	
FAF8	3ABCFB	00810	LD	A, (MODE)	; GET MODE BYTE	
FAFB	3C	00820	INC	A	; UNCONTROLLED?	
FAFC	28CB	00830	JR	Z, CTEST	; YES: CHECK FOR ANOTHER	
FAFE	3D	00840	DEC	A	; DECREMENT IT	

FAFF 3D	00850	DEC	A	;AGAIN
FB00 32BCFB	00860	LD	(MODE),A	;AND SAVE
FB03 20C4	00870	JR	NZ,CTEST	;IF NON-ZERO, DO AGAIN
FB05 18BF	00880	JR	GETKEY	;ZERO: GET PERMISSION
FB07 21FF3F	00890 KEYBD	LD	HL,3FFFH	;LAST POS ON SCREEN
FB0A 46	00900	LD	B,(HL)	;GET CHAR THERE
FB0B 365C	00910	LD	(HL),92	;REPLACE W/ DOWN-ARROW
FB0D C5	00920	PUSH	BC	;SAVE CHARS
FB0E CD2B00	00930 CALKEY	CALL	002BH	;GET CHAR FROM KBD
FB11 0601	00940	LD	B,1	;POSSIBLE MODE 1
FB13 FE20	00950	CP	' '	;BLANK?
FB15 2816	00960	JR	Z,OUT	; YES: MODE 1
FB17 FE0A	00970	CP	10	;LINE FEED?
FB19 2812	00980	JR	Z,OUT	; YES: MODE 1
FB1B 0610	00990	LD	B,16	;POSSIBLE MODE 16
FB1D FE0D	01000	CP	0DH	;ENTER?
FB1F 280C	01010	JR	Z,OUT	; YES: MODE 16
FB21 FE5B	01020	CP	91	;UP-ARROW?
FB23 2812	01030	JR	Z,BACKUP	; YES: REVERSE SCROLL
FB25 3ABCFB	01040	LD	A,(MODE)	;MAYBE CHANGED IN KEYLNK
FB28 B7	01050	OR	A	;ZERO?
FB29 2006	01060	JR	NZ,OUT1	; NO: GET OUT
FB2B 18E1	01070	JR	CALKEY	; YES: TRY AGAIN
FB2D 78	01080 OUT	LD	A,B	;NEW MODE
FB2E 32BCFB	01090	LD	(MODE),A	; .
FB31 C1	01100 OUT1	POP	BC	;RETRIEVE CHARS
FB32 78	01110	LD	A,B	;RESTORE LRHC OF SCREEN
FB33 32FF3F	01120	LD	(3FFFH),A	; .
FB36 C9	01130	RET		;AND GET OUT
FB37 3ABDFB	01140 BACKUP	LD	A,(COUNT)	;HOW MANY LINES BACKED UP?
FB3A FE10	01150	CP	16	;16 YET?
FB3C 28D0	01160	JR	Z,CALKEY	; YES: CAN'T DO ANY MORE
FB3E 3C	01170	INC	A	;ONE MORE
FB3F 32BDFB	01180	LD	(COUNT),A	;SAVE IT
FB42 C1	01190	POP	BC	;RESTORE LRHC CHAR
FB43 78	01200	LD	A,B	; .
FB44 32FF3F	01210	LD	(3FFFH),A	; .
FB47 C5	01220	PUSH	BC	;BUT STILL SAVE C
FB48 11C0FF	01230	LD	DE,TBUF	;SAVE BOTTOM LINE IN TBUF
FB4B 21C03F	01240	LD	HL,3FC0H	; .
FB4E 014000	01250	LD	BC,64	;ONE LINE
FB51 EDB0	01260	LDIR		; .
FB53 11FF3F	01270	LD	DE,3FFFH	;SCROLL DOWN
FB56 21BF3F	01280	LD	HL,3FBFH	; .
FB59 01C003	01290	LD	BC,960	;15 LINES
FB5C EDB8	01300	LDDR		; .
FB5E 2ABEFB	01310	LD	HL,(SPTR)	;ADDR OF NXT SAVED LINE
FB61 0E40	01320	LD	C,64	;WANT PREVIOUS SAVED LINE
FB63 B7	01330	OR	A	;SO SUBTRACT 64
FB64 ED42	01340	SBC	HL,BC	; .
FB66 11C0FB	01350	LD	DE,BUFFER	;COULD BE TOO LOW
FB69 DF	01360	RST	18H	;IS IT? (COMPARE HL-DE)
FB6A 3003	01370	JR	NC,HLOK1	; NO: JUMP AROUND
FB6C 2180FF	01380	LD	HL,BUFFER+960	; YES: FIX ADDRESS
FB6F 22BEFB	01390 HLOK1	LD	(SPTR),HL	;SAVE NEW POINTER
FB72 11003C	01400	LD	DE,3C00H	;RESTORE TOP LINE
FB75 EDB0	01410	LDIR		; .
FB77 ED5BBEFB	01420	LD	DE,(SPTR)	;SAVE TBUF IN BUFFER
FB7B 21C0FF	01430	LD	HL,TBUF	; .
FB7E 0E40	01440	LD	C,64	;ONE LINE
FB80 EDB0	01450	LDIR		; .
FB82 C1	01460	POP	BC	;RESTORE CHAR IN C
FB83 1882	01470	JR	KEYBD	;BACK FOR ANOTHER COMMAND

Add POWER to your TRS-80TM

SOFTWARE by MiProg

XEDIT, a high powered compact disk based editor designed for the TRS-80TM Model I or II. Whether it is BASIC, ASSEMBLY, or FORTRAN, XEDIT is packed full of commands needed by programmers who are serious about their work. Here are just a few features:

- Edits most file formats
- Block text copy command
- Locate, Delete, and Change with windows
- Inserts and maps up to five input files
- Upper/lower case compatible
- Operates with or without line numbers
- Rapid access disk cache
- Recovers from most DOS errors
- Fast file entry point map
- Change text command for any number of occurrences
- DOS Directory and Kill commands
- Line printer paging with adjustable forms
- Sophisticated reprinting line editor, handles line feeds
- Disk BASIC, Disk EDTASM, and EDIT-80 format compatible
- Display status command, includes free memory, current pointer printer forms, number of input files, output filename and format.

XEDIT will handle files of any size up to 2.7 Megabytes or 10K lines in length. Comes complete with instructions covering operation, externals, and file formats.

Model I (32K single disk system)

Formatted diskette	\$44.95
Cassette tape.	\$39.95

Model II

Formatted diskette	\$89.95
------------------------------	---------

Model III (32K single disk system)

Formatted diskette	\$79.95
Cassette tape.	\$75.95

ASM/CMD, a disk based assembler which generates object code to disk or tape (disk only on Model II). Accepts any file format including ASCII Disk BASIC. Listing may be outputted to display, disk file, or paged with adjustable forms to printer. Operates under standard Z80 Zilog Mnemonics with 9 pseudo operations. Comes complete with operating manual.

Model I (16K single disk system)

Formatted diskette	\$34.95
Cassette tape.	\$29.95

Model II

Formatted diskette	\$59.95
------------------------------	---------

Model III (32K single disk system)

Formatted diskette	\$49.95
Cassette tape.	\$45.95

PACK/CMD removes spaces from text files generated by XEDIT, and EDIT-80 to reduce file lengths by 5 to 40 percent. PACK will also strip comment fields and line numbers for additional space savings. Text can be masked for upper case only. Does not destroy compatibility of assembly and FORTRAN source files. Comes complete with instructions.

Model I (16K single disk system)

Formatted diskette	\$14.95
Cassette tape.	\$ 9.95

Model II

Formatted diskette	\$19.95
------------------------------	---------

Model III (32K single disk system)

Formatted diskette	\$14.95
Cassette tape.	\$ 9.95

Special package, XEDIT, ASM, and PACK.

Model I

Formatted diskette	\$79.95
Cassette tape.	\$74.95

Model II

Formatted diskette	\$149.95
------------------------------	----------

Model III

Formatted diskette	\$129.95
Cassette tape.	\$125.95

XDIR/CMD, an extended directory that offers more than the standard TRSDOSTM directory. XDIR will do multiple drive directories with all file attributes including extent locations, file length, EOF index, EOF record, protection level, LRL, password indication, track lockout indication, and much more. XDIR will also display to the printer.

Model I (16K disk system)

Formatted diskette	\$19.95
Cassette tape.	\$15.95

CALL/CMD extends and improves the TRSDOSTM AUTO function. Can be enabled and disabled by prompts, and through keyboard, resident program, or the call file.

Model I (16K single disk system)

Formatted diskette	\$19.95
Cassette tape.	\$15.95

TANDON/CMD improves TRSDOSTM by allowing higher step rate, extending access to 40 tracks for the new Tandon disk drives. Also fixes the break key problem.

Model I (16K single disk system)

Formatted diskette	\$14.95
Cassette tape.	\$ 9.95

DEXER/CMD, a disk exerciser emulator program designed to speed repair of any TRS-80TM compatible disk drive. DEXER eliminates the need for the Shugart SA809 test fixture and decreases repair time with easy to use commands and on screen display of required set up data. DEXER was written specifically for the repair technician and Shugart or Tandon disk drives. Shugart alignment diskette or equivalent and a 30Mhz oscilloscope required. One key commands allow easier adjustments necessary for Shugart alignment. DEXER is not for general disk testing and is recommended only for service personnel who have previous experience in disk drive repair.

Model I (16K single disk system)

Formatted diskette	\$24.95
Cassette tape.	\$19.95

Dip shunts for conversion and upgrades for the TRS-80TM. Comes complete with instructions for A, D, E, and G level boards and new 2 chip level II.

Two dip shunts and instructions \$1.00

Please send check or money order to:

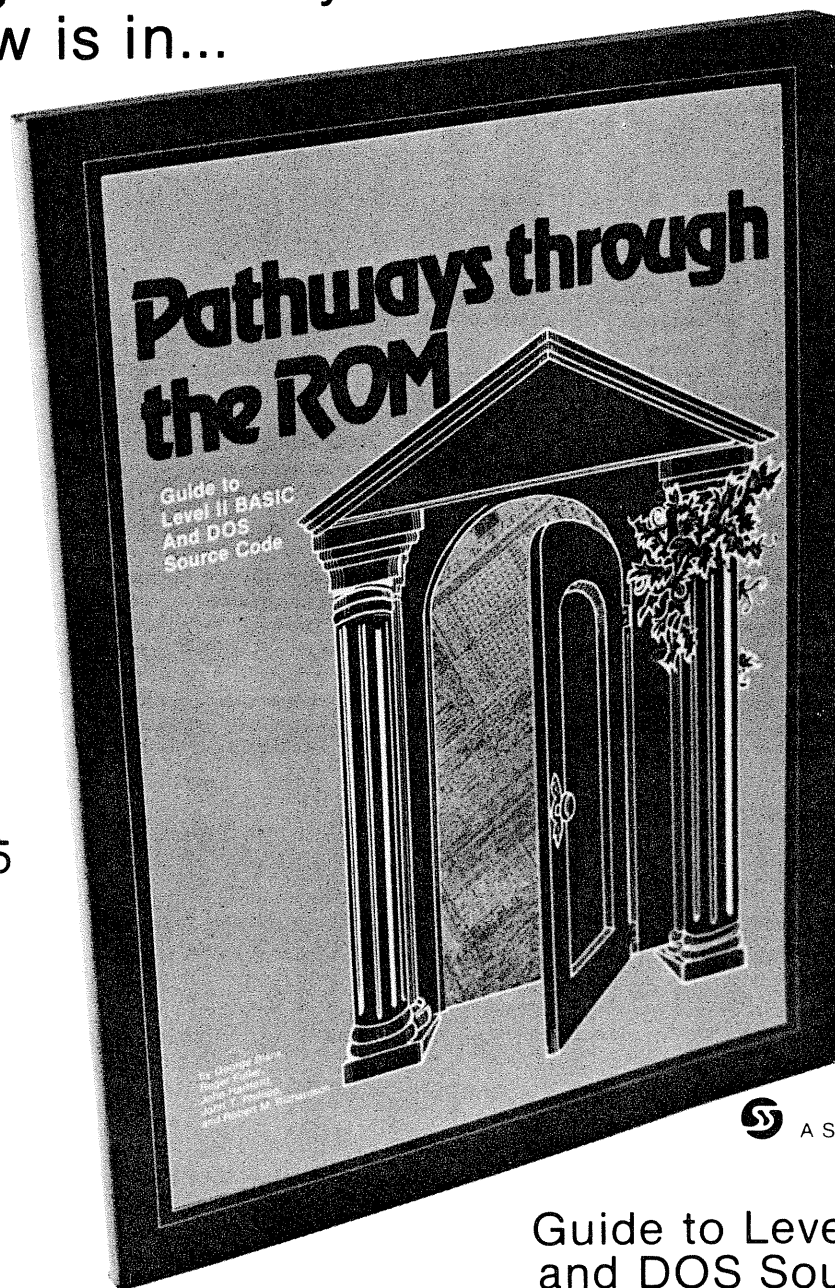
MiProg
P. O. Box 27014
Minneapolis, MN
55427
612-574-1711

Minnesota residents add 4% sales tax. Outside continental U.S.A., add \$3.50 postage and handling.

TRS-80TM and TRSDOSTM is a trademark of Radio Shack, a Division of Tandy Corporation.

FB85	ED58BEFB	01480	TOPSAV	LD	DE, (SPTR)	;SAVE TOP LINE OF DISPLAY
FB89	21003C	01490		LD	HL, 3C00H	; .
FB8C	014000	01500		LD	BC, 64	; .
FB8F	EDB0	01510		LDIR		; .
FB91	EB	01520		EX	DE, HL	;CHECK SPTR FOR OVERFLOW
FB92	11C0FF	01530		LD	DE, BUFFER+1024	; .
FB95	DF	01540		RST	18H	;TOO BIG? (COMPARE HL-DE)
FB96	3803	01550		JR	C, HLOK2	; NO: OKAY
FB98	21C0FB	01560		LD	HL, BUFFER	; YES: WRAP AROUND TO TOP
FB9B	22BEFB	01570	HLOK2	LD	(SPTR), HL	;SAVE NEW SPTR
FB9E	C9	01580		RET		;AND GET OUT
FB9F	CD0000	01590	KEYLNK	CALL	\$-\$;CALL REGULAR KBD
FBA2	4F	01600		LD	C, A	;SAVE CHAR
FBA3	3A8038	01610		LD	A, (3880H)	;SHIFT KEY DOWN?
FBA6	B7	01620		OR	A	; .
FBA7	79	01630		LD	A, C	; (RESTORE CHAR)
FBA8	C8	01640		RET	Z	; NO: RETURN
FBA9	FE20	01650		CP	' '	;SHIFT-BLANK?
FBA8	280A	01660		JR	Z, MODE0	; YES: MODE 0
FBA9	FE0D	01670		CP	0DH	;SHIFT-ENTER?
FBAF	C0	01680		RET	NZ	; NO: RETURN
FB80	3EFF	01690		LD	A, 0FFH	; YES: MODE 255
FB82	32BCFB	01700		LD	(MODE), A	; .
FB85	AF	01710		XOR	A	;RETURN NULL CHAR
FB86	C9	01720		RET		; .
FB87	AF	01730	MODE0	XOR	A	;MODE=0
FB88	32BCFB	01740		LD	(MODE), A	; .
FB8B	C9	01750		RET		;AND SO DOES RETURNED CHAR
FBBC	FF	01760	MODE	DEFB	255	;STARTS UNCONTROLLED
FBBD	00	01770	COUNT	DEFB	0	;NOTHING BACKED UP
FB8E	C0FB	01780	SPTR	DEFW	BUFFER	;TOP OF BUFFER
0400		01790	BUFFER	DEFS	1024	;SCROLL SAVE AREA
0040		01800	TBUF	DEFS	64	;TEMPORARY SAVE AREA
FA68		01820		END	START	;AUTOSTARTS AT START
00000	TOTAL	ERRORS				
BACKUP	FB37	01140	01030			
BASIC	06CC	00100	00290			
BUFFER	FBC0	01790	00220	01350	01380	01530 01560 01780
CALKEY	FB0E	00930	01070	01160		
COUNT	FBBD	01770	00610	00650	01140	01180
CTEST	FAC9	00610	00830	00870		
CTLCHR	FAA1	00410	00360			
DOIT	FAC2	00580	00330	00400	00420	00440 00470
DOS	402D	00110				
DSPLY	FA8A	00300	00160			
GETKEY	FAC6	00600	00520	00880		
HLOK1	FB6F	01390	01370			
HLOK2	FB9B	01570	01550			
KEYBD	FB07	00890	00600	01470		
KEYLNK	FB9F	01590	00190	00200		
MEM16K	7A68	00120				
MEM32K	BA68	00130				
MEM48K	FA68	00140	00150			
MODE	FBBC	01760	00480	00540	00810	00860 01040 01090 01700
			01740			
MODE0	FB87	01730	01660			
OUT	FB2D	01080	00960	00980	01010	
OUT1	FB31	01100	01060			
SCOK	FABD	00550	00500			
SCROLL	FAB0	00480	00390	00630		
SPTR	FB8E	01780	00670	01310	01390	01420 01480 01570
START	FA68	00160	01820			
TBUF	FFC0	01800	00680	00770	01230	01430
TOPSAV	FB85	01480	00560	00710		

Unlock the hidden power
of your computer for fast and
easy programming! Use ROM
routines in your BASIC
and Assembly Language
programs! All you need to
know is in...



ALL
ONLY
\$19.95

plus \$1 shipping

INCLUDES:

SUPERMAP

From Fuller Software (\$18.95)

**TRS-80
DISASSEMBLED
HANDBOOK**

by Robert Richardson (\$10.00)

HEX MEM

by John Phillipp
Monitor written in BASIC

**Z-80
DISASSEMBLER**

by George Blank



A SoftSide Publication

**Guide to Level II BASIC
and DOS Source Code**

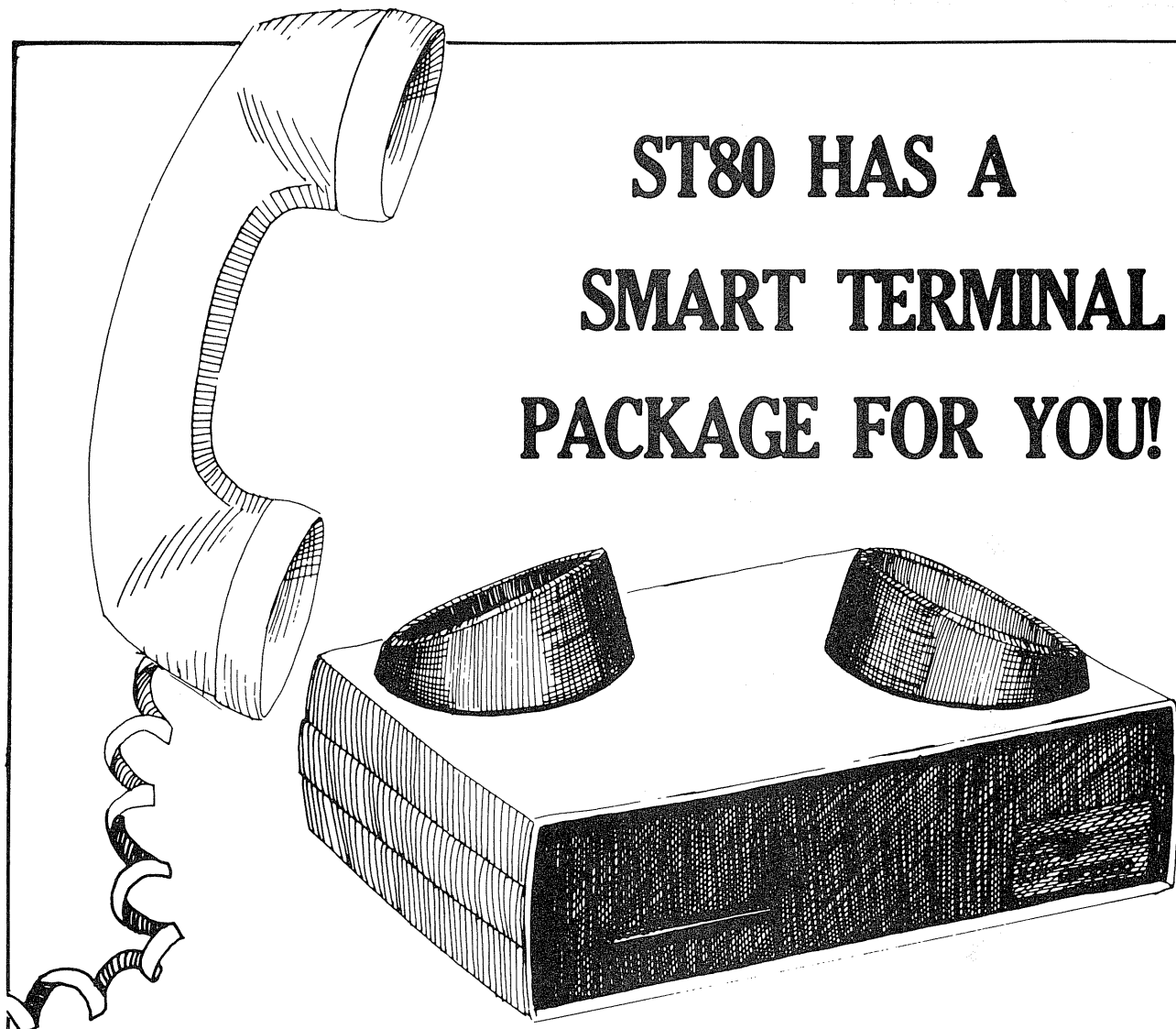
Description of the contents of the Level II BASIC ROM by memory locations, by function, and in lesson format. Includes several BASIC and Assembly Language programs in listing format to examine and use ROM routines.



ORDER TOLL-FREE
(In NH call 673-5144)

1-800-258-1790

The Software Exchange



ST80 HAS A SMART TERMINAL PACKAGE FOR YOU!

All four programs include the ability to use an unmodified TRS-80TM keyboard to produce RUB, ESC, and other control characters for time sharing, software control of the RS-232-C board, repeat key, bell, software support for the three most common upper/lower case hardware conversion, and line printer output.

ST80* UC

Preset parity, word length, and baud rate (regardless of switch settings on the RS-232-C board) for THE SOURCE, MICRONET, and FORUM 80, automatic testing of the RS-232-C board, and even spooling of prepared messages on tape directly into FORUM 80 using a basic program supplied as a line listing. 4K Level II cassette, \$24.95

ST80*

Reprogram your RS-232-C board from the keyboard, and run at different baud rates. Note: does not have auto testing of the RS-232-C or tape spooling. 4K Level II cassette \$49.95

ST80* D

Connection time clock, option of user-created translation tables for keyboard, gathering and pre-formatting data to be sent directly from disk to host computer, spooling of received files to disk or printer, editing of received files, and auto logon. Use it with VTOS 3.1, and get device-driven I/O, job logging, and chaining. 32K disk program, \$79.95

ST80* III

ST-80 D with extra utility programs. 32K disk program, \$150.00

The Software Exchange

6 South Street, Milford, NH 03055
ORDER TOLL-FREE: (In NH call 673-5144)
1-800-258-1790



* A trademark of Lance Micklus, Inc.

Allows you to store up to 15 mini-diskettes in a strong molded plastic carrier.

SOUNDWARE adds a whole new dimension to your computer games.

Programs come alive with laser sounds, bounces, clicks, sirens, bird calls, music notes, tunes, and whatever else your imagination dreams up. Just slip in two AA batteries, plug into your computer, and have fun.

SOUNDWARE
SOFTWARE programs are also available to enhance the enjoyment of your computer. \$29.95

Premium quality digital recording tapes. These 5-screw cassettes come packaged in boxes of 10. They are offered in 10 and 20 minute lengths.

C-10	\$6.95 + \$1
C-20	\$7.95 + \$1

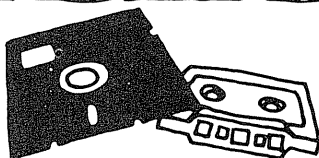
DYSAN:

104/1 One-sided single density (one side certified error free at single density of 3979 bpr.) In track and between track testing is performed to ensure that the entire recording surface of every disk is 100% error-free. Box of 5 diskettes	\$29.95 + \$1
--	---------------

Box of 10	\$34.95 + \$2
Box of 100	\$299.00 + \$3

Encased in a tough jacket which resists handling damages. 100% certified error-free performance. Low modulation provides better signal stability.

Box of 10	\$39.95 + \$2
-----------------	---------------



3-M Scotch 7400 head cleaning kit is simple and easy to use. You simply saturate the write head cleaning fabric in the cleaning diskette with the cleaning solution, insert the diskette into the drive and turn it on. The rotating cleaning fabric alternately wipes the heads with the solution and the dry surface, removing contamination from the read write head. Each kit contains 10 cleaning diskettes which will allow you a total of 150 cleanings. \$29.95

It consists of a template for locating the adjustment screw. An alignment tape is also included and a special screwdriver as well. The kit will allow you to align the heads to an industrial standard, ensuring consistent loading results. \$9.95

PREVENTS:

Computer drive's clamping hub
from tearing disk's center hole;
Coating removal, scuffing,
dimpling;
Data loss caused by improper
rotation.
\$14.95 Refills \$7.95

Prevents damages to your diskettes. These are high density, ultra lightweight polymer shipping envelops. Pack of 5..... \$4.95 + \$1

The Software Exchange

6 South St. Milford, NH 03055

ORDER TOLL-FREE: (In NH call 673-5144)

1-800-258-1790



IT'S A DAISYWHEEL COMPUTER
PRINTER & AN ELECTRONIC
TYPEWRITER
BUT ...

IT WON'T WORK!

...UNLESS YOUR
PRINTER & YOUR
SOFTWARE ARE
TOTALLY
COMPATIBLE

Only
\$2850

Suggested Retail
FREIGHT INCLUDED

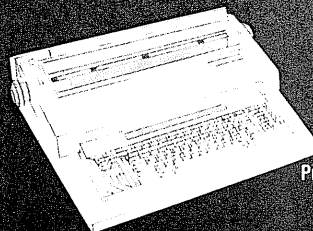


Lowest Price
Daisywheel Printer with ASCII Option
on the Market!

The TYPEPRINTER 221^{T.M.}

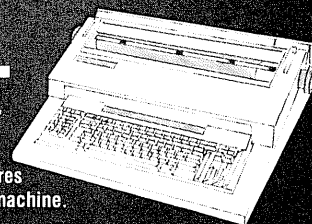
Is a TYPEWRITER QUALITY, DAISYWHEEL PRINTER that is Totally Compatible with All Word Processors. That's because the TYPEPRINTER 221 may be PROGRAMMED in PLAIN ENGLISH, Imbedded within The Text File of All Word Processing Software!

Use the 221 as your...



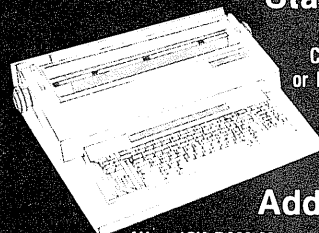
Electronic Typewriter

When not being used as a Computer Printer, the 221 becomes a fully functional Electronic Typewriter.



Computer Printer

It's a Daisywheel Computer Printer with more standard features and available options than any other machine.

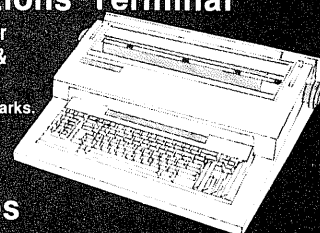


Stand Alone Terminal

Available options allow the 221 to Communicate with Distant Computer or Information Services such as Source, Micronet & others.

Tele-Communications Terminal

Option available to allow your 221 to access the Teletype & Telenex networks.
Telex & Teletype are registered trademarks.



Additional Options

4K or 16K RAM Memory which can be used as INPUT or OUTPUT Buffers. Also use as an Automatic Spooler to your computer. Bi-Directional Communications from The 221 to your Pet, Apple or TRS-80. Nothing else to buy. Lawyers, Accountants and others will find our Automatic Strike-Out Type and High Density Spacing options very useful.

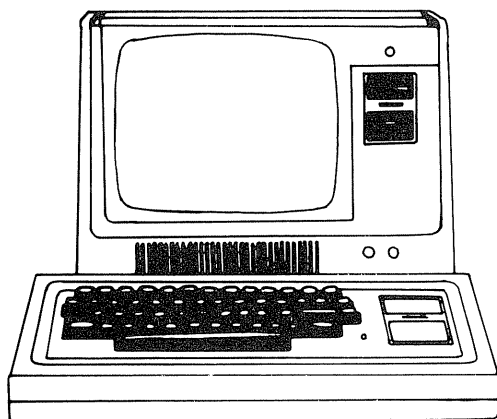
Built-in Features

The 221 Centers Copy Automatically, Sets Columns, Prints in Reverse, Bold Face and Underlines Automatically. The 221 also Justifies Right, Types in Three Pitches and does Proportional Spacing. It Types in Spanish, French, German, Italian and Portugese as well as English. And much, much more!



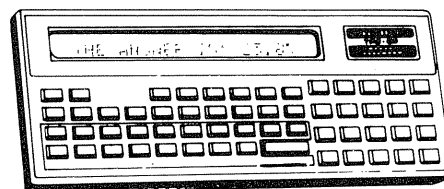
6 South Street, Milford, NH 03055

TOLL FREE ORDERS:
1-800-258-1790
(In NH call 673-5144)



MODEL II \$3699.00

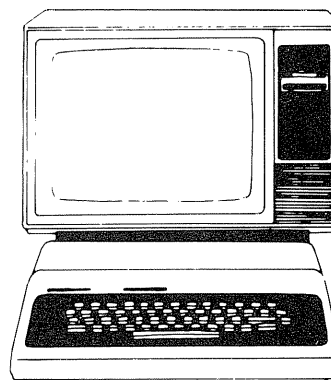
	LIST PRICE	OUR PRICE
Model II, 64K RAM	\$3899.00	\$3699.00
Model III, 16K RAM	\$999.00	\$929.00
Model III, 32K Dual Disk	\$2495.00	\$2299.00
Pocket Computer w/Interface	\$298.95	\$269.00
TRS-80 Color Computer	\$399.00	\$359.00
TRS-80 Color Computer Expanded	\$599.00	\$519.00
COMM-80 Interface	\$179.95	\$159.95
CHATTER BOX Interface	\$279.95	\$259.95
DISK-80 Interface	\$349.95	\$329.95
Expansion Interface, no RAM	\$299.00	\$279.00
Expansion Interface, 16K RAM	\$418.00	\$339.00
Expansion Interface, 32K RAM	\$537.00	\$399.00
RS-232-C Board	\$99.00	\$89.00



POCKET COMPUTER \$269

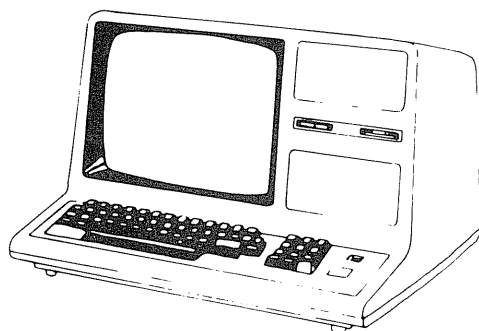
with interface

	LIST PRICE	OUR PRICE
TRS-232 Printer Interface		\$59.95
16K Memory Kit, TRS-Keypad	\$119.00	\$59.00
16K Memory Kit, TRS-Exp. Int.	\$119.00	\$59.00
Upper/lower Mod Kit	\$59.00	\$24.95
Video Reverse Kit		\$23.95
CPU Speed-up Kit		\$24.95
Percom Electric Crayon, w/cable		\$279.95
TRS-80 Dust Cover (3pc set)	\$9.95	\$7.95
TRS-80 Computer Case	\$109.00	\$99.95
TRS-80 Monitor Case	\$84.00	\$84.00



COLOR COMPUTER \$359

	LIST PRICE	OUR PRICE
Percom, TFD-100, 40-track	\$429.95	\$399.00
Percom, Dual TFD-100 Drives	\$849.00	\$799.00
Percom, TFD-40, 40-track	\$399.95	\$379.00
Percom, TFD-200, 77-track	\$675.00	\$629.00
Hardside, 40-track Disk Drive	\$399.00	\$359.00
Percom Data Separator		\$29.95
Extender Card	\$15.95	\$15.00
2-Drive Cable	\$29.95	\$29.00
4-Drive Cable	\$39.95	\$39.00



MODEL III \$929

TERMS:

TERMS: Prices and specifications are subject to change. HARDSIDE accepts VISA & MASTERCARD. Certified checks and Money Orders; Personal checks accepted (takes 3 weeks to clear). HARDSIDE pays all shipping charges (within the 48 states) on all PREPAID orders OVER \$100.00. On all orders under \$100 a \$2.50 handling charge must be added. COD orders accepted (orders over \$250 require 25% deposit) there is a \$5.00 handling charge. UPS, Blue Label, and Air Freight available at extra cost.

TO ORDER TOLL-FREE:
1-800-258-1790
 (In NH call 673-5144)

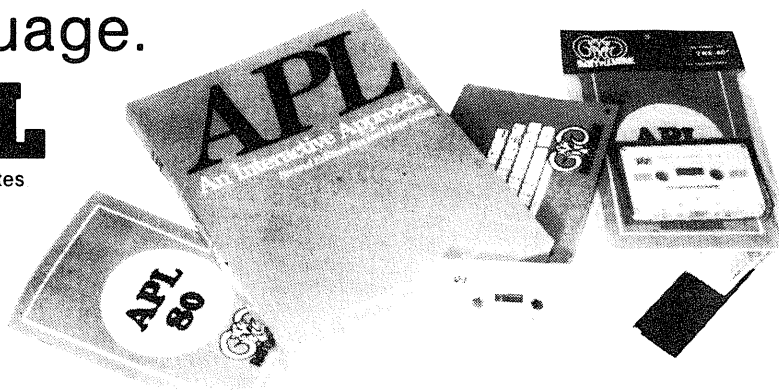
HARDSIDE 



Elegance and power in a mathematical language.

APL

by Phelps Gates



Software you can rely on.

Now, a high-level, scientific programming language that doesn't cost \$200 or \$300 for the home computer. This language is perfect for the mathematician, scientist, engineer, or anyone who just wants to learn a new language. The power of this language is in its strong mathematical operations, especially with regard to matrices and vectors. Programs requiring matrix multiplication or other matrix problem solving that would require hours of programming time in BASIC are solved quickly and with minimal effort in APL. Not only is math made easy, but upon gaining proficiency in APL programming various string manipulations become child's play.

To aid in learning APL, lessons are included on the disk. Starting from the basics, you are brought step by step through the various programming techniques involved with APL. These lessons act as a tutor in a "learning by doing" atmosphere which will have you "talking APL" in no time. Also available is the book, **APL: An Interactive Approach**, which reinforces many of the examples given in the lessons. The book also provides additional insight into APL programming.

LIMITATIONS

Due to the absence of the special APL character set on the TRS-80TM, APL-80 uses shifted letters to represent the various APL characters. These shifted letters are identified on the screen by a graphics block before each shifted letter. If you have a modified TRS-80TM, a lower case driver is included to display the shifted letters on the screen.

In addition to the keyboard limitations, there are several other limitations. Lamination, domino, and matrix inverse are not implemented but can be derived with user-defined functions.

Multiple specifications must be split into two statements unless the left-hand assignment is to a quad. This also applies to implied multiple specifications.

Reduction and reshape (p) are not permitted for empty arguments, the argument of add/drop may not be scalar, empty indices are not permitted.

A quad (q) can't be typed in response to a quad (nor can the name of a function which itself gets input from a quad). Quote-quad (m) is permitted.

No more than 32 user functions can be defined in a single workspace and a function may not contain more than 255 lines.

A comment (c) must occupy a separate line; a comment can't follow a function statement on the same line.

In the tape version, arrays are limited to five (5) dimensions.

FEATURES

APL-80 on disk contains the following features:)SAVE and)LOAD workspace on disk,)COPY other workspaces into current ones; Return to DOS for directory or commands without losing your workspace; Send output to lineprinter; Five workspaces of lessons included; Sequential and random files; 15 digit precision; Monadic and dyadic transposition; Easy editing within FUNCTION lines; Latent expression (FUNCTION can "come up running" when loaded); Tracing of function execution; Real-time clock; User-control of random link; Workspace is 25587 bytes (in 48K machine); Arrays may have up to 63 dimensions.

COMMANDS APL-80

APL-80 supports the following commands: Absolute value, add, and, assign, branch, catenate, ceiling, chr\$/asc, circular, combinatorial, comment, compress, deal, decode, divide, drop, encode, equal, expand, exponential, factorial, floor, format, grade down, grade up, greater, greater/equal, index generator, indexing, index of, inner product, label, less, less/equal, logarithm, maximum, member, minimum, multiply, nand, negate, nor, not, not equal, or, outer product, peek, poke, quad, quote quad, random, ravel, reciprocal, reduction, reshape, residue, reverse, rotate, scan, shape, sign, system, subtract, take, transposition.

SPECIFICATIONS

Minimum system requirements: 32K disk system (48K recommended) Includes APL-80, Five workspaces of lessons, instruction manual.

Price:\$39.95 on disk

Reduced feature: 16K Level II tape version, no lessons.

Transpositions, format, and inner product not implemented. Reduced domain for some functions. 6 digit accuracy.

Price:\$14.95 on cassette

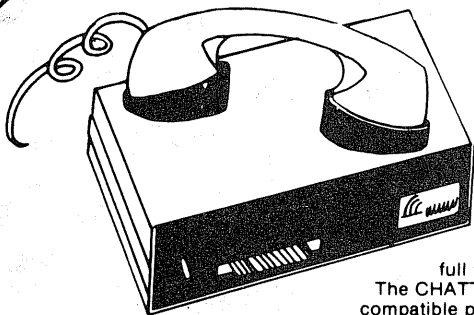
APL: An Interactive Approach

Price:\$16.95 (\$3.00 shipping charge)

The Software Exchange

TO ORDER TOLL-FREE
1-800-258-1790
(In NH call 673-5144)





The Chatterbox

A TRS-80 Interfacing Alternative

The CHATTERBOX is a unique packaging combination of the presently available COMM-80 I/O Interface for the TRS-80* and an acoustic modem. This one box is all that is required to turn even a barebones 4K TRS-80* into a full time-sharing terminal.

The CHATTERBOX includes built-in programmable 50-19200 baud serial port, a Centronics compatible parallel printer port, a 300 baud acoustic originate modem, and a spare TRS-BUS expansion connector. It comes complete with power supply, ribbon cable and connector, user's manual, and terminal software for immediate operation. When the modem is in use, the complete data conversion is automatically routed to the serial output port where it can be logged on a printer.

The CHATTERBOX is the only peripheral needed to allow a TRS-80* to communicate with time-sharing systems such as MICRONET and the SOURCE.

It is completely hardware and software compatible with existing TRS-80* products and connects either to the keyboard connector or screen printer port on the RS Expansion Interface. Features: Full 8-bit parallel port; RS-232-C serial port (up to 19,200 baud); Acoustic modem; TRS-BUS connector for future expansion; Connects to Keyboard or E.I.; Includes terminal software; Users manual; Power supply \$259.95

PRINTERS

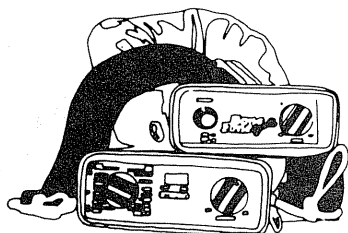
	LIST PRICE	OUR PRICE
Centronics 730	\$795.00	\$749.00
Centronics 730-3	\$895.00	795.00
Centronics 737	\$995.00	\$869.00
Centronics 779	\$1395.00	\$1095.00
Centronics 779 w/lower case	\$1595.00	\$1195.00
NEC 5510 SpinWriter	\$3195.00	\$2595.00
NEC 5520 SpinWriter	\$3395.00	\$2895.00
NEC 5530 SpinWriter	\$3195.00	\$2495.00
NEC Tractor-Feed Option	\$249.00	\$225.00
LRC 7000* (64-col.)	\$405.00	\$299.00
LRC 7000* (40-col.)	\$389.00	\$289.00
OkiData Microline-80	\$800.00	\$699.00
Tractor-Feed Option	\$140.00	\$129.00
RS-232-C Option	\$299.00	\$279.00

LRC to TRS-80	\$20.00
LRC to PET, IEEE	\$59.00
LRC to RS232C, male or female	\$65.00
730 or 737 to TRS-80	\$29.00
NEC or 779 to TRS-80	\$35.00
RS-232-C to RS-232-C, male to male	\$24.95

GENERAL INTEREST

	LIST PRICE	OUR PRICE
BSR System X-10 Home Controller	\$124.95	\$109.95
BUSY BOX, TRS-80	\$114.95	\$99.95
BUSY BOX, S-100	\$119.95	\$114.95
AC-SFK-31 Line Filter	\$24.95	\$19.95
ISO-2 Line Filter & Isolator	\$56.95	\$49.95
ISO-2/CBS Line Filter-Isolator	\$70.95	\$59.95
ISO-7/CB Super Filter-Isolator	\$146.95	\$99.95
Mini-Flex Diskette File	\$24.95	\$19.95
CASIO C-80 Calculator Watch	\$49.95	\$44.95
BONE FONE	\$69.95	\$56.95
LOGOS-9 Printing Calculator	\$99.95	\$79.95

SPECIAL



**BONE
FONE
\$56.95**

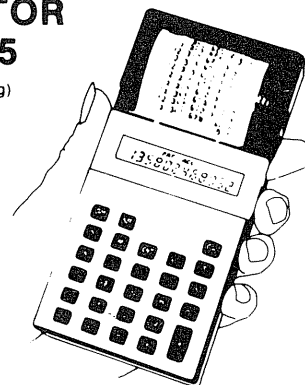
(plus \$2.50 postage and handling)

**CASIO
C-80 CALCULATOR
WATCH \$44.95**

(plus \$2.50 postage and handling)

**OLIVETTI
LOGOS-9
PRINTING
CALCULATOR
\$79.95**

(plus \$2.50 postage and handling)



TO ORDER TOLL-FREE
1-800-258-1790

(In NH call 673-5144)



Our signature of quality.

TSE began in the basement of our publisher's home. In those days we did everything from reviewing submissions, writing documentation, and duplicating cassettes, to licking envelopes. There was correspondence with the authors, telephoning, equipment problems, authors' contract negotiations and more and more envelopes which needed stamps.

TSE explored the 'software' territory in those 'frontier days' carefully . . . scouting out only the very best, leaving the weaker pieces by the roadside. The number of submissions was increasing, and we had all that we could do to provide proper service for our customers. We decided then and there that we would publish only 'the' very finest software available and commit ourselves to a policy of 'selectivity' and strong customer service.

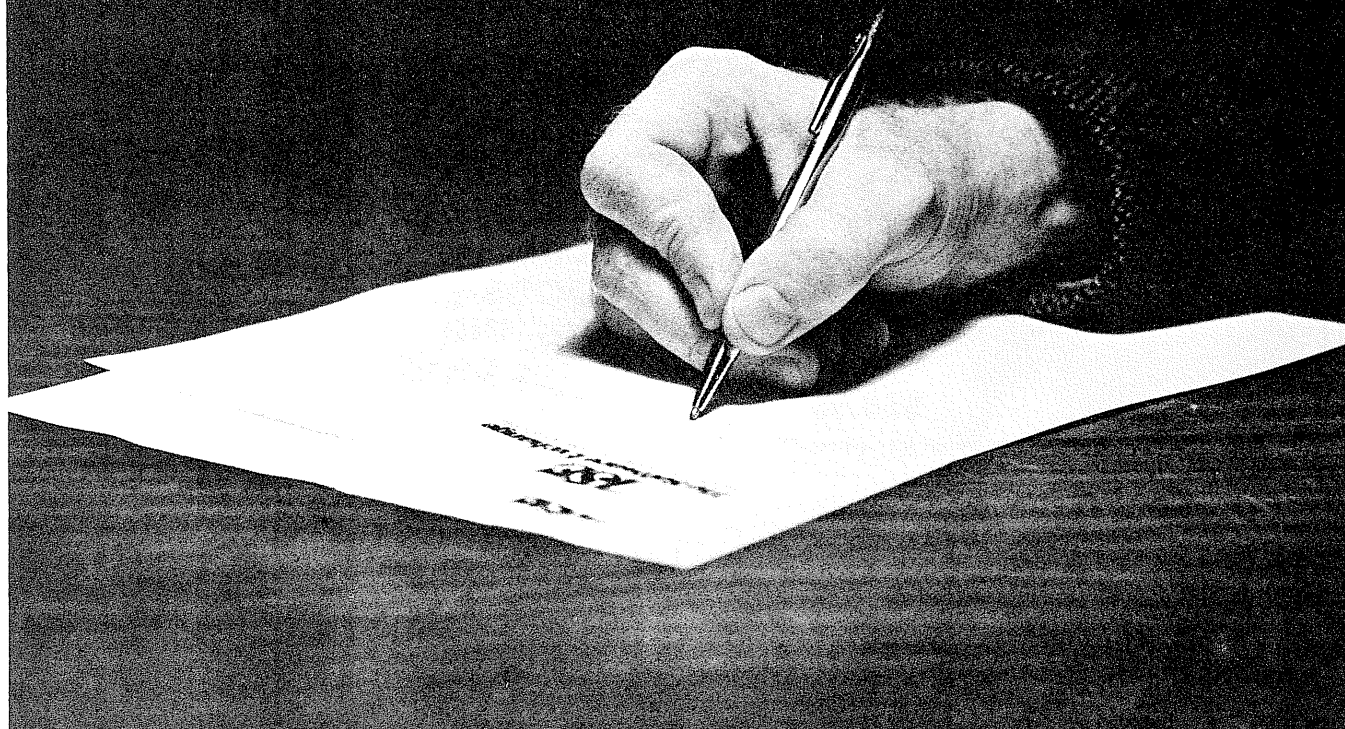
Well, it worked Since those days in the basement we have grown from a Mom and Pop operation with a dozen software titles to a company with 40 employees and over 500 titles. TSE distributes software for over 50 different vendors. We considered ourselves 'pioneers,' and as such we learned many things about our business and about our customers. We've made mistakes . . . who hasn't, however, the two most important factors in our success have been a keen eye towards selecting good software and a commitment towards strong customer support.

We think that you will profit from TSE's experiences and come to find a consistency in product selection and a mandate to continued customer service in the months and years ahead.

William F. Gollan
Marketing

The Software Exchange

6 South Street, Box 68, Millford, NH 03055 603-673-5144



How to Read Disk Directories

Debbi Tesler
N Hollywood, CA

You may ask, "Why would I want a program that reads the Directory? The 'DIR' command already does that for me." Let me suggest a few reasons for wanting to make the contents of the Directory available to your own programs:

1. to develop a file cataloging system;
2. to present the Directory contents in other formats;
3. to map the granules (useful in file recovery).

We had a fourth and far more pressing reason: we wanted to convert about forty 35-track diskettes to 40-track capacity. Our MPI drives supported 40 tracks, but for the first six months, we used TRSDOS and then NEWDOS. During this time, we created a sizeable library of 35-track diskettes. When we upgraded to NEWDOS40, we were immediately faced with a conversion problem. Short of using SUPERZAP to modify the GAT (which by itself would not solve the problem), the only obvious solution was to copy each file, one at a time, to another diskette. This would have required a great deal of manual effort, and it was clear that copying 600 files by hand would be tedious, time-consuming, and prone to error. Since we had bought a computer to relieve ourselves of tedium, time-wasting, and mistakes, this manual approach clearly would have been counter-productive.

My son pointed out that a good solution would be to develop a program (aha, automation!) that would read the Directory and then build a list of commands to perform the needed copying, so I tackled the reading of the Directory.

This article will show you how to read the Directory from TRSDOS or NEWDOS.

The Method

We decided to read the Directory in Assembly language, since it avoided the barriers to System files. The method was to find the contents of an ordinary opened Data Control Block (DCB), and then modify it to reference the Directory. If you don't know Z80 Assembler or TRSDOS Disk I/O calls, be unconcerned: you can skip the method and just use the programs as they are listed later in this article.

The TRSDOS manual explains the format of most of an opened DCB, and a few minutes with DEBUG revealed the contents of the "reserved" fields. One reserved field contains the primary track and ending sector of the file; another contains the next record number to be accessed. It was simple to modify these fields to point to Track 17, Sectors 0-9, and

to call DOS to READ one physical record at a time into an I/O buffer. Each READ resulted in a return code '6' (attempt to read system data), and with the data requested. Naturally, we ignored the return code; more precisely, anything other than a '6' indicated an error!

Opening one file in order to read another was cumbersome, and proved unnecessary. The second version of the program contained a data area that looked like an already-opened DCB (pointing to the Directory, of course). No OPENs or CLOSEs were issued, yet the READs still worked. (It was faster than before, since those transients didn't have to be read in first).

The final version of the program, which is presented here, interfaces with a BASIC program. The BASIC program provides the drive number to be processed, I/O buffers, and whatever manipulation of the data may be needed. These things are done more easily in BASIC than Assembler, and we wanted to minimize our efforts.

Because it is possible to place the Directory on almost any track, not just on track 17 (is anyone actually doing that???), this final version first reads the Bootstrap record (track 0, sector 0) and obtains from the third byte within that record the track address of the Directory. I'm convinced that anyone who knows enough to successfully move the Directory could have modified this program as well, but for the sake of completeness, even this function is included.

The Assembler program is entirely relocatable, which makes it easy to include in DATA statements if you like to POKE them in place. That technique is used in the conversion program, but not in this article. For present purposes, the Assembler program can be placed anywhere in protected memory, and its address supplied to BASIC via the DEFUSR statement. If you don't have an Assembler, you can use DEBUG to enter the hex code (shown in the left-hand side of the listing), then the DUMP command to make a permanent file of the result.

The Data

The Directory track contains 2,560 bytes. The Assembler program reads these into memory, placing the data in an area supplied by BASIC. For ease of processing, this area was defined in string space. Unfortunately for the ascending-sequential nature of our thought processes, string space runs top-down, not bottom-up. Therefore, a word of explanation is in order before you try to process the Directory data.

The BASIC program defines a string array, D\$(79). Counting element zero, there are eighty elements in this array, and they will occur next to each other (they will be contiguous). By initializing each of them to be 32 bytes in length, a contiguous area of $80 \times 32 = 2,560$ is obtained (magic, isn't it?). However, D\$(0) has the highest address of these array elements, and D\$(79) has the lowest address. Backwards, isn't it?

I chose to ignore this inconsistency: the BASIC program passes the address of D\$(79), which is the lowest address of the 2,560 byte area. The Assembler program just reads the data in from that point to the end of the array area, greatly simplifying its logic and minimizing the program size. It is up to the BASIC program to process the array backwards, that is, to realize that Track 17, Sector 0 begins in D\$(79), while Track 17, Sector 9 ends in D\$(0). Since each sector contains 256 bytes and each array element contains 32 bytes, it takes eight array elements to contain each sector.

If you bear all that in mind, you should find it fairly straightforward to process the Directory contents.

One other point: Sector 0 contains the GAT table, Sector 1 contains the HIT table, and the Directory file entries themselves begin in Sector 2. Each entry takes 32 bytes (that's why the array elements are 32 bytes each), and there are 64 entries. For detailed information on these entries, see Harv Pennington's *"TRS-80 Disk & Other Mysteries."* That book made this development effort much easier than it otherwise might have been.

Summary

Figure 2 lists the Z80 Assembler program, "GETDIR/ASM," that actually reads in the Directory track. This program expects to be called from BASIC. You don't have to know Assembler in order to enter the object code directly into memory using DEBUG. If you do know Assembler, you can see how easy it is to read anything from a diskette, regardless of protection status.

Figure 1 lists the BASIC program, "READIR/BAS." The program is not particularly useful, except as a working illustration of the way in which GETDIR/ASM is called. You may easily add more processing logic at the end of this program, thereby turning it into a Directory listing routine or a simple catalog builder.

•

```

110 :REM: READDIR 1.1 02/16/80 21:15 -- READ DIRECTORY
120 :REM: THIS JUST READS THE DIRECTORY TRACK. IT IS USEFUL
130 :REM: AS A BASE ONTO WHICH ADDITIONAL FUNCTIONS ARE ADDED.
140 :REM:
150 :REM: DIRECTORY TRACK IS READ INTO ARRAY D$, WHICH
160 :REM: CONTAINS EIGHTY 32-BYTE ELEMENTS (D$(0)...D$(79))
170 :REM: THESE ELEMENTS SHOULD BE PROCESSED HIGH TO LOW, AS
180 :REM: SECTOR 0 PART 1 IS IN D$(79), PART 2 IN D$(78)...
190 :REM: SECTOR 0 PART 7 (LAST 32 BYTES) IN D$(72), ... ,
200 :REM: SECTOR 9 PART 7 IS IN D$(0).
210 :REM:
220 CLEAR 2600
230 DEFINT A-Z
240 :REM: CALCULATE AND ALLOCATE I/O BUFFERS IN STRING SPACE
250 BL=32 : ' BUFFER ELEMENT LENGTH <= 128
260 NB=2560/BL-1 : ' NO. BUFFERS NEEDED, LESS 1
270 DIM P(1) : ' 2-ELEMENT PARM LIST
280 DIM D$(NB) : ' NB*BL = 2560 BYTES FOR DIR.
290 NM=50 : ' NO. ELEMENTS FOR MLC PGM
300 DIM M(NM) : ' HOLDS MACH LANG READ ROUTINE
310 FOR I=0 TO NM : ' STORE MLC ROUTINE
320 READ M(I)
330 NEXT I
340 :REM: ALLOCATE CONTIGUOUS I/O BUFFER SPACE
350 FOR I=0 TO NB
360 D$(I)=STRING$(BL,0) : ' ALLOCATE I/O BUFFERS
370 NEXT I
380 PRINT "ENTER DRIVE NUMBER (0-3, DEFAULT=1)";
390 D=1
400 INPUT D : ' GET DRIVE NUMBER
410 IF D<0 OR D>3 THEN 380 : ' VALIDATE
420 P(0)=D : ' BUILD PARM LIST
430 P(1)=0 : ' EXPLICIT ALLOCATION
440 X=0 : ' PRE-ALLOCATE ALL VARS
450 :REM: POINT TO MACHINE LANGUAGE DIRECTORY READ ROUTINE...
460 DEFUSR0=VARPTR(M(0))
470 P(1)=VARPTR(D$(NB)) : ' 2ND PARM -> LOWEST ADDRESS
480 X=USR0(VARPTR(P(0))) : ' PASS 2 PARMS TO SUBRT
490 PRINT "RETURN CODE=";X : ' SHOW RESULTS
500 FOR I=NB TO 0 STEP -1
510 :REM: PRINT D$(I)
520 NEXT I
530 DATA -14859,-12843, 11,-8731,-12831, 2687,-8834, 17783, 899
540 DATA 5, 9038, 838, 24330, 2563, 8535, 63,-6691, 2497, 1771,-2816,
550 DATA -8719, 18800, 30173,-8894, 17268, 14029,-2748, 1060, 2622, 8
550 DATA 376,-7701, 9836,-12032,-3647,-25917,-32758, 96
550 DATA 0, 256,-64, 0, 2816, 4352,-255,-1,-1,-1,-1,-1,-1,-1,-
550 END

```

Figure 1

```

00010 ; GETDIR 1.4 02/18/80 21:00 - WITH BOOT VALIDATION
00020 ;PURPOSE: READ THE DIRECTORY TRACK INTO MEMORY.
00030 ;
00040 ;METHOD: USE A PHONY DCB THAT LOOKS AS THOUGH IT HAS
00050 ; BEEN OPENED PREVIOUSLY, BUT WHICH CONTAINS
00060 ; THE PHYSICAL ADDRESS OF THE DIRECTORY.
00070 ; READ TEN PHYSICAL RECORDS FROM THIS FILE,
00080 ; CHANGING THE RECNO AND I/O BUFFER FROM ONE REC
00090 ; TO THE NEXT. RESULT IS 2560 BYTES IN MEMORY.
00100 ;
00110 ; CALLING SEQUENCE:
00120 ;

```

Figure 2

```

00130 ; DIM P%(1), D$(79)
00140 ; FOR I=0 TO 79 : D$(I)=STRING$(32,0) : NEXT I
00150 ; ..
00160 ; ..
00170 ; X%=0 :REM ALLOCATE BEFORE USING 'VARPTR' COMMAND
00180 ; P%(0) = DRIVE NUMBER
00190 ; P%(1) = VARPTR (D$(79))
00200 ; X% = USR0 (VARPTR (P%(0)) )
00210 ;
00220 ; INPUTS:
00230 ;
00240 ; TWO INTEGERS, PASSED AS ELEMENTS 0 AND 1 OF AN
00250 ; ARRAY WHOSE VARPTR IS THE ARGUMENT OF A USR CALL.
00260 ; (THESE ARE IN P% ABOVE).
00270 ;
00280 ; PARM 1 IS THE DRIVE NUMBER TO BE READ.
00290 ; PARM 2 IS THE ADDRESS OF THE DOPE VECTOR OF THE
00300 ; LAST ELEMENT OF THE ARRAY TO BE USED AS
00310 ; THE I/O BUFFER.
00320 ;
00330 ; PROCESSING:
00340 ;
00350 ; TRACK 0, SECTOR 0 IS READ INTO MEMORY FROM
00360 ; THE DRIVE SPECIFIED IN P%(0). IF IT LOOKS LIKE A
00370 ; BOOT RECORD, ITS THIRD BYTE, WHICH IDENTIFIES THE
00380 ; DIRECTORY TRACK, IS STORED IN THE PHONY DCB. IF
00390 ; IT DOES NOT LOOK LIKE THE BOOT, THE DIRECTORY IS
00400 ; ASSUMED TO BE ON TRACK 17, AND X'11' IS STORED
00410 ; IN THE PHONY DCB. THE ENTIRE DIRECTORY TRACK IS
00420 ; THEN READ INTO THE AREA DEFINED THROUGH PARM 2,
00430 ; WHICH CONTINUES FOR 2650 BYTES. IT IS ASSUMED THAT
00440 ; D$ HAS BEEN DIMENSIONED WITH 20 ELEMENTS INCLUDING
00450 ; D$(0)... (DIM D$(79))... AND THESE ELEMENTS HAVE
00460 ; BEEN ASSIGNED TO 32 BYTE LENGTHS BEFORE THE CALL.
00470 ; THE FIRST PART OF SECTOR 0 IS IN D$(79),
00480 ; THE SECOND PART OF SECTOR 0 IS IN D$(78),
00490 ; AND THE LAST PART OF SECTOR 9 IS IN D$(0).
00500 ; KEEPING THE I/O BUFFERS IN BASIC'S STRING SPACE
00510 ; MINIMIZES THIS ROUTINE'S SIZE AND MAKES THE DATA
00520 ; EASILY AVAILABLE TO THE BASIC PROGRAM.
00530 ;
00540 ; OUTPUTS:
00550 ;
00560 ; 1. RETURN CODE PASSED AS THE FUNCTION VALUE. IF
00570 ; IT IS NOT '6', THE DIRECTORY WAS NOT READ.
00580 ;
00590 ; 2. THE ENTIRE DIRECTORY TRACK IN THE USER BUFFER,
00600 ; AS DESCRIBED ABOVE. THIS DATA IS VALID ONLY
00610 ; IF THE RETURN CODE WAS '6'.
00620 ;
00630 ; THIS PROGRAM HAS NO I/O BUFFERS, AND READS DIRECTLY
00640 ; INTO THE USER'S STRING ARRAY. THE PHONY DCB IS
00650 ; CONTAINED WITHIN THIS PROGRAM, HOWEVER.
00660 ;
00670 ; ERRORS:
00680 ;
00690 ; DRIVE NUMBER IS NOT CHECKED.
00700 ; INVALID PARAMETERS WILL PROBABLY CAUSE ABENDS.
00710 ; THE RETURN CODE FROM THE LAST DISK READ IS PASSED
00720 ; BACK TO THE USER AS THE FUNCTION RESPONSE. IT
00730 ; SHOULD BE CHECKED AGAINST THE CODES IN THE TRSDOS
00740 ; MANUAL, AND IF NOT '6' (ATTEMPT TO READ SYSTEM DATA)

```

```

00750 ; SOMETHING WENT WRONG... '6' IS THE CORRECT VALUE.
00760 ;
00770 ; THIS PROGRAM IS RELOCATABLE AND CONTAINS NO FIXED
00780 ; ADDRESSES TO ITS OWN LOCATIONS. IT DOES NOT USE
00790 ; THE ALTERNATE REGISTER SET, BUT DOES USE 'IX'.
00795 ; IT MAY BE PLACED ANYWHERE IN HIGH, PROTECTED MEMORY.
00800 ;
4436 00810 XREAD: DEFL 4436H
402D 00820 OPSYS: DEFL 402DH ;DOS RETURN POINT
0111 00830 ADIR: DEFL 0111H ;PRIMARY DIRECTORY EXTENT
0A7F 00840 GETPRM: DEFL 0A7FH ;CALL POINT FOR USER PARM
0A9A 00850 PUTPRM: DEFL 0A9AH ;CALL POINT TO RETN VALUE
00860 ;
B001 00870 ORG 0B001H ;PGM START
B001 00880 GETDIR: DEFL $
B001 F5 00890 PUSH AF ;SAVE ALL REGS
B002 C5 00900 PUSH BC ;EXCEPT HL
B003 D5 00910 PUSH DE
00920 ;
B004 CD0B00 00930 CALL 0BH ;GET CURRENT ADDRESS
B007 00940 BASE: DEFL $ ;HL -> 'BASE' NOW.
B007 E5 00950 PUSH HL ;USE IT FOR DATA INDEX
B008 DDE1 00960 POP IX
00970 ;
B00A CD7F0A 00980 CALL GETPRM ;GET USER PARMS
00990 ;
01000 ; PROCESS PARAMETER 1 (DRIVE NUMBER TO BE READ).
01010 ;
B00D 7E 01020 LD A,(HL) ;P1 IS DRIVE NUMBER
B00E DD776C 01030 LD (IX+DRIVE-BASE),A ;SET INTO DCB
01040 ;
01050 ; PROCESS PARAMETER 2 (ADDR OF ADDR OF I/O BUFFERS).
01060 ;
01070 ; A. PICK IT UP. IT IS THE ADDRESS OF A DOPE VECTOR.
01080 ; B. FROM THE DOPE VECTOR, GET THE ADDRESS OF A STRING.
01090 ; THIS WILL BE THE STARTING SINK FOR I/O DATA.
01100 ;
B011 23 01110 INC HL ;POINT TO 2ND PARM
B012 23 01120 INC HL
B013 4E 01130 LD C,(HL) ;P2 -> DOPE VECTOR OF
B014 23 01140 INC HL ;HIGHEST NUMBERED, LOWEST
B015 46 01150 LD B,(HL) ;ADDRESSED STRING ARRAY
01160 ;
01170 ; BC -> DOPE VECTOR OF D$(19).
01180 ;
B016 03 01190 INC BC ;GET PAST LENGTH VALUE
B017 0A 01200 LD A,(BC) ;GET STRING ADDR, LSB
B018 5F 01210 LD E,A ;COPY TO OTHER REG
B019 03 01220 INC BC
B01A 0A 01230 LD A,(BC) ;GET STRING ADDR, MSB
B01B 57 01240 LD D,A ;COPY THESE INTO DE
01250 ;
01260 ; 'DE' NOW POINTS TO THE STRING ARRAY IDENTIFIED BY THE
01270 ; DOPE VECTOR WHOSE ADDRESS WAS GIVEN IN PARM 2. THIS
01280 ; SHOULD BE THE HIGHEST NUMBERED (D$(19)) ARRAY ELEMENT,
01290 ; WHICH IS THE LOWEST ADDRESSED ELEMENT. THE ADDRESS
01300 ; IN 'DE' WILL BE MOVED TO 'HL' BELOW FOR FINAL USE.
01310 ;
01320 ;
01330 ; SET UP REGISTERS: A : SCRATCH
01340 ; B : NRN (NEXT RECORD NUMBER TO READ)
01350 ; C : NOT USED

```

```

                                01360 ;                DE : ADDRESS OF DCB USED FOR I/O
                                01370 ;                HL : ADDRESS OF I/O BUFFER TO BE USED
                                01380 ;                IX : ADDRESS OF 'BASE' REF POINT.
                                01390 ;
B01C 216600                    01400 LD      HL,DCB-BASE      ;GET REL ADDR OF DCB
B01F DDE5                      01410 PUSH    IX              ;GET ABS ADDR OF PGM BASE
B021 C1                        01420 POP     BC
B022 09                        01430 ADD     HL,BC              ;FORM ABS ADDR OF DCB
                                01440 ;
B023 EB                        01450 EX      DE,HL              ;DE->DCB, HL->IO BUFFER
B024 0600                      01460 LD      B,0              ;B=NRN (NEXT RECORD NMBR)
                                01470 ;
                                01480 ; READ THE BOOT RECORD IN ORDER TO FIND THE DIRECTORY.
                                01490 ;
B026 DD7070                    01500 LD      (IX+NRN-BASE),B ;GET RECORD ZERO
B029 DD7074                    01510 LD      (IX+DIRTRK-BASE),B ;TRACK ZERO
                                01520 ;
B02C DD7569                    01530 LD      (IX+BUFPTR-BASE),L ;INTO FIRST
B02F DD746A                    01540 LD      (IX+BUFPTR+1-BASE),H ;BUFFER
                                01550 ;
B032 CD3644                    01560 CALL    XREAD          ;READ THE BOOT RECORD
                                01570 ;
B035 4C                        01580 LD      C,H              ;SAVE MSB OF BUFF ADDR
                                01590 ;
                                01600 ; SEE IF THIS LOOKS LIKE A BOOT RECORD
                                01610 ;
B036 DD367411                  01620 LD      (IX+DIRTRK-BASE),11H ;SET DEFAULT
B03A 7E                        01630 LD      A,(HL)           ;SEE IF BOOT
B03B B8                        01640 CP      B                ;BYTE 0 SHOULD BE 0
B03C 2012                      01650 JR      NZ,NOBOOT        ;USE DEFAULT IF NOT
B03E 23                        01660 INC     HL
B03F 7E                        01670 LD      A,(HL)           ;
B040 FEFE                      01680 CP      0FEH             ;BYTE 1 SHOULD BE X'FE'
B042 200C                      01690 JR      NZ,NOBOOT        ;
B044 23                        01700 INC     HL                ;BYTE 2 -> DIR TRACK
B045 7E                        01710 LD      A,(HL)           ;
B046 B8                        01720 CP      B                ;MUST BE > 0
B047 3807                      01730 JR      C,NOBOOT         ;
B049 FE28                      01740 CP      40              ;MUST BE < 40
B04B 3003                      01750 JR      NC,NOBOOT        ;THAT IS, TRACK 1-39
                                01760 ;
B04D DD7774                    01770 LD      (IX+DIRTRK-BASE),A ;STORE IN DCB
B050                          01780 NOBOOT: DEFL    $
                                01790 ;
                                01800 ; DCB NOW APPEARS TO BE OPEN FOR DIRECTORY TRACK.
                                01810 ;
B050 F5                        01820 PUSH    AF              ;BALANCE 'POP' IN 'LOOP'
                                01830 ;
                                01840 ; END OF SETUP. PROGRAM IS READY TO READ A SECTOR AT A
                                01850 ; TIME FROM THE DIRECTORY INTO USER STRING SPACE.
                                01860 ;
B051                          01870 LOOP: DEFL    $
B051 F1                        01880 POP     AF              ;CLEAR RETCODE FROM STACK
B052 DD7070                    01890 LD      (IX+NRN-BASE),B ;SUPPLY RECORD NUMBER
                                01900 ;
B055 DD716A                    01910 LD      (IX+BUFPTR+1-BASE),C ;POINT TO BUFFER
                                01920 ;
B058 CD3644                    01930 CALL    XREAD          ;READ ONE SECTOR
B05B F5                        01940 PUSH    AF              ;SAVE RETURN CODE
                                01950 ;
B05C 0C                        01960 INC     C                ;BUMP PTR TO NEXT BUFFER
                                01970 ;

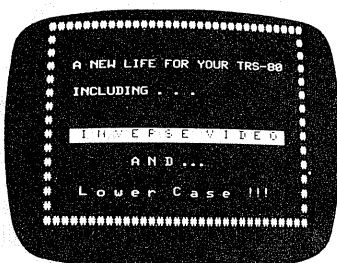
```

```

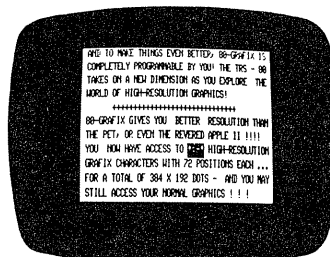
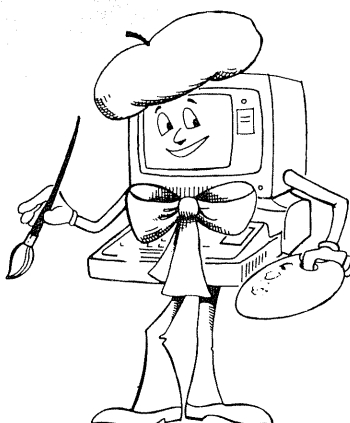
B05D 04      01980      INC      B      ;BUMP RECORD COUNT
B05E 3E0A    01990      LD        A,10    ;SEE IF 10 WERE READ
B060 B8      02000      CP        B
B061 20EE    02010      JR        NZ,LOOP  ;IF NOT, READ AGAIN
          02020 ;
B063          02030 RETURN: DEFL      $
B063 E1      02040      POP      HL      ;GET LAST RETURN CODE
B064 6C      02050      LD        L,H     ;SWAP TO LD ORDER
B065 2600    02060      LD        H,0     ;MAKE INTEGER VALUE
          02070 ;
B067 D1      02080      POP      DE      ;RESTORE REGS
B068 C1      02090      POP      BC
B069 F1      02100      POP      AF
          02110 ;
B06A C39A0A  02120      JP        PUTPRM   ;PASS RETCODE BACK.
          02130 ;
          02140 ; NEXT IS THE PHONY DCB.  IT APPEARS TO BE ALREADY
          02150 ; OPENED, POINTING TO DRIVE 1, TRACK X'11',
          02160 ; SECTORS 0-9.  THIS IS THE DIRECTORY AREA.
          02170 ;
B06D 8060    02180 DCB:      DEFW      6080H      ;BYTES 0-2 ARE RESERVED
B06F 00      02190      DEFB      0           ;AND CONTAIN X'806000'
B070 0000    02200 BUFPTR: DEFW      0           ;BYTES 3-4 => I/O BUFFER
B072 00      02210      DEFB      0           ;BYTE 5 = LGCL REC DELIM
B073 01      02220 DRIVE:  DEFB      1           ;BYTE 6 GIVES DRIVE NO.
B074 C0      02230      DEFB      0C0H      ;BYTE 7 RESERVED, PRE-SET
B075 FF      02240      DEFB      255        ;BYTE 8 = EOF OFFSET
B076 00      02250      DEFB      0           ;BYTE 9 = LRECL (=256)
B077 0000    02260 NRN:     DEFW      0           ;BYTES 10-11 = NEXT RECNO
B079 0B00    02270      DEFW      000BH      ;BYTES 12-13 = EOF RECNO
B07B 1101    02280 DIRTRK: DEFW      ADIR      ;BYTES 14-15 = PRI XTNT
B07D FFFF    02290      DEFW      0FFFFH     ;REMAINDER IS SECONDARY
B07F FFFF    02300      DEFW      0FFFFH     ;EXTENTS
B081 FFFF    02310      DEFW      0FFFFH
B083 FFFF    02320      DEFW      0FFFFH
B085 FFFF    02330      DEFW      0FFFFH
B087 FFFF    02340      DEFW      0FFFFH
B089 FFFF    02350      DEFW      0FFFFH
B08B FFFF    02360      DEFW      0FFFFH
          02370 ;
B08D          02380 ZEND:    DEFL      $           ;PGM END
B08C          02390 ZSIZE:   DEFL      $-GETDIR   ;PGM SIZE
B001          02400      END        GETDIR
00000 TOTAL ERRORS
ADIR  0111 00830 02280
BASE  B007 00940 01030 01400 01500 01510 01530 01540 01620
          01770 01890 01910
BUFPTR B070 02200 01530 01540 01910
DCB    B06D 02180 01400
DIRTRK B07B 02280 01510 01620 01770
DRIVE  B073 02220 01030
GETDIR B001 00880 02390 02400
GETPRM 0A7F 00840 00980
LOOP   B051 01870 02010
NOBOOT B050 01780 01650 01690 01730 01750
NRN    B077 02260 01500 01890
OPSYS  402D 00820
PUTPRM 0A9A 00850 02120
RETURN B063 02030
XREAD  443E 00810 01560 01930
ZEND   B08D 02380
ZSIZE  008C 02390

```

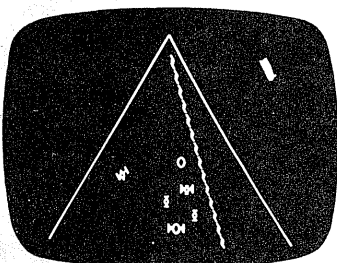
FROM **PROGRAMMA** HI-RESOLUTION GRAPHICS FOR THE TRS-80®



LOWER CASE
The 80-GRAFIX board includes two sets of lower case characters at no additional cost.



INVERSE VIDEO
The 80-GRAFIX board allows you to do inverse video to high-light your screen displays.



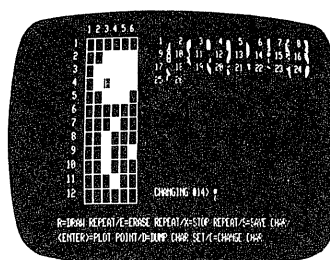
DEMONSTRATION PROGRAMS
The 80-GRAFIX board is supplied with a Character Generator software and several demonstration programs.

FINALLY, AT LAST...

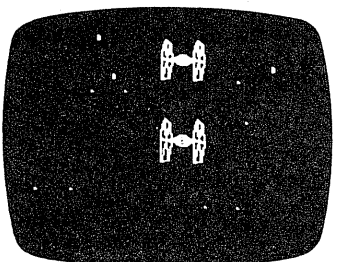
HI-RESOLUTION GRAPHICS is available for your TRS-80 computer system. The 80-GRAFIX board from PROGRAMMA International, Inc. gives your TRS-80 high resolution capability that is greater than the Commodore CBM/PET or even the revered APPLE II.

80-GRAFIX gives the TRS-80 an effective screen of 384X192 pixels, versus the normal 127X192 for the TRS-80, 80X50 for the CBM/PET, or the 280X192 of an APPLE II. As an added feature, 80-GRAFIX offers you lower case characters at no additional cost. Of course, you can also create your own set of up to 64 original characters using the supplied Character Generator software.

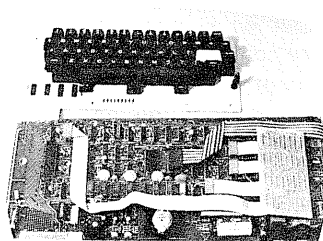
The 80-GRAFIX board is simple to install (note that this voids your Radio Shack warranty), and programming is done through BASIC. 80-GRAFIX opens up a whole new realm of software development and excitement never dreamed of for the TRS-80!



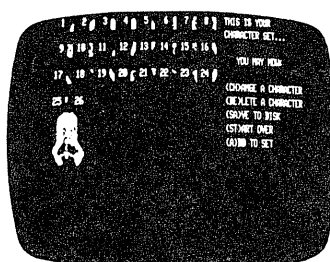
CHARACTER GENERATOR
The supplied character generator software allows you to create your own character set of up to 64 original characters.



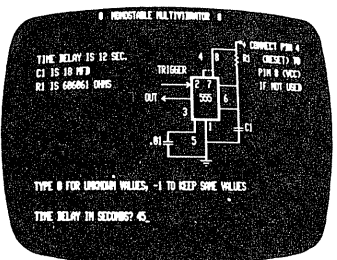
REAL-TIME GRAPHIC GAMES
With the 80-GRAFIX board you can write exciting real-time games using BASIC.



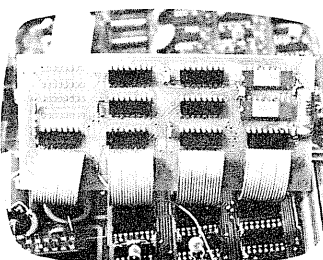
EASY INSTALLATION
The 80-GRAFIX board is simple to install and fits inside the TRS-80 case.



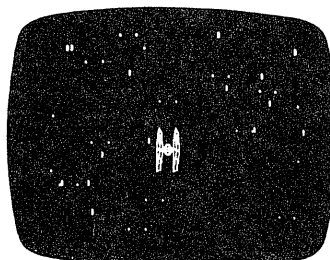
GRAPHICS GALORE
The 80-GRAFIX board and the supplied Character Generator allow you to become an artist.



ELECTRONIC DESIGN
The 80-GRAFIX board has unlimited application in Electronic design and Education.



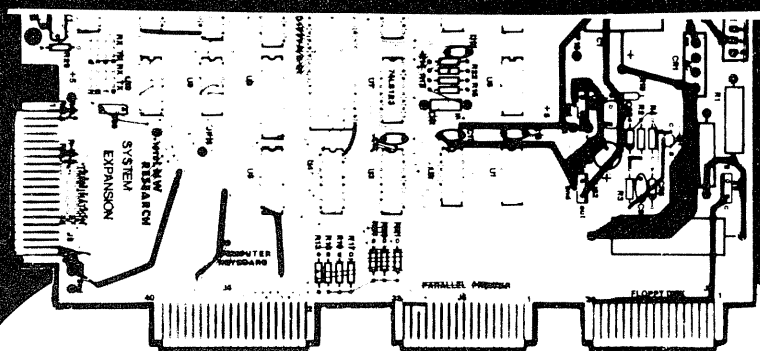
80-GRAFIX HI-RESOLUTION
Finally, the only means to protect your computer investment is to order an 80-GRAFIX board TODAY!



EXCITEMENT & FUN
Open up a new realm of software development with the 80-GRAFIX board.

Available exclusively through PROGRAMMA at the cost of \$149.95
Please check with us for availability prior to ordering
VISA and MASTERCARGE accepted
TRS-80 is a registered trademark of the Tandy Corp.

**PROGRAMMA
INTERNATIONAL, INC.**
3400 Wilshire Blvd.
Los Angeles, CA 90010
(213) 384-0579 • 384-1116 • 384-1117



**A Bare Circuit Board, Plug Compatible with the
TRS-80 Model I, Level II Computer**

LNW Research - PO Box 16216 - Irvine, CA 92714
\$59.95 - Includes 40-page Manual

Finally, there is a decent alternative to Radio Shack's TRS-80 Expansion Interface for those who like to build kits. LNW Research has produced and is selling a bare printed circuit board designed to supply all of the features of the Tandy Expansion Interface, but at a considerable savings in price.

For a mere \$59.95 you receive a naked (no parts) circuit board and two manuals, an assembly manual and a user manual. It is up to the purchaser to acquire the resistors, capacitors, IC's, cabinet, and the rest of the items needed to complete this kit. Even so, it is cheaper than Radio Shack's equivalent module.

The kit, when completely assembled, supplies all of the standard features of the Expansion Interface sold by Tandy; floppy disk and printer control circuits and ports, space for 16/32K additional RAM, and an on-board clock. You get all of these plus an RS-232 port.

This port can be set up for either serial interface signals or as a 20 ma current loop for driving a Teletype. If you want, you can even install a switch so you exercise the option of changing from one mode of operation to the other. Another difference between the two RS-232 ports is that

the interface controls, (baud rate, number of bits/words, etc.) of the LNW module are hardwired by the use of jumper wires and are not under software control. However, this is not really an inconvenience because most of the software on the market can be easily reprogrammed to use whatever settings you established in the LNW RS-232 Interface.

The circuit board is very cleanly laid-out, with all of the different sections clearly marked. The board even has the part numbers (as given on the schematics) to make it easier to get the right electronic components in the right spots.

The manuals are very complete, containing parts lists, work sheets, full schematics, assembly and operating instructions.

The manuals even tell you how to set up the parallel printer port to drive an RS-232 printer without sacrificing the use of your RS-232 port.

A note of caution is needed here: The manual assumes that the purchaser is a knowledgeable kit builder. It is *not* oriented towards the novice. However, this is not to say that a novice couldn't build it if he/she exercised a good deal of careful

reading and schematics part matching.

By far the best feature of this kit is its modularity. It is not necessary to build the entire Interface before it can be used. If you need only the extra 16 or 32K of RAM that an Expansion Interface gives you, then that is all that you need to build (you will also need to build the power supply filter, as well as plugging in the standard Radio Shack TRS-80 power supply). This means that you can add 32K of RAM to your computer for less than the price of a Radio Shack no-RAM Expansion Interface. A considerable savings. Likewise, if you need only the RS-232, then that is all you need build.

Later, when you get a line printer, or add disk drives to your system, just pull out the LNW kit assembly manual and add on the electronics needed to use that feature of the LNW board.

My LNW Expansion Kit has been completed and in operation for 6 months now, and of all the Expansion Interfaces I have come into contact with, it is definitely the most reliable one around.

All in all, the LNW Research Expansion Interface is the way to go if you like kit building and want more value for your dollar. **Yvon Kolya**



Med Systems!

Proven Educational Software

The Human Adventure allows movement through a human body's cardiovascular system. All major organ systems are accessible and fully described by the computer. A graphic CAT-scan constantly shows the user his position in the body. The exploration mode allows simple exploration, while the game mode places the user in a race against time to cure the patient of cancer using his knowledge of the body's layout. Recommended for reading age through adult.

The Playful Professor is a mathematics learning aid that provides tutoring in integer mathematics and fractions for the four basic operations. Demonstrated solutions are completed step-by-step in a blackboard format easily understood by grade school children. Problems are presented in a game format that places the pupil in a sixty room mansion. To win, the player must catch the ghost with the key, then get to the front door before the ghost (or other player) recaptures the key. Movement is based on problem solving. Difficulty may be different for each player, allowing parents to be beaten by their children. Recommended for age 4 through adult.

Money Master tutors the young child in the use of money. The child is allowed to wander freely by paying tolls or buying objects. The tutoring screen depicts money graphically, and interactively instructs in the use of coins. This includes making payments and receiving change. New mazes are generated for each game. Graphic obstacles are randomly chosen from a library of several dozen. An average game lasts 20-30 minutes. Recommended for early readers through adult.

Each program \$9.95 on cassette for TRS-80 Level II 16K, or Model III 16K. All three on diskette - \$29.95, Model I only.

Satisfaction Guaranteed! All Med Systems Software products come with a 14-day moneyback guarantee. If for any reason you are not satisfied, return your order within 14 days for a prompt and cheerful refund.

Ordering Information. Orders are processed within two working days. Mastercard and Visa card holders please remember to include the expiration date. We pay all postage and handling within the U.S., Canada, and U.S. territories. European orders please include \$2.00 for air post.

Med Systems Software

P.O. Box 2674 Department A58
Chapel Hill, North Carolina 27514
(919) 933-1990

Graphic 3-D Adventures

These machine language programs are the first in a new breed of adventure. Instead of wandering through the English language, typing GO EAST or GO WEST, you move through a colossal maze represented on the screen three-dimensionally. Hallways recede into infinity or come to dead-ends. Doors open to left and right. As you encounter objects, monsters, and mayhem, one or two word commands may be used. The command set is extensive and sophisticated. Movement is via the arrow keys. Graphics generation is instantaneous. Mazes are bit-coded and **HUGE**. There is simply nothing like these programs on the market today.

Deathmaze 5000 places you on the top floor of a five-story building. Each floor is a maze of twisting passageways. Floors are connected by elevators and open pits. You have but one goal. **Escape Alive!** Where is the only door out of this nightmare? Monsters, bats, mad dogs, hunger, and many more horrors plague your every step as you struggle to escape the most complex adventure ever written.

Labyrinth places you in a maze of gigantic proportions. But you are not alone! A minotaur searches for you, seeking a grisly meal. You must find weapons, spells, and treasures. You must deal with ghosts and cave gnomes. You must avoid the minotaur until the moment is right for the final battle. And if this isn't enough, the Labyrinth twists space and time so that you may not know whether you are coming or going!

Each program \$12.95 on cassette for TRS-80 Level II 16K, or Model III 16K. Both on diskette - \$29.95, Model I only.

ATTENTION DEATHMAZE FANATICS!

Still on the first level? You would look much better wearing the hat. But don't charge the wrong wall!

<input type="checkbox"/> Human Adventure	\$ 9.95	\$ _____
<input type="checkbox"/> Playful Professor	\$ 9.95	\$ _____
<input type="checkbox"/> Money Master	\$ 9.95	\$ _____
<input type="checkbox"/> Deathmaze 5000	\$12.95	\$ _____
<input type="checkbox"/> Labyrinth	\$12.95	\$ _____
<input type="checkbox"/> Educational Diskette	\$29.95	\$ _____
<input type="checkbox"/> Deathmaze/Labyrinth Diskette	\$29.95	\$ _____
	TOTAL	\$ _____

Name _____
Street _____
City _____ State _____ Zip _____

☐ MASTERCARD ☐ VISA ☐ Check
Mastercard or Visa # _____
Expiration Date _____



Slalom Run!

Greg Perry
Tulsa, OK

Slalom Run is an interactive game designed to prove that the TRS-80 need not have high resolution graphics and tremendous speed in order to be enjoyable.

I saw a similar program at a computer show several years ago when I did not know the first thing about computers, and that program was enough to get me hooked for life. It was easy to play, but difficult to become good at. It can be fun for kids (from 10 to 99 years of age!). Slalom Run will fit just about any TRS-80 (except Level I).

Features

The game begins with a skier at the top of a mountain. The skier is the graphics block at the top of the screen when the game starts. He starts down the hill and tries to get to the end of the run without hitting any obstacles. The obstacles are actually the letter "Y" which randomly appear beneath the skier.

The player must set a speed and experience level at the start of the game. The speed of the run is the length that the slope is run. If the user inputs an "S" for short run, the run will take about 20 to 25 seconds on the average. "L" for long slope will make the run about twice as long.

The experience level will be an important factor in determining if you

can make it down the slope. The experience level will set the speed of the skier. The race downhill will be quicker and it will be harder to dodge the pylons because more agility is needed. As a beginner, you might want to start with an experience level of 5 and decide after a few runs which level will be best for you. A word of warning - it is almost impossible to make it down the slope with an experience level of one. One is the hardest experience level and ten is the easiest.

About the Program

The playing field of "Y's" is initialized at the beginning of the game. These are 15 elements in a string array called A\$ and these 15 strings are 63 characters long. When the skier is running down the slope, the Y's are written to the screen by randomly selecting one of the above A\$ array elements and printing them at the bottom of the screen. This automatically scrolls the previously written Y's up a line. This is how the skier skies "down" the slope.

A check is made constantly in line 820 to see if the skier has moved since the last position. The more the skier moves, and the more he dodges the pylons, the better a rating he gets at the end of the game. This is to penalize some players who may happen to find a column on the screen that has very

few Y's in it. It takes more skill to move back and forth through the Y's and therefore more reward is given (totaled in the variable SC) at the end of the run.

The skier is moved back and forth by the left and right arrow keys. If a key is continuously pressed, the skier will keep moving.

The skier accelerates as he nears the end of the run just as in real skiing, so the player has to keep his eyes on the screen!


The score will appear at the end of the game and the option to play again is given. If a player runs into a pylon and wipes out, he is notified quite obviously by the program, and will have to start back at the top of the slope.

The program is designed for Level II and up, and as it stands it takes about 6.1K of memory. By removing the remarks and some of the instructions it should fit into 4K as well.

There is documentation for almost every section of code, and anyone who wants to change it should be able to follow the program logic with the remarks that are there. If there are too many Y's in the playing field for you, change line 280 to: P=RND(19).

Don't catch cold out there in the snow!

```
10 REM *** SLALOM
20 REM *** WRITTEN 6-10-1980
30 REM *** BY GREG M. PERRY
40 CLS
```



```

50 REM *** INITIALIZE VARIABLES
60 CLEAR 1200
70 O = 0 : SC = 0
80 DIM A$(15)
90 REM *** PRINT HEADINGS
100 PRINT CHR$(23)
110 PRINT : PRINT : PRINT
120 PRINT " S K I "
130 PRINT : PRINT " T H E "
140 PRINT : PRINT " T A N D Y "
150 PRINT : PRINT " M O U N T S ! ! "
160 REM *** PUT THE SKIING FIELD IN THE STRING ARRAY A
170 REM *** EACH ELEMENT IN THE TABLE A$ WILL BE FILLED
180 REM *** WITH 'Y'S RANDOMLY TO GIVE A DIVERSE PLAYING FIELD
190 FOR I = 1 TO 15
200 Z = 1
210 REM *** SINCE THIS VARIABLE INITIALIZATION TAKES A WHILE
220 REM *** THIS GOSUB WILL GIVE THE USER SOMETHING TO READ
230 REM *** WHILE THE DATA IS BEING CREATED
240 IF I = 4 THEN GOSUB 1070
250 REM *** IF YOU WANT MORE SPACES IN THE
260 REM *** PLAYING FIELD, CHANGE THE FOLLOWING
270 REM *** RND(13) TO A RND(19)
280 P = RND(13)
290 A$(I) = STRING$(P, " ") + STRING$((RND(2)), "Y") + A$(I)
300 IF RND(3) = 1 THEN A$(I) = "Y Y" + A$(I)
310 Z = Z + P
320 A$(I) = LEFT$(A$(I), 63)
330 IF Z > 63 THEN NEXT I ELSE GOTO 280
340 PRINT : PRINT
350 PRINT : PRINT : PRINT "PRESS ENTER TO CONTINUE" : INPUT E$
360 REM *** INSTRUCTIONS
370 CLS : PRINT : PRINT
380 PRINT "YOU WILL START AT THE TOP OF 'PANTY HOSE RUN'."
390 PRINT : PRINT "WHEN YOU FIRST START YOUR DESCENT, THE GOING
"
400 PRINT "WILL BE SLOW, BUT YOU WILL GRADUALLY SPEED-UP AS"
410 PRINT "YOU GAIN MOMENTUM. YOU HAVE GOT TO DODGE THE"
420 PRINT "TREES THAT YOU ENCOUNTER (Y'S) BY PRESSING"
430 PRINT "THE LEFT AND RIGHT ARROW KEYS."
440 PRINT "(ALWAYS KEEP ONE OF THE 2 ARROWS HELD DOWN AT ALL"
450 PRINT "TIMES TO DODGE THE TREES SMOOTHLY.)"
460 PRINT : PRINT "YOU WILL BE RATED AT THE END OF THE GAME."
470 PRINT "THE MORE YOU DODGE THE TREES, THE BETTER SCORE YOU"
480 PRINT "WILL HAVE, SO MOVE BACK AND FORTH AS MUCH AS YOU CAN
! ! "
490 PRINT TAB(34);STRING$(18, "=")
500 INPUT "PRESS ENTER TO CONTINUE";E$
510 CLS : PRINT : PRINT
520 PRINT "TO INITIALIZE THE SPEED FACTOR AND THE ROUGHNESS"
530 PRINT "OF THE RUN, TYPE IN YOUR EXPERIENCE FACTOR"
540 PRINT : INPUT "ENTER A 1 TO 10 (1 IS ADVANCED, 10 IS BEGIN
NER";E
550 IF E < 1 OR E > 10 THEN 540
560 PRINT : PRINT : INPUT "WOULD YOU LIKE A LONG RUN OR SHORT R
UN (TYPE 'L' OR 'S')";E$
570 IF LEFT$(E$, 1) = "L" THEN S = 2
580 IF LEFT$(E$, 1) = "S" THEN S = 1
590 IF LEFT$(E$, 1) <> "L" AND LEFT$(E$, 1) <> "S" THEN GOTO 560

600 PRINT : PRINT " -REMEMBER - THE MORE YOU MOVE, THE BETTER Y
OUR SCORE!-"
610 FOR I = 1 TO 800 : NEXT I
620 REM *** START PLAYING THE GAME
630 CLS
640 REM *** P IS THE STARTING POSITION OF THE SKIER
650 P = RND(5) + 30
660 FOR T = 1 TO 5
670 REM *** EVERY TIME THE LOOP OF T FALLS THROUGH THE PLAY
680 REM *** WILL SPEED UP A LITTLE. THIS WILL HAPPEN 5 TIMES
690 FOR T1 = 1 TO S * 35
700 REM *** THIS LONG INSTRUCTION IS REQUIRED TO GET ACCURATE
710 REM *** AND PROMPT RESULTS, PRINTING THE NEW POSITION OF
720 REM *** THE SKIER AS SOON AS ON OF THE ARROW KEYS

```

```

730 REM *** IS PRESSED.
740 IF PEEK(14400) = 32 THEN P = P - 1: PRINT @ P, CHR$(191); :
    PRINT @ P + 1, " "; : ELSE IF PEEK(14400) = 64 THEN P = P +
    1 : PRINT @ P, CHR$(191); : PRINT @ P-1, " "; : ELSE PRINT
    @ P, CHR$(191);
750 REM *** CHECK TO SEE IF THERE IS A TREE ('Y' = CHR$(89))
760 REM *** ABOUT TO HIT THE SKIER
770 IF PEEK(15424 + P) = 89 OR PEEK(15424 + P) = 25 THEN GOTO 11
    80
780 REM *** SC WILL BE INCREMENTED EVERY TIME THE SKIER
790 REM *** MOVES FROM THE LAST POSITION ('O'), SO
800 REM *** THIS WILL GIVE US AN IDEA OF HOW WELL HE DODGED
810 REM *** THE TREES
820 IF O < P THEN SC = SC + 1.3
830 REM *** KEEP THE SKIER OFF THE SIDES OF THE SCREEN
840 IF P < 4 THEN P = 4 ELSE IF P > 60 THEN P = 60
850 O = P
860 PRINT @ 960, A$(RND(15));
870 REM *** LINE 785 WILL SCROLL THE SCREEN BY ONE LINE
880 REM *** IN ORDER TO MAKE ROOM FOR THE NEXT LINE
890 REM *** TO BE PRINTED
900 PRINT@1023, " ";
910 FOR L = 1 TO (E+3) * (5-T) : NEXT L : NEXT T1 : NEXT T
920 REM *** THE SKIER HAS MADE IT THROUGH THE ENTIRE COURSE
930 CLS : PRINT : PRINT "YOU'VE MADE IT DOWN THE SLOPE !"
940 PRINT : PRINT "I'LL NOW COMPUTE A SCORE BASED ON HOW WELL"
950 PRINT "YOU DODGED THE TREES ON YOUR WAY DOWN."
960 FOR I = 1 TO 200 : NEXT I
970 IF E = 1 THEN SC = SC * 2
980 IF SC > 200 THEN SC = 200
990 REM *** FIGURE THE PERCENTAGE OUT OF 100 OR 200
1000 REM *** DEPENDING ON THE LENGTH OF THE SLOPE PICKED
1010 SC = SC / 200 * 100
1020 PRINT : PRINT : PRINT "YOUR SCORE IS";SC;"%"
1030 PRINT : IF SC < 50 THEN PRINT "YOU CAN DO BETTER THAN THAT
    !" : PRINT "YOU HARDLY MOVED THE ENTIRE RUN!" : GOTO 1060
1040 PRINT : IF SC < 75 THEN PRINT "PRETTY GOOD, BUT YOU'LL NEV
    ER MAKE THE OLYMPICS" : GOTO 1060
1050 PRINT : PRINT "GOOD JOB!! GIVE YOURSELF A GOLD MEDAL!!"
1060 FOR I = 1 TO 1300 : NEXT I : CLS : PRINT : PRINT : GOTO 13
    10
1070 REM ** PRINT TITLES
1080 CLS : PRINT : PRINT
1090 PRINT "WELCOME TO THE MOST ABSOLUTE, NOTHING-TO-BE-MATCHED
    "
1100 PRINT "      EXCITING DOWN-HILL SLALOM THIS SIDE OF THE"
1110 PRINT "      CANADIAN SLOPES !"
1120 PRINT : PRINT
1130 PRINT "NOW YOU CAN ENJOY THE THRILL OF THOSE IN & OUT DASH
    ES"
1140 PRINT TAB(8);"IN THE SAFETY OF YOU VERY OWN HOME !"
1150 PRINT "      (AND WITHOUT RISK OF A BROKEN LEG, TO BOOT!) "
1160 RETURN
1170 REM *** ACCIDENT REPORT
1180 CLS : PRINT CHR$(23)
1190 FOR T1 = 1 TO 4
1200 FOR I = 1 TO 9
1210 PRINT TAB( I * 3 ) "O W !"
1220 NEXT I
1230 FOR I = 9 TO 1 STEP -1
1240 PRINT TAB( I * 3 ) "O W !"
1250 NEXT I
1260 NEXT T1
1270 FOR I = 1 TO 500 : NEXT I
1280 CLS : PRINT : PRINT
1290 PRINT "WELL, THAT DIDN'T TURN OUT TOO WELL."
1300 PRINT : PRINT : PRINT "IF YOU THINK YOU HAVE THE HANG OF S
    KIING NOW,"
1310 PRINT "WHY DON'T YOU TRY IT AGAIN ?      IF YOU WANT TO, AL
    L"
1320 PRINT "THAT YOU HAVE TO DO IS ENTER 'YES', OTHERWISE 'NO'"
1330 INPUT "YOUR CHOICE ";E$
1340 IF LEFT$(E$,1) = "Y" THEN GOTO 510
1350 PRINT : PRINT "THANKS, ANYWAY. I HOPE YOU ENJOYED SKIING!"
1360 END

```



MICRO CLUB CHARTER MEMBERSHIPS

How would you like to be the first to obtain new applications for your TRS-80® and save money?

Then the MICRO CLUB is for you!

A vast array of software, especially games, exists for your microcomputer, but software alone can only partially fulfill the power and usefulness of your computer.

You probably own a disk and a printer--hardware which also enhances the usefulness of your computer. Other than these common peripherals, relatively little hardware exists in comparison to software, and most of the hardware available is prohibitively expensive.

The **MICRO CLUB** has been established to meet this need for a variety of inexpensive hardware devices--tools which will help you get more satisfaction from your computer.

Each month the **MICRO CLUB** will announce one or more new hardware products complete with complementary software and comprehensive documentation. Some products to be offered in the coming months include a *DATE/TIMER event timer*, an *energy efficiency MEASURING INSTRUMENT*, and a *PROM BURNER*. Best of all, **MICRO CLUB** members will usually pay less than \$100 for products which eventually will retail for \$150 or more.

Membership in the **MICRO CLUB** costs \$10 for 12 issues of our new products newsletter describing our product of the month as well as other hardware and software specials available exclusively to **MICRO CLUB** members. Even the membership fee is fully refundable, i.e.: you will receive a \$10 discount on your first selection. Although there's no obligation to purchase any product, you need make only two selections during the 12 month period to have your membership automatically extended for another year.

The **MICRO CLUB** will make an excellent present. Simple fill in the coupon and mail to **MICRO CLUB**.

Please register me as a Charter Member of the Micro Club and send me newsletters describing your next 12 monthly selections. I understand that my membership fee of \$10 will be refunded against my first selection and that I need only make two selections a year to remain a member.

Mail To:
MICRO CLUB
4401 S. Tamarac Parkway
Denver, CO 80237

Micro Club is an affiliate of Apparat, Inc., specializing in hardware and software.

NAME _____
ADDRESS _____ STATE _____ ZIP _____
PROCESSOR: TRS-80® _____ APPLE® _____
_____ CHECK ENCL.(\$10) VISA _____ MASTERCHARGE _____
CARD NO. _____ EXP. DATE _____

For additional memberships, send along the above information on a separate sheet. For further details CALL 303-741-1778.

Estimating Program

T R Dettmann
Associate Editor



**Estimate the man-days
required to write a program.
(With a program!)**

Believe it or not, programming is a business too! You would be surprised at how many people who are doing it for money aren't treating it like a business.

Most of the business computer tools are pretty standard: Receivables, Payables, Ledger, etc., even for a programming business. But how do you estimate a job?

Have you ever gone out to a business, been given detailed specs on a job you want to do, and then been asked for an estimate? You probably have if you are trying to program for pay. What's more, you probably wound up taking a figure off the top of your head and saying: "Here's my estimate."

It almost seems as if it's a national pastime for businessmen to put programmers on the spot. Most occupations that require job estimates have some sort of detailed program estimating technique. Carpenters, bricklayers, all of them, have estimating aids. Now it's time for the programmer to have such an aid.

The program in this article grew out of a reading (and multiple rereadings) of the book *The Program Development Process (Part 1 - The Individual Programmer)* by Joel D. Aron. You probably can't find this book outside of a major computing center library, but it is part of a well known series of books called the *Systems Programming Series*, published by Addison-Wesley.

The series is sponsored by IBM and is aimed mostly at large system programmers. It includes volumes on Programs, Programming, Data bases, Compilers, Interactive Graphics, Sorting, and Recursive Programming. In all, there are 11 books in the series.

In the program development process, Aron goes over the whole process of developing programs,

This program works without modification on Models I and III. It does not require a printer. Model II users can also run this program if the following changes are made:

Change line 1210 to read: PRINT@1700, "PRESS ENTER TO END, (ESC) TO RE-RUN"

Change line 1220 to read: C\$=INKEY\$:IF C\$=" " THEN 1220 ELSE IF ASC(C\$)=&H1B THEN 230 ELSE IF ASC(C\$) <> 13 THEN 1220

emphasizing what the individual programmer should do in order to work most efficiently. Chapter 3 deals with problem analysis and planning.

When I first read the book over a year ago, I was quite impressed by the planning chapter, in particular by the attempt to provide the programmer with some real numbers for estimating the time it would take to do a job.

As the book points out, there is no real agreement about what these numbers really should be, but several tables are provided that are taken from published studies of the program development process. These tables form the basis for the program listing with this article.

Estimating a Job

In order to estimate a job, it is necessary for the programmer to first estimate the difficulty of various parts of the job. The book (and the program), first breaks the job down into its attributes, starting with Input-Output.

Four characteristics of the Input and Output are measured and assigned appropriate weights. They are:

1. The number of record types of FIXED FORMAT. That is, the number of *different types* of records of information that do not change in format.

2. The number of VARIABLE FORMAT records. These are records whose character or size changes during the program.

3. The number of commands, messages, or inquiries that the program will have to handle. For example, how many commands can be entered from the keyboard to control execution?

4. The number of special devices used for the program. By special devices, we mean things which are not normal connections to the standard model computer system (burglar alarms, etc.).

Next the program gets information about storage requirements for the program:

1. The number of arrays that will be used by the program.

2. The number of files that will be used during the operation of the program.

3. The number of files that will be used that have a special structure. For example, how many files are stored as linked lists?

4. The number of multiple file relationships that the program will have to handle. For example, how many files will have information moved from one to another during program execution.

“Have you ever gone out to a business, been given detailed specs on a job, and then asked for an estimate?”

Now we have to find out what the processing objectives are for the program:

1. The program asks for a number from 0 to 10 to indicate the Real-Time performance objectives. A program that would be expected to give immediate displays of any item in an inventory would rate a higher degree of difficulty than one that allows a search for the item before displaying it.

2. The objectives for data communications are requirements for computer-to-computer or computer-to-terminal information transfer. Normally, this will be a zero.

3. The importance of graphics displays is next. Obviously, graphics are very important to a game, but far less important to most business programs.

Still under the processing information, the book assigns a whole table to determining the effect of the choice of a language on the program. Clearly, if you choose to program in Assembly Language, it will take you longer to code the program than if you chose to do it in a high level language (Unless you are a super-programmer).

The degree of difficulty though, will depend on the nature of the program and the assembler or interpreter you use. To account for that, the book (and the program) assigns weights based on whether the program is prepared in Assembly Language, a high level language, or from existing modules. In addition, weights are assigned based on how difficult that language is to use.

Next the program asks for your own qualifications as a programmer and your knowledge of the specific job you are going to program. The length of time needed to complete the job is obviously much greater for a trainee than for a senior programmer (6 times greater by the weights given in the book).

Once all of the questions are asked, the program will then display the total time in Man-Days to complete the job and the suggested breakdown of days to complete it. The breakdown is based on the complexity of the program. For a very complex program, more time is assigned to design on a percentage basis.

Some items in the book's tables were left out of the program intentionally (such as shared arrays) since they aren't used much in microcomputer programming. This doesn't affect the final results since the weights for these items would have been zero for a typical project.

Over the last year I have compared the output from the program to the results of actual jobs (I won't say what kind of programmer I said I was!). Amazingly, even without modifying the weights from the book, the estimates I got came out close to the actual time required to do a project.

In one case, I completed a project with approximately two man-months of effort that the program estimated 60 man-days for! I can't argue with that.

Despite the agreement in my own tests, this program can hardly be considered a really accurate estimate for all cases. It does however give the programmer a definite feel for the length of time needed to complete a program and forms the basis for a reasonable estimate for estimating a job. It can also make an impressive display for a customer when you use his system to estimate the program development time in front of his eyes!

(Verification - if you answer "1" to every question in this program, these are the results you should get.)

JOB ESTIMATE

THE ESTIMATE FOR THIS JOB IS 15.5 MAN DAYS.

THESE DAYS WILL BE SPLIT UP AS FOLLOWS:

5.425 DAYS FOR DESIGN

3.875 DAYS FOR CODING AND DEVELOPING TEST DATA

5.425 DAYS FOR DEBUGGING

.775 DAYS PREPARING DOCUMENTATION.

```

10 REM *** PROGRAMMING JOB ESTIMATING ***
20 REM VERSION 1.0 04/19/80 ***
30 REM (C) 1980 BY TERRY R. DETTMANN ***
40 REM ***
50 REM ***
60 REM ***
70 REM ***
80 REM ***
90 REM ***
100 REM ***
110 REM ***
120 REM ***
130 REM ***
140 REM ***
150 REM ***
160 REM ***
170 REM ***
180 REM ***
190 REM ***
200 REM ***
210 REM ***
220 REM ***
230 REM ***
240 REM ***
250 REM ***
260 REM ***
270 REM ***
280 REM ***
290 REM ***
300 REM ***
310 REM ***
320 REM ***
330 REM ***
340 REM ***
350 REM ***
360 REM ***
370 REM ***
380 REM ***
390 REM ***
400 REM ***
410 REM ***
420 REM ***
430 REM ***
440 REM ***
450 REM ***

: DIM PD!(3,4),PEW!(14),LC!(3,4,6),EL!(4),UN!(3,5)
: REM ARRAYS:
90 REM PD! PROGRAM DEVELOPMENT ACTIVITIES
100 REM PEW! PROGRAM UNIT ESTIMATING WEIGHTS
110 REM LC! LANGUAGE CAPABILITY WEIGHTS
120 REM EL! PROGRAMMER EXPERIENCE WEIGHTS
130 REM UN! JOB UNIQUENESS WEIGHTS
140 REM
150 REM READ THE WEIGHTS FROM DATA STATEMENTS
160 FOR I=1 TO 3: FOR J=1 TO 4: READ PD!(I,J): NEXT J, I
170 FOR I=1 TO 14: READ PEW!(I): NEXT I
180 FOR I=1 TO 3: FOR J=1 TO 4: FOR K=1 TO 6: READ LC!(I,J,K): NEXT K, J, I
190 FOR I=1 TO 4: READ EL!(I): NEXT I
200 FOR I=1 TO 3: FOR J=1 TO 5: READ UN!(I,J): NEXT J, I
210 REM INITIALIZE THE DAYS TO ZERO & DEFINE GRAPHICS LINE
220 DAYS! = 0
230 HDR$=STRING$(60,140)
240 REM ----- BEGIN ESTIMATE -----
250 REM WHAT ARE THE I/O CHARACTERISTICS OF THE PROGRAM?
260 CLS:PRINTHDR$:PRINTTAB(20)"ESTIMATING":PRINTHDR$
270 PRINTTAB(5)"I/O CHARACTERISTICS"
280 PRINTTAB(10):INPUT"NUMBER OF FIXED FORMAT RECORDS":N
290 DAYS! = PEW!(1)*N + DAYS!
300 PRINTTAB(10):INPUT"NUMBER OF VARIABLE FORMAT RECORDS":N
310 DAYS! = DAYS! + PEW!(2)*N
320 PRINTTAB(10):
: INPUT"NUMBER OF COMMANDS, MESSAGES, AND INQUIRIES":N
330 DAYS! = DAYS! + PEW!(3)*N
340 PRINTTAB(10):INPUT"NUMBER OF SPECIAL DEVICES":N
350 DAYS! = DAYS! + N*PEW!(4)
360 REM WHAT ARE THE STORAGE REQUIREMENTS FOR THE PROGRAM?
370 CLS:PRINTHDR$:PRINTTAB(20)"ESTIMATING":PRINTHDR$
380 PRINTTAB(5)"STORAGE"
390 PRINTTAB(10):INPUT"NUMBER OF ARRAYS USED":N
400 DAYS! = DAYS! + N*PEW!(5)
410 PRINTTAB(10):INPUT"NUMBER OF FILES TO BE USED":N
420 DAYS! = DAYS! + N*PEW!(6)
430 PRINTTAB(10):
: INPUT"NUMBER OF FILES WITH LIST STRUCTURES OR OVERFLOWS":N
440 DAYS! = DAYS! + N*PEW!(7)
450 PRINTTAB(10):
: INPUT"NUMBER OF MULTIPLE FILE RELATIONSHIPS":N

```

```

450 DAYS! = DAYS! + N*PEW!(8)
470 REM WHAT LEVEL OF PROCESSING DIFFICULTY WILL THE PROGRAM HAVE?
480 CLS:PRINTHDR$;PRINTTAB(20)"ESTIMATING":PRINTHDR$
490 PRINTTAB(5)"PROCESSING"
500 PRINTTAB(10)"ENTER A NUMBER IN THE RANGE REQUESTED FOR EACH"
510 PRINTTAB(10)"QUESTION ASKED":PRINT
520 PRINTTAB(10);:INPUT"REAL-TIME PERFORMANCE OBJECTIVES (0-10)";N
530 DAYS! = DAYS! + N
540 PRINTTAB(10);:INPUT"DATA COMMUNICATIONS (0-5)";N
550 DAYS! = DAYS! + N
560 PRINTTAB(10);:INPUT"GRAPHIC DISPLAYS (0-10)";N
570 DAYS! = DAYS! + N
580 REM WHAT ARE THE CAPABILITIES OF THE PROGRAMMING LANGUAGE BEING USED?
590 CLS:PRINTHDR$;PRINTTAB(20)"ESTIMATING":PRINTHDR$
600 PRINTTAB(5)"LANGUAGE CAPABILITY"
610 PRINTTAB(10);:PRINT"(1) ASSEMBLY LANGUAGE OR (2) HIGH LEVEL LANGUAGE"
620 PRINTTAB(10);:INPUT"OR (3) PACKAGES";L
630 IF L=3 THEN 750
640 PRINT
650 :PRINTTAB(10)"FOR EACH OF THE FOLLOWING QUESTIONS":
660 :PRINTTAB(10)"ANSWER (1) EASY (2) MEDIUM (3) HARD (4) VERY HARD"
670 PRINTTAB(15);:INPUT"RESTRUCTURING DATA";N
680 DAYS! = DAYS! + LC!(L,N,1)
690 PRINTTAB(15);:INPUT"MONITOR STATUS";N
700 DAYS! = DAYS! + LC!(L,N,2)
710 PRINTTAB(15);:INPUT"RETRIEVE AND PRESENT DATA";N
720 DAYS! = DAYS! + LC!(L,N,3)
730 PRINTTAB(15);:INPUT"CALCULATIONS";N
740 DAYS! = DAYS! + LC!(L,N,4)
750 PRINTTAB(15);:INPUT"PROGRAM LINKAGE";N
760 DAYS! = DAYS! + LC!(L,N,5)
770 GOTO840
780 PRINTTAB(5)"PACKAGE PROGRAMMING"
790 PRINTTAB(10)"WILL YOU USE A (1) PROGRAM PACKAGE, (2) UTILITY"
800 PRINTTAB(10)"PROGRAM OR (3) A REPORT PROGRAM GENERATOR";:
810 INPUT N1
820 PRINT:PRINTTAB(10)"WILL YOU (1) PREPARE CONTROL CARDS ONLY"
830 PRINTTAB(10)"OR (2) ADD YOUR OWN CODE";:INPUT N2
840 PRINTTAB(10)"IS THIS PROJECT (1) EASY (2) MEDIUM (3) HARD"
850 PRINTTAB(10)"OR (4) VERY HARD";:INPUT N3
860 DAYS! = DAYS! + LC!(L,N3,2*(N1-1)+N2)
870 REM - - - - - KNOW HOW - - - - -
880 REM HOW GOOD IS THE PROGRAMMER?
890 CLS:PRINTHDR$;PRINTTAB(20)"ESTIMATING":PRINTHDR$
900 PRINTTAB(5)"KNOW HOW"
910 PRINTTAB(10)"WHAT IS YOUR EXPERIENCE LEVEL?"
920 PRINT
930 PRINTTAB(15)"1. SENIOR PROGRAMMER"
940 PRINTTAB(15)"2. PROGRAMMER"
950 PRINTTAB(15)"3. APPRENTICE"
960 PRINTTAB(15)"4. TRAINEE"
970 PRINT:PRINTTAB(10);:INPUT"LEVEL";EX
980 REM - - - - - HOW UNIQUE IS THE JOB? - - - - -
990 CLS:PRINTHDR$;PRINTTAB(20)"ESTIMATING":PRINTHDR$
1000 PRINTTAB(5)"JOB UNIQUENESS"
1010 PRINTTAB(10)"HOW MUCH JOB KNOWLEDGE IS REQUIRED?"
1020 PRINTTAB(10)"(1) SOME (2) SOME (3) NONE";:INPUT N1
1030 PRINTTAB(10)"HOW MUCH DO YOU HAVE AVAILABLE?"
1040 PRINTTAB(15)"1. DETAILED KNOWLEDGE"
1050 PRINTTAB(15)"2. GOOD GENERAL WITH FRAGMENTARY DETAILED KNOWLEDGE"
1060 PRINTTAB(15)"3. FAIR GENERAL WITH LITTLE OR NO DETAILED KNOWLEDGE"
1070 PRINTTAB(15)"4. NO DETAILED BUT GENERAL KNOWLEDGE OF RELATED"
1080 PRINTTAB(15)"5. NO DETAILED OR GENERAL KNOWLEDGE"
1090 PRINTTAB(10);:INPUT"KNOWLEDGE LEVEL";N2
1100 EW! = EL!(EX) + UN!(N1,N2)
1110 MD! = EW!*DAYS!
1120 PRINT:PRINTTAB(10)"WHAT KIND OF LOGIC IS INVOLVED IN THIS PROGRAM"
1130 PRINTTAB(10);:INPUT"(1) AVERAGE (2) COMPLEX (3) COMPLEX CONTROL";LG
1140 REM - - - - - DISPLAY ESTIMATE - - - - -
1150 CLS:PRINTHDR$;PRINTTAB(20)"JOB ESTIMATE":PRINTHDR$
1160 PRINTTAB(10)"THE ESTIMATE FOR THIS JOB IS ";MD!;
1170 "MAN DAYS"
1180 PRINT
1190 PRINTTAB(10)"THESE DAYS WILL BE SPLIT UP AS FOLLOWS:"
1200 PRINTTAB(15)MD!*PD!(LG,1);"DAYS FOR DESIGN"
1210 PRINTTAB(15)MD!*PD!(LG,2);"DAYS FOR CODING AND DEVELOPING TEST DATA"
1220 PRINTTAB(15)MD!*PD!(LG,3);"DAYS FOR DEBUGGING"
1230 PRINTTAB(15)MD!*PD!(LG,4);"DAYS PREPARING DOCUMENTATION"
1240 PRINT
1250 PRINTTAB(5)"PRESS ENTER TO END, CLEAR TO RERUN"
1260 C#=INKEY$:IF C#="" THEN 1220 ELSE IF ASC(C#)=31 THEN 220 ELSE IF ASC(C#)<13 THEN 1220
1270 END
1280 REM - - - - - ESTIMATING WEIGHT DATA - - - - -
1290 DATA .35,.25,.35,.05,.40,.20,.35,.05,.35,.20,.40,.05
1300 DATA 1,2,1,1,1,3,5,2,0,0,0,1,2
1310 DATA 3,3,2,2,0,4,5,4,3,3,0,5,7,6,5,4,0,6,9,8,7,5,0
1320 DATA 1,1,1,1,0,2,2,2,2,2,0,3,4,4,3,3,0,4,6,6,4,4,0
1330 DATA 2,2,1,2,1,2,3,4,1,4,1,4,3,8,1,4,1,8,3,16,1,4,1,8
1340 DATA 0.5,1.0,1.5,3.0
1350 DATA 0,.25,.5,.75,1.0,0,.25,.5,.75,0,0,.25,.25,.25,.25

```

Goodies from GALACTIC

Specialty Programs for TRS-80 Model I - II - III

EDAS 4.0 (Editor/Assembler)

This is the highly acclaimed "USER ORIENTED" Assembler for the TRS-80 Model II by GALACTIC. Loaded with features such as assemble to memory, block move, link to debugger, default filenames, reverse video editing, warm start entry and much more. Now the programmer can write, assemble, test, and debug his code without ever leaving EDAS.

EDAS 4.0 with complete manual (120 pages)

Model II Version Was \$229.00

NOW ONLY \$179.00

MASS/MAIL SYSTEM

This is the NAME and ADDRESS system for subscription control or large mailing lists. It will handle up to 10,500 records, with a worst access time of less than 15 seconds and usual access of less than one second. All adds, deletes, and edits are instant for the operator and are then completed later in a "batch monitor". Extensive documentation and ongoing support. Requires TRS-80 Model II and 2 disk drives minimum. Contact GALACTIC direct for detailed specifications and prices for your exact needs.

Model II Version Contact GALACTIC for Price

STOCK MARKET MONITOR

This day to day market monitor is designed for the active trader. The system will track the performance of an issue against the market as well as against itself. The package comes with complete documentation and explanations of the formulas that are used by the program. The system is available for the Model I and the Model III TRS-80.

Model I and III cassette version \$89.00

Model I and III disk version \$99.00

INVENTORY MASTER

Tired of being a slave to an out-of-control inventory? Let GALACTIC'S INVENTORY MASTER put you in control of your inventory. INVENTORY MASTER operates on a TRS-80 Model I and Model III 48K disk system (Minimum of 2 drives with capabilities of up to 4 drives). Drive spanning capabilities allow you to track 2700 inventory items with a 4 drive system (5100 items for the Model III). Unique machine language sort allows for instantaneous item insertion (approx. 15 seconds with 2700 items in system). Item access can be immediate using system-supplied control numbers. Modeled after a proven main-frame system costing tens of thousands of dollars. Complete add/edit/delete capabilities supported. Placement of orders can be machine-generated as well as user-generated, with editing capabilities. Full report-generator included. Exquisitely documented.

Model I Version \$159.00

Model III Version \$259.00

MODEL II HOST I/O SYSTEM

From the original author of the TRS-80 HOST and TERM systems in the RADIO SHACK "COMMUNICATIONS PACKAGE". This system allows the full control of the HOST facility by your BASIC program. Set the number of nulls to be sent after a C/R, set a command line to be executed if carrier is lost, turn HOST on and off, switch to channel A or B as desired, enable and disable the ability for the remote terminal to "BREAK" BASIC, identify whether a character came from the HOST'S keyboard or from the REMOTE'S and more. No knowledge of assembler needed. All options may be accessed from BASIC or ASSEMBLER. Complete with detailed documentation. Don't isolate your Model II, Let outside terminals access it's computing power.

Model II with TRSDOS 1.2 \$179.00

Model II with TRSDOS 2.0 \$199.00

MAIL/FILE SYSTEM

This is the name, address, phone number data base manager that has set the standard by which other systems are compared. This system contains advanced editing and output capabilities. The TRS-80 Model I system will handle up to 600 records per file, while the Model III version will handle up to 1150 records and the Model II will handle 2500 records per file. All versions are file compatible and maintain constant sort indexes on both NAME and ZIP CODE. International PHONE numbers and ZIP CODES are supported. Thousands of code combinations are available. The Model II version also has a "word processor" type input editor and fast assembler sorting. Complete documentation is included with each version of MAIL/FILE.

Model I Version \$ 99.00

Model III Version \$149.00

Model II Version \$199.00

ULTRA TREK

This is an all new concept for this type of game, and compares to the others like chess compares to checkers. ULTRA-TREK is a complex, logical game, intended for the serious contestant. It is doubtful that you will ever master this game, but you will certainly enjoy trying! This program requires a TRS-80 Level II, 16K or more. The program is written totally in BASIC and uses 15.5K of RAM.

Model I & Model III Version

(cassette only) \$14.95

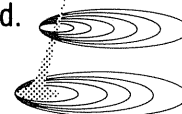
galactic software ltd.

A Division of GS & WS, Inc.

11520 N. Port Washington Rd.

Mequon, Wisconsin 53092

(414) 241-8030



Money Orders & COD's Shipped Within 24 Hours. Checks allow 2 weeks.

Now Inexpensive Word Processing with Starwriter Daisy Wheel Printer



\$1795

Starwriter/Starwriter II Daisy-Wheel Printer

With a print speed of 25 cps, the Starwriter incorporates the latest LSI technology and employs an industry-standard 96-character print wheel—an easy-to-change format that readily accommodates multi-lingual applications. Starwriters produce letter-quality printing on three sharp copies with either 136 columns or 163 columns. Self-test capability, programmable VFU (vertical format unit) and front panel indicator lamps for paper, select and power status. Compatible with sheet-feeders and can handle up to 15 inches wide. Tractor optional.

- ★ Bi-Directional
- ★ Logic seeking
- ★ 136 printable columns
- ★ 25 CPS printspeed
- ★ Elite pitch
- ★ Pica pitch

- ★ 40 msec. line feed time
- ★ Uses standard Diablo ribbon
- ★ 1/48 inch minimum line spacing
- ★ Standard parallel or RS232-C* compatible
- ★ 115V ± 10%, 50/60 Hz, 70 W power requirements

*Add \$60.00 for Serial

Ask about Starwriter II 45 cps



LORDS

Small Systems Design
P.O. Box 99
Port Angeles, WA 98362
Telephone (206) 457-3064

DOSPLUS 3.1

1.5 MEG on MODEL 1 Increase Your Disk Storage!!!

Package 1 - 1.5 Megabytes for only \$1,540.00

- 2: Model 160-2 AEROCOMP 80-track double headed drives
- 1: PERCOM DOUBLER
- 1: DOSPLUS 3.1D
- 1: 2 Drive Cable (for the AEROCOMP drives)

Package 2 - Upgrade your Model I to a Model III for only \$320.00

- 1: PERCOM DOUBLER
- 1: DOSPLUS 3.1D

Package 3 - Add on disk storage and go double density for only \$1,040.00

- 2: Model 40-1 AEROCOMP 40-track "floppy" drives
- 1: PERCOM DOUBLER
- 1: DOSPLUS 3.1D
- 1: 4 drive cable (for the AEROCOMP drives)

Package 4 - Add an 80-track drive and double density for only \$800.00

- 1: Model 80-1 AEROCOMP 80-track "floppy" drive
- 1: PERCOM DOUBLER
- 1: DOSPLUS 3.1D
- 1: 3 drive cable (for the AEROCOMP drive)

Remember, ONE 80-track drive, running double density, will give you as much storage as FOUR 35-track drives running single density.

If interested, call or write us at the address below, and let us banish your disk storage problems forever!

MICRO SYSTEMS SOFTWARE INC.

5846 Funston Street, Hollywood, FL 33023
(305) 983-3390

NAME _____

ADDRESS _____

CITY _____

STATE _____ ZIP _____

PHONE _____

ACCOUNT # _____

MC ☐ VISA ☐ EXP. DATE _____

PROGRAM NAME _____

QUANTITY _____

AEROCOMP DISC DRIVES FOR TRS — 80*

- **MODEL 80-1 DISC DRIVE \$449.95 ea.**
Single-sided, "floppy", 96TPI
(80 track; single density
unformatted 250K bytes/side;
double density unformatted 500K
bytes/side).
- **MODEL DISC DRIVE 160-2 \$599.95**
Double-sided, 96TPI.
(160 track/80 per side; single
density unformatted 500K
bytes; double density
unformatted 1 megabyte).

All models are capable of single or double density and are complete with power supply and silver enclosure. Send for information on AEROCOMP 2-and 3-drive systems available in 40 and 80 track.

The Doubler™: Percom's new proprietary double-density adapter for the TRS-80* computer.

- Increase formatted storage capacity of your minidiskettes from 1 1/2 to almost 4 times.
- Use with standard 5-inch drives rated for double-density operation.
- The DOUBLER™ reads, writes and formats either single - or double density disks. Only \$219.95.

NEW FROM MICRO-SYSTEMS!!!

Micro-Systems Software Inc. now has double density software available for TRS-80* Model I's that are equipped with the Percom Doubler. +

First is a disk editor called "Disk Zap 2.3". This editor will work either single or double density diskettes. It is track and sector oriented, and offers total access to all parts of the disk. It has the ability to format and backup diskettes as well as editing them.

Second is our new double density DOS. DOSPLUS 3.1D, like our regular DOS, will run 35-80 track drives; but offers the increased disk storage of double density.

Disk Zap is \$19.95 and DOSPLUS 3.1 or 3.1D is \$99.95. To order, call or write us at the address below. Master Card and Visa welcome. Orders accompanied by a personal check will be shipped when the check clears the bank.



**Micro-Systems
Software Inc.**

Specializing in the Tandy Line



(305) 983-3390

5846 Funston Street
Hollywood, FL 33023

* TRS - 80 is a trademark of Tandy Corp.
+ Doubler is a trademark of Percom Data Corp.
Now available in Spanish

Panattoni's Panacea

Build your own Printer Interface

Larry S Panattoni

In the last issue I removed the cover from the Radio Shack Expansion Interface, and revealed the fact that it contains several different sections; each of which I will discuss separately; explaining the circuitry and its operation. I will then show how each section can be constructed to operate independently.

Last issue the "line printer" interface circuitry was explained. In this issue I will show how it can be constructed separately. This will be of use for those who have a printer and do not wish to purchase an entire interface unit at this time. It is primarily designed to interface with a Radio Shack printer, such as the Centronics; but the pin connections are semi-standard within the industry and will work on several different makes. However, if you have a different make you should check the schematic of your printer before wiring the pin connections.

Address Decoding & Operation

A couple of items needed in order to allow the circuitry presented last issue to function as an independent unit are an address decoding circuit and a power source.

Figure 1 shows the decoding circuitry which will decode the address of the printer, (37E8 Hex), for both Read and Write operations. The read operation is used to enable the S-80 to obtain status information about the printer - mainly if it is busy or not. And naturally, the write operation is used by the S-80 to send data to be printed.

This printer interface can be

connected directly to the S-80 40-pin bus connector on the back of the keyboard; where the address leads are picked up from the various pin locations as shown in Figure 1.

Without going into a lot of detail, the operation of the decoding circuitry is such that when the address "37E8" is addressed, and the S-80 applies a low on the RD lead (pin 15), IC4-D becomes enabled and outputs a low read pulse, which is sent to Figure 2 as shown.

A similar operation occurs when a write operation is performed. In this case the S-80 applies a low on the WR lead (pin 13) which combined with the address 37E8, enables IC4-C to output a low write pulse, which again is sent to Figure 2.

Figure 2 is somewhat the same configuration as that used in the expansion interface as discussed last issue. The pin connections have been modified to allow connection to the S-80 keyboard and a couple of extra features were added.

To generalize its operation: A low read pulse from Figure 1 enables IC5 (74LS367), a Hex buffer, allowing the data lines to read the condition of the "Out of Paper" and "Busy" signals from the printer. On a write operation the low pulse from Figure 1 loads the 8 bits of data from the data bus into latches IC6 and IC7 (74LS175). This data is then held by the latches until they are strobed into the printer buffer by the 1.5 microsecond strobe pulse from IC8 (74LS123).

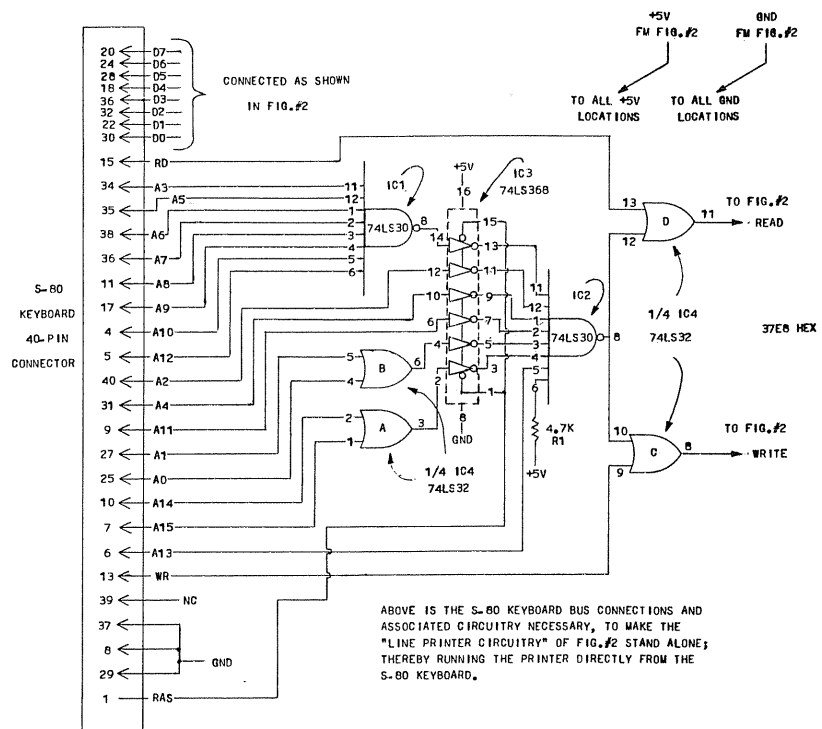


Figure 1

Figure 2

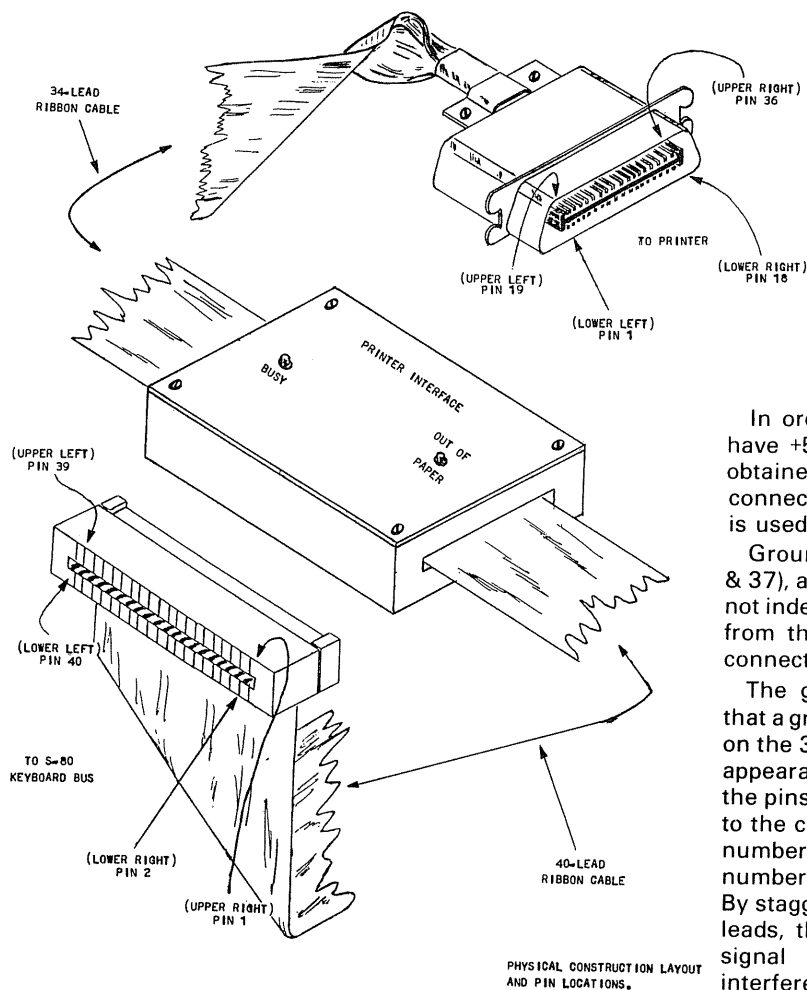
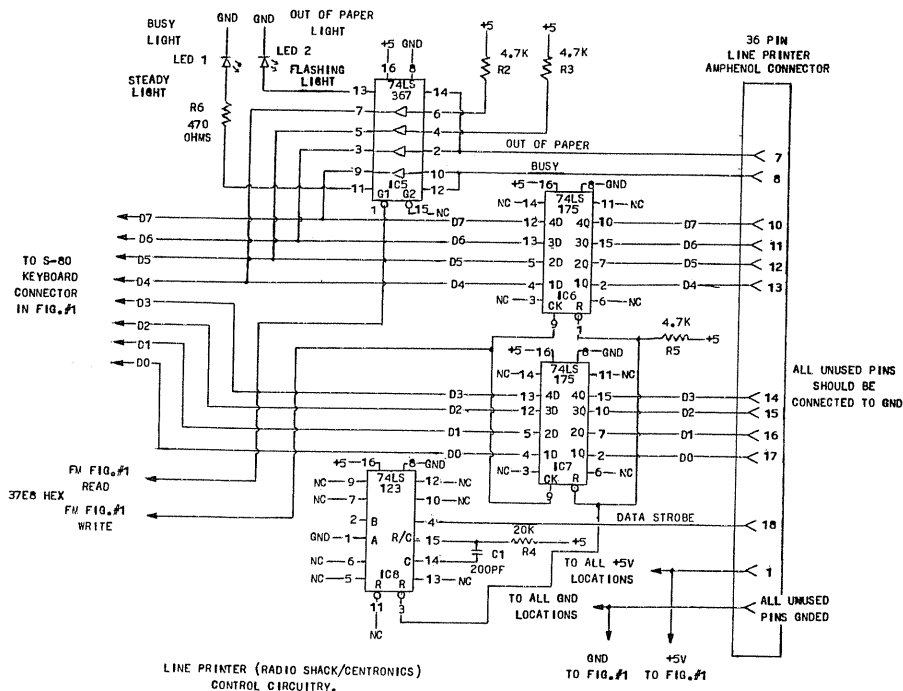


Figure 3

Power Source

In order for this "Printer Driver" to function, it must have +5 Volts to Power the IC's. This +5 Volts can be obtained from the printer itself. The Centronic's printer connector has +5 volts on Pin 1, as indicated in Figure 2. It is used to power all the IC's in Figures 1 and 2.

Ground is obtained from both the S-80 Bus (pins 8, 29, & 37), and the printer connector. All pins on the connector not identified in Figure 2 are grounded. The ground leads from the S-80, the printer, and Figure's 1 and 2, are connected together to provide a common ground.

The ground leads from the printer are assigned so that a ground lead will be placed between each signal lead on the 34 lead ribbon cable. Figure 3 shows the physical appearance of the printer connector and identifies how the pins are numbered. When wiring the 34 ribbon cable to the connector; the first lead goes to the lower left pin number 1; the second lead will go to the upper left pin number 19; the third to the second from the lower left; etc. By staggering the upper and lower pins with the adjacent leads, the ground leads automatically fall between the signal leads, providing shielding and preventing interference with the data signals.

Added Features

Figure 2 has two LED indicators; 1.) A "Busy" condition indicator (steady light) and 2.) an "Out of Paper" condition indicator (flashing light). Even though these may not be necessary they can be quite useful.

You may have experienced a time when your computer was printing something, then seems to hang up with nothing being printed on paper or screen. At first you wonder if you had a power hit and lost your program, or if your printer broke down. After a few moments of investigation, you found you were out of paper. With these added indicators, the "Out of Paper" LED would begin flashing as soon as this condition existed.

The other LED serves a similar function. It informs you when your printer is "Busy" - but not "Out of Paper." Therefore, if the "Busy" LED is on and the printer is not printing, you know the computer program is not the culprit; but instead the printer is locked up due to reasons other than being out of paper. A couple of possible reasons could be 1.) the paper is binding and unable to advance, or 2.) foreign material may be on the upper or lower guide bars, preventing the carriage assembly from returning or advancing, both of which I have experienced. In this case, the "Busy" LED would have displayed a steady light during the "Busy" condition; indicating there is a problem with the printer.

Parts List

Below is a list of parts necessary for construction of the Printer Interface Unit. Most can be purchased from your local Radio Shack dealer. The others can be obtained from electronic discount houses, whose ads are listed in the back section of many electronics magazines.

IDENTITY	TYPE	DISCRIPTION
IC1, IC2	74LS30	8 - Input NAND Gate
IC3	74LS368	Hex Inverter
IC4	74LS32	Quad 2 - Input OR Gate
IC5	74LS367	Hex Buffer
IC6	74LS175	Quad Latch
IC8	74LS123	Dual Multivibrator
R1,R2,R3,R5	4. 7K Ohms	Resistors
R4	20K ohms	Resistor
R6	470 Ohms	Resistor
C1	200 PF	Capacitor
LED 1	20 mA	Steady Light
LED 2	FRL -4403	Flashing Red LED

S-80 Connector

Line Printer connector
40 Lead Ribbon
34 Lead Ribbon

40-Pin Bus Connector

36-Pin Amphenol Connector
Cable
Cable

Coming Up

Next issue I will talk about the section of the Expansion Interface which enables you to increase your memory up to 48K RAM. Its design and operation will be discussed in detail. This will be followed up with construction plans for an inexpensive out-board memory expansion unit ideal for those of you who desire more memory, but do not wish to invest in a complete Expansion Interface unit at this time.

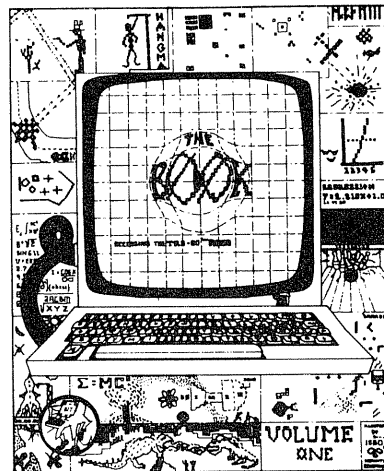
BACK TO BASIC

THE BOOK

ACCESSING THE TRS-80* ROM

Dealer Inquiries Invited

If you ever do
Assembly
language
programming,
or you just want
to know more
about your
TRS-80 ROM,
"THE BOOK"
is for you.



Volume I will give you access to over fifty machine language subroutines in the Radio Shack Level II BASIC. It includes information on the numeric data formats and a commented listing of the ROM routines.

"THE BOOK, Volume I", encompasses all arithmetic functions and mathematical operations. There are separate routines for integers, single precision, and double precision numbers and the data format for each of these number types is explained. The routines that perform ASCII to binary and binary to ASCII conversion are identified and explained to provide you a means of data I/O.

A fully commented listing provides the details on the step-by-step execution of these ROM routines. Although a complete disassembly is not provided in order to avoid copyright infringement, you can obtain a complete disassembly using the disassembler program listed in "THE BOOK." Volume I also includes a complete, detailed memory map of the entire machine and a symbol table noting over 500 addresses.

"THE BOOK" will save you hour upon hour of assembler program development time. Don't start programming without it.

Order your copy of "THE BOOK", today!

Insiders Software Consultants, Inc.

P.O. Box 2441, Dept. SUM 1
Springfield, VA 22152

*TRS-80 is a trademark of
Tandy Corp.

☐ Please send me Volume I of THE BOOK
at \$14.95 plus \$1.50 for postage.

NAME: _____

ADDRESS: _____

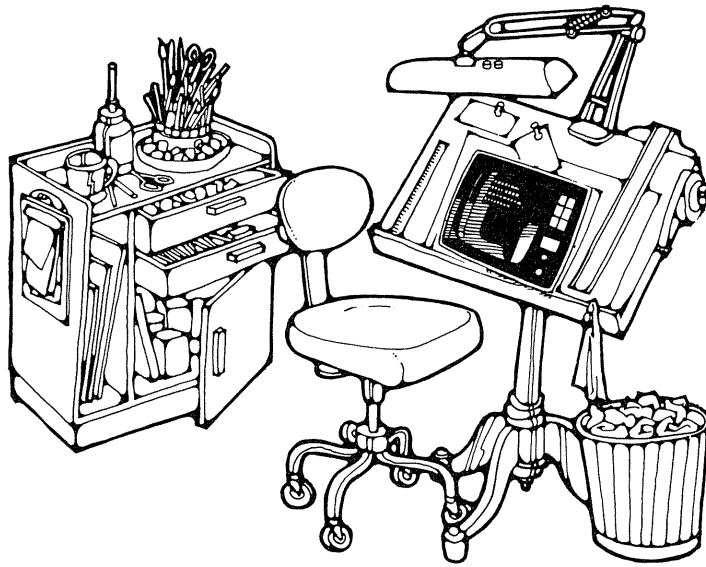
CITY, STATE _____ & ZIP CODE: _____

☐ Check payable to Insiders Software Consultants, Inc.

☐ MASTER CHARGE MC Bank Code:

☐ VISA Exp. Date: _____ Card Number: _____

Signature: _____



TRS-80 Graphics Editor

William K. Mason
Hornitos, California

"That's a good effort, for a seven year-old."

The lady thought she was praising the graphics work of my son Alan. I didn't say anything. The Klingon spacecraft on the TRS-80 screen was meant to appear menacing. . . . Unfortunately, one wing was too thick, the port holes were crooked, and the laser had a crimp in its barrel. The object on the screen resembled not so much a deadly Klingon as it did a cross-eyed squid. The author of this crude effort was, alas, me.

I decided then and there that I needed a graphics editor, a program that would allow me to correct screen drawings as the Level II Editor allows me to correct program lines.

The first step was to make a list of my typical graphics errors: figures too close together or too far apart, lopsided figures, figures too small, supposedly identical figures not identical, figures without enough detail, figures with unfathomable graphics codes. . . . It was a long list.

Attacking it one error at a time, I arrived, finally, at the Graphics Editor listed in this article. Using the Editor you may make a drawing on the screen, and then move parts of it around while leaving other parts fixed; the Editor combines, inserts, deletes, duplicates, rotates, tilts, magnifies; it prints out graphics codes. It helps me turn a rough sketch into something I can admit composing.

Instructions are included in the Graphics Editor program. Here are a few additional notes:

1. You always have a flashing indicator to show your current position on the screen. You may switch back and forth between a flashing dot and a flashing cursor. The flashing dot erases as it moves; the flashing cursor does not.

Hitting keys at random will show you that there are different responses to the same key depending on whether the dot or cursor is flashing. The only key that always gives the same response is the '9' key: it brings

up the instructions.

2. You can mark off a part of your screen drawing and make it the "designated figure." The designated figure can be moved and stored independently of the rest of the drawing.

This is handy, for example, if you've got two figures that aren't the right distance apart. You don't have to redraw them. Make one of them the designated figure and move it.

3. I haven't the patience to draw large figures. I usually get about halfway through and say the heck with it, it wasn't going to come out right anyway. To cope with this, the Editor has a magnification feature. The idea is: you first draw a small picture and let the computer enlarge it. Then you fill in the details.

4. To get a symmetrical figure, and avoid a lopsided one, sketch half of it and let the computer sketch the other half. In a bit more detail: draw half the picture and make it the designated figure. Let the Editor make rotated or tilted copies of it. Move the designated figure next to the copy that represents the other half of the figure you want. Make a new designated figure out of the two halves. Then move it, magnify it, etc.

5. Whenever you designate a figure, the Editor pauses to print out a string of graphics and control codes that defines the figure. You may want to use the string in some other program, an animation program for example. Animating a figure in BASIC is best done by defining the figure with a string, and then using the PRINT@ command to alternately print and erase the figure.

6. You can move your drawing back and forth between the screen and protected memory. This is useful, among other things, for putting copies of the designated figure at different points in your drawing. Move the figure to the first point and save the screen to memory. Move the figure to the next point. Load memory back to the screen. This erases the figure you've just

moved, but you can load it back: the screen then includes two copies of the figure. Save the screen and move the figure to the third point, etc..

The screen is saved to protected memory locations 31667-32690. You can save a picture permanently by dumping these locations to tape. Similarly, you can load an initial drawing from tape into these locations before you CLOAD and RUN the Graphics Editor. The dumping and loading can be done with T-Bug or some other monitor, or with a program like the "Fast Array Save" in the *80-U.S. Journal* (July/Aug. 1980).

7. How do you designate a figure? You first bring up the flashing cursor. Then you use the 'Shift' and arrow keys to place '+' signs at strategic points on the screen. Then you position the flashing cursor and hit 'Enter.'

There are three ways to place the '+' signs and the cursor. The first way is to cover the entire figure with '+' signs, leave the flashing cursor on a '+' sign, and hit 'Enter.' This way requires the most keystrokes, but results in a string of minimum length. The second way is to place '+' signs in diagonally opposite corners of an imaginary box, leave the flashing cursor inside the box, and hit 'Enter.' This way uses the fewest keystrokes, but includes everything inside the box as part of the designated figure. The third way, a compromise between the other two, is to cover the boundary of the figure with '+' signs, making square corners at turning points, leave the cursor inside, and hit 'Enter.'

The third way, incidentally, comes from a routine for counting territory in the game of GO. It's the only part of my GO playing program that works.

8. The editing commands use only a fraction of the keys available. If you want to dream up a few subroutines of your own, all the letters of the alphabet are there, waiting. ●

For Model I, Level II 16K

```

10 REM * TRS-80 GRAPHICS EDITOR *
20 REM * W. MASON, BOX 316, HORNITOS, CA, 95325 *
30 REM * INITIALIZE *
40 POKE16561,177:POKE16562,123:REM* SETS MEM SIZE=31667*
50 CLEAR2000:CLS
60 DEFSTR A-H:DEFINT I-Z:XS=63:YS=15
70 DIM S(63,15),F(15,1)
80 PRINT"GRAPHICS EDITOR--BY W. MASON"
90 PRINT
100 INPUT"DO YOU WANT INSTRUCTIONS":A
110 IF LEFT$(A,1)="Y" GOSUB1790
120 F=CHR$(191):REM SOME INITIAL FIGURE MAY BE INSERTED HERE FOR
    R'F'.
130 PF=0:REM PF IS PRINT@ LOCATION OF F
140 E=" ":REM E ERASES F
150 F(0)=F
160 REM * MACHINE LANGUAGE TRANSFER BETWEEN MEMORY & SCREEN *
170 POKE16526,180:POKE16527,127
180 FORX=32692 TO 32711:READ Y:POKE X,Y:NEXT
190 DATA 205,127,10:REM* CALL 2687; GET ARGUMENT INTO HL *
200 DATA 203,69:REM* BIT 0,L; TEST L FOR 0 OR 1 *
210 DATA 33,179,123:REM* LD HL,31667; HL HAS MEM START *
220 DATA 17,0,60:REM* LD DE,15360; DE HAS SCREEN START *
230 DATA 1,0,4:REM* LD BC,1024; BC HAS NUMBER OF LOCS TO MOVE *
240 DATA 40,1:REM* JR 7,1; IF TEST WAS 0, SKIP NEXT LINE *
250 DATA 235:REM* EX DE,HL; INTERCHANGE START LOCATIONS *
260 DATA 237,176:REM* LDIR; MAKE TRANSFER *
270 DATA 201:REM* RET; RETURN *
280 ON ERROR GOTO1600
290 X=0:Y=0:CLS
300 REM * KEYBOARD SCAN--REGULAR MODE *
310 SET(X,Y):FORI=0TO20:NEXT:RESET(X,Y)
320 P1=PEEK(14368):P2=PEEK(14400):P3=PEEK(14464)
330 IF P1=32 GOSUB490 :GOTO310
340 IF P1=2 THEN K=USR(1):GOSUB1790 :K=USR(0):GOTO310
350 IF P1=128 AND P3=1 THEN750
360 IF P2=0 THEN310
370 IF P1=128 GOSUB610 :GOTO310
380 IF P2=1 GOSUB560 :GOTO310
390 IF P2=2 CLS:GOTO310

```

```

400 IF P2=128 GOSUB590 :GOTO310
410 GOSUB420 :GOTO310
420 REM * DRAW OR ERASE *
430 IF P3 THEN SET(X,Y)
440 IF P2 AND 8 THEN Y=Y-1:IFY(0)THENY=47
450 IF P2 AND 16 THEN Y=Y+1:IFY(47) THEN Y=0
460 IF P2 AND 32 THEN X=X-1:IFX(0) THEN X=127
470 IF P2 AND 64 THEN X=X+1:IFX(127) THEN X=0
480 RETURN
490 REM * PRINT SYMBOL *
500 P=INT(X/2)+64*INT(Y/3):A=INKEY$
510 PRINTAP," ";FORI=1TO10:NEXT:PRINTAP,"="";FORI=1TO10:NEXT
520 A=INKEY$:IFA=" "THEN510
530 PRINTAP,A;
540 X=X+2:IFX(127)THENX=0
550 RETURN
560 REM * SCREEN TO MEMORY *
570 K=USR(1):CLS:PRINT"RECORDING":FORI=1TO1000:NEXT
580 K=USR(0):RETURN
590 REM * MEMORY TO SCREEN *
600 K=USR(0):RETURN
610 REM * MOVE FIGURE *
620 IF P2=0THEN680
630 PRINTAP,E;
640 Y1=INT(PF/64):X1=PF-64*Y1
650 GOSUB690
660 PF=X1+64*Y1
670 IF P2(2) PRINTAP,F;
680 RETURN
690 REM * UPDATE POSITION *
700 IF P2 AND 32THENX1=X1-1:IFX1(0)THENX1=63
710 IF P2 AND 64 THEN X1=X1+1:IFX1(63) THEN X1=0
720 IF P2 AND 8 THEN Y1=Y1-1:IFY1(0)THENY1=15
730 IF P2 AND 16 THEN Y1=Y1+1:IFY1(15)THENY1=0
740 RETURN
750 REM * KEYBOARD SCAN--DESIGNATE MODE *
760 X1=INT(X/2):Y1=INT(Y/3)
770 P=X1+64*Y1:S=PEEK(15360+P)
780 PRINTAP,"-";FORI=0TO20:NEXT:POKE15360+P,S
790 P1=PEEK(14368):P2=PEEK(14400):P3=PEEK(14464)
800 IF P1=4 GOSUB1480 :GOTO770
810 IF P1=2 GOSUB900:GOTO770
820 IF P1=64 GOSUB1680 :GOTO770
830 IF P1=32 GOSUB1730 :GOTO770
840 IF P2=0 THEN780
850 IF P2=128 THEN900
860 IF P2=1 GOSUB990 :GOSUB2010 :K=USR(0):GOTO900
870 IF P2=2 GOSUB1450 :K=USR(0):GOTO770
880 IF P3 GOSUB910 :GOTO770
890 GOSUB690 :GOTO770
900 FORI=1TO200:NEXT:X=2*X1:Y=3*Y1:GOTO300
901 REM * SAVE WORK, CALL INSTRUCTIONS *
902 IF LF=0 THEN K=USR(1):GOSUB1790:K=USR(0):FORI=1TO200:NEXT:R
    ETURN

```

```

903 GOSUB1790:K=USR(0):PRINT0,"PAUSE";
904 FORY=YS TO YB:FORX=XS TO XB:IF S(X,Y)=1 POKE15360+X+64*Y,43
905 NEXTX,Y:FORX=0TO4:I=15360+X
906 IF S(X,Y)=1 THEN POKEI,43 ELSE POKEI,PEEK(31667+X)
907 NEXTX:RETURN
910 REM * MAKE LOCATION P PART OF DESIGNATED FIGURE *
920 IF LF=0 THEN K=USR(1)
930 LF=1:S(X1,Y1)=1:POKE15360+P,43
940 IFX1(XSTHENXS=X1
950 IFX1(XBTHENXB=X1
960 IFY1(YSTHENYS=Y1
970 IFY1(YBTHENYB=Y1
980 GOSUB590:RETURN
990 REM * FILL IN TERRITORY *
1000 IF S(X1,Y1)<1 THEN S(X1,Y1)=2
1010 MF=0:K=X1:L=X1:M=Y1:N=Y1
1020 IFK=X THEN K=K-1
1030 IFL(XB THEN L=L+1
1040 IFM=Y THEN M=M-1
1050 IFN(YB THEN N=N+1
1060 FORX=K TO L:FORY=M TO N
1070 IF S(X,Y)=2 THENI100
1080 NEXTX,Y:IFMF=1 THEN MF=0:GOTO1020
1090 GOTO1140
1100 POKE15360+X+64*Y,43
1110 S(X,Y)=1:FORI=-1TO1:FORJ=-1TO1
1120 IF S(X+I,Y+J)=0 THEN S(X+I,Y+J)=2:MF=1
1130 NEXTJ,I:GOTO1080
1140 REM * GET STRINGS E AND F *
1150 FORY=YS TO YB:F(Y)="":L=0:MF=0
1160 FORX=XS TO XB
1170 IF S(X,Y)<1 THEN IF MF=1 THEN L=L+1:GOTO1220 ELSEI220
1180 IF MF=0 THEN MF=1:T(Y,0)=X
1190 A=CHR$(PEEK(31667+X+64*Y))
1200 IF L=0 THEN F(Y)=F(Y)+A:GOTO1220
1210 F(Y)=F(Y)+STRING$(L,25)+A:L=0
1220 NEXTT(Y,1)=XB-L+1:NEXTY
1230 F=F(Y):PF=T(Y,0):I=T(Y,1):IFYS=YB THENI310
1240 FORY=YS+1 TO YB:IF F(Y)="" THEN F=F+CHR$(26):GOTO1300
1250 K=1-T(Y,0):IF K=0 THEN A=""
1260 IF K=0 THEN A=STRING$(K,24)
1270 IF K=0 THEN A=STRING$(-K,25)
1280 F=F+CHR$(26)+A+F(Y)
1290 I=T(Y,1)
1300 NEXTY
1310 E="" :FOR X=1 TO LEN(F):A=MID$(F,X,1)
1320 IF ASC(A)<32 THEN E=E+A ELSE E=E+" "
1330 NEXT X
1340 WB=YB:WS=YS:VB=XB:VS=XS
1350 REM * PRINT CODE FOR F *
1360 CLS:L=1:PRINT"F$=";M=ASC(LEFT$(F,1)):IF LEN(F)=1 THENI440
1370 FORX=2 TO LEN(F)
1380 Y=ASC(MID$(F,X,1))

```

```

1390 IF Y=M THEN L=L+1:GOTO1430
1400 IF L=1 PRINT"CHR$(";M;")+";:GOTO1420
1410 PRINT"STRING$(";L;";";M;")+";:L=1
1420 M=Y
1430 NEXTX
1440 IF L=1 THEN PRINT"CHR$(";M;")" ELSE PRINT"STRING$(";L;";";M;")"
1450 PRINT"15 SECOND PAUSE":LF=0:XS=63:XB=0:YS=15:YB=0
1460 FORI=0TO63:FORJ=0TO15:S(I,J)=0:NEXTJ,I
1470 RETURN
1480 REM * MAGNIFY FIGURE *
1490 CLS
1500 GOSUB1560
1510 FORY=3*(WB-WS+1) TO 0 STEP-1:FORX=2*(VB-VS+1) TO 0 STEP-1
1520 I=POINT(X,Y):SET(X,Y):RESET(X,Y)
1530 IF I=0 THEN 1550
1540 FOR L=0 TO 1:FOR M=0 TO 1:SET(2*X+L,2*Y+M):NEXTM,L
1550 NEXTX,Y:RETURN
1560 REM * PUT FIGURE IN UPPER LEFT CORNER *
1570 FORY=0 TO WB-WS: IF F(Y+WS)=" THENI590
1580 PRINT064*Y,TAB(T(Y+WS,0)-VS) F(Y+WS);
1590 NEXTY:RETURN
1600 REM * ERROR TRAP *
1610 IF ERR/2+1=5 THEN RESUME NEXT
1620 IF ERR/2+1=9 THEN RESUME NEXT
1630 IF ERR/2+1<15 THEN ON ERROR GOTO0
1640 FORI=1TO3:CLS:PRINTCHR$(23)
1650 PRINT"TAKING ONLY 255 BYTES OF FIGURE"
1660 FORJ=1TO1000:NEXT:CLS:FORJ=1TO50:NEXT
1670 NEXTI:YB=Y:RESUME 1310
1680 REM * DIFFERENT VIEWS OF FIGURE *
1690 CLS:GOSUB1560
1700 FORY=3*(WB-WS+1) TO 0 STEP-1:FORX=2*(VB-VS+1)TO 0 STEP-1
1710 IF POINT(X,Y) THEN SET(127-X,Y):SET(127-X,47-Y):SET(X,47-Y)
1720 NEXTX,Y:RETURN
1730 REM * TILT FIGURE *
1740 CLS:IF VB-VS>15 PRINT"FIGURE TOO LONG":FORI=1TO1000:NEXT:RETURN
1750 GOSUB1560
1760 FORY=0 TO 3*(WB-WS+1):FORX=0 TO 2*(VB-VS+1)
1770 IF POINT(X,Y) THEN SET(60+2*Y,X):SET(61+2*Y,X)
1780 NEXTX,Y:RETURN
1790 REM * INSTRUCTIONS *
1800 CLS:PRINT"INSTRUCTIONS":PRINT
1810 PRINT"THE GRAPHICS EDITOR OPERATES IN TWO MODES: REGULAR M
CODE
AND DESIGNATE MODE. REGULAR MODE HAS A FLASHING DOT (";CHR$(130
);";).
DESIGNATE MODE HAS A FLASHING CURSOR (-).".
1820 PRINT:PRINT"THE COMMANDS IN EACH MODE ARE:";GOSUB2010
1830 PRINT"COMMANDS (REGULAR MODE)":PRINT"ARROW(S)";TAB(18)"MOV
E FLASHING DOT IN INDICATED DIRECTION.";PRINTTAB(19)"ERASES
AS IT MOVES."

```


9 TRACK TAPE DRIVE FOR A MICRO-COMPUTER ?

INTERFACES FOR MICROCOMPUTERS . . .

- **POS MEMORY/PORT MODULE** for TRS-80 — Here is a programmable device controller which plugs to the TRS-80 40-pin bus and provides 1.7K of PROM and/or RAM plus 18 input and 18 output lines addressed as 3 parallel ports. Designed as a controller for daisy-wheel printers and 9-track tape drives, its uses are limited only by one's imagination. Includes sockets for 1.7K RAM, 1.7K EPROM, or 1K PROM and .7K RAM (memory ICs not included). Requires +5VDC, +12VDC power source. Memory is addressed at 3000H to avoid conflict with other system and user-available memory. Ship wt.: 3 lbs. Price \$150.00
- **DAISY-WHEEL PRINTER INTERFACE** for TRS-80: This interface will drive Diablo HyType I, HyType II, and Qume Q Series and Sprint 3 printers (specify cable required). Includes 1K user-available memory for custom print routines (such as graphics, bidirectional printing, etc.). Programmed to respond to print commands from BASIC, ELECTRIC PENCIL™, and SCRIPSIT™ software. Draws its power from printer. Ship wt.: 5 lbs. Price \$250.00
Cables, each \$ 25.00
(Specify HyType I, HyType II, or Qume)
- **ASCII INTERFACE** for IBM I/O SELECTRIC — This Centronics-style parallel printer interface will drive an IBM Model 731 or 735 I/O typewriter (EBCD and Correspondence codes). No software needed. Features on-board EPROM which holds up to 8 ASCII-to-IBM code translation tables for different type spheres. Closed-loop operation runs at maximum printer speed; stops and starts on a single character without loss of data. Requires +12VDC and +5VDC power source. Ship wt.: 3 lbs. Price \$249.95
Power Supply \$ 49.95
(+5VDC, +12VDC, +24VDC for Solenoids on Printer)
- **CONVERT 15" IBM OFFICE SELECTRIC to I/O TYPEWRITER** — Kit includes assembled solenoids, switches, wire harness, magnet driver PCB plus instructions for installation and mCPU interface. Price \$200.00

Open up the world of IBM to the microcomputer. Now your micro can read and write IBM/ANSI compatible NRZ1 format 9-track magnetic tapes, as used on the largest IBM computer installations. Read government statistics. Write tapes that can be loaded into an IBM 370 computer. Here is an IBM-compatible interface that won't void the IBM warranty!

A medium-speed (37.5 ips) 800 BPI tape drive can transfer data at a rate of 30,000 characters per second, making it an ideal back-up storage media for the new hard-disk systems. The Phase-Encoded 1600 BPI drives have twice the data capacity of the NRZ1 drive, and can hold up to 40 megabytes of data on a single reel of tape.

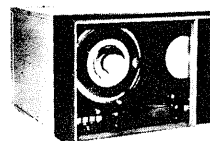
- **POS-100 NRZ1 TAPE DRIVE CONTROLLER/FORMATTER** — For over 2 years Pacific Office Systems has been quietly selling its POS-100 NRZ1 Tape Drive Controller/Formatter for "Perfec" industry standard NRZ1 tape transports and the S-100 computer. The POS-100 operates with 2K of 8080 or Z-80 subroutines to Write a Record, Tape Mark or Check Sum Characters, Read a Record Forward or Backward, Go Forward or Backward a Record or File, Erase a Gap, Rewind, etc. Software also checks status of tape drive during operation and provides error messages. Software can be supplied on CPM or NorthStar diskette, or on EPROMs.

The POS-100 consists of S-100 bus card, 6' ribbon cable, tape drive controller card, cable to Perfec-Standard NRZ1 Tape drive, plus documentation and Z-80 or 8080 software (specify). Power is derived from tape drive and S-100 bus. Ship Wt.: 10 lbs.

Suggested Retail Price \$995.00

Optional CPM utility programs to copy data from diskette to tape, from tape to diskette, and from tape to CRT or Printer (record length variable up to size of system memory): Price \$100.00

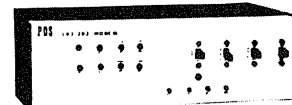
- **REFURBISHED MAG TAPE DRIVES** — Contact Pacific Office Systems for currently available refurbished 800 BPI and 1600 BPI mag tape drives by such manufacturers as PERTEC, CIPHER DATA PRODUCTS, KENNEDY, etc. Prices vary according to capabilities and condition of each unit.



COMING SOON . . .

- **A bus adapter for the TRS-80 Model One (40-pin bus).**
- **POS Stand-alone 800/1600 BPI Controller/Formatter:** A microprocessor-based controller and formatter for both 800 BPI (NRZ1) and 1600 BPI (PE) tape drives, with interfaces for serial RS-232 and 8-bit parallel ports. Controller is programmed to respond to simple ASCII commands to seek, read or write a file of data on tape via its 4K or 16K buffer memory. Interrupts can be used for status and command messages between controller and host CPU. Data transfer rates are expected to run as high as 19,200 baud (RS-232 serial port), and 100,000 baud (8-bit parallel port).

MODEMS



- **POS 103/202 "MIX or MATCH" MODEM** — BELL 103 and/or BELL 202 Frequencies: Unique POS control design permits use in one housing of both Bell-compatible 103 (0-300 baud) and 202 (0-1200 baud) modem modules originally made by VADIC Corp. for a telephone company subsidiary. FEATURES: RS-232 serial interface, auto-answer, auto-dial, LED display, telephone line interface via acoustic coupler, manual DAA, or auto-answer DAA (sold separately). FULLY ADJUSTED; no special tools required. 3,000 mile range over standard dial-up telephone lines.

— POS 103 MODEM \$199.95
— POS 202 MODEM \$299.95
— POS 202 MODEM (Auto-Answer) \$349.95
— POS 103/202 MODEM \$499.95
— FCC-approved Auto-Answer DAA \$125.00

PACIFIC OFFICE SYSTEMS

918 Industrial Avenue
Palo Alto, CA 94303
(415) 493-7455

```

1840 PRINT"SHIFT"+ARROW(S)";TAB(18)"DRAW LINE IN INDICATED DIR
    ECTION."
1850 PRINT"/"+ARROW(S)";TAB(18)"MOVE DESIGNATED FIGURE IN INDI
    CATED DIRECTION";PRINTTAB(19)"(DESIGNATED FIGURE DEFINED LAT
    ER)."
1860 PRINT"CLEAR";TAB(18)"CLEAR SCREEN.";PRINT"/"+CLEAR";TAB(18)"SAVE
    SCREEN TO MEMORY."
1870 PRINT"SPACE BAR";TAB(18)"LOAD MEMORY TO SCREEN.";PRINT"/"+
    "/"+SPACE BAR";TAB(18)"NEXT DESIGNATED FIGURE TO SCREEN.";P
    RINT"/";TAB(18)"PRINT NEXT SYMBOL ENTERED (= WILL":PRINTTAB
    B(19)"FLASH UNTIL SYMBOL IS HIT)."
1880 PRINT"/"+SHIFT";TAB(18)"GO TO DESIGNATE MODE.";PRINT"/"+S
    ";TAB(18)"GO TO INSTRUCTIONS.";GOSUB2010
1890 PRINT"COMMANDS (DESIGNATE MODE)":PRINT"ARROW(S)";TAB(18)"M
    OVE FLASHING CURSOR IN INDICATED";PRINTTAB(19)"DIRECTION. CUR
    SOR DOES NOT ERASE."
1900 PRINT"SHIFT"+ARROW(S)";TAB(18)"LEAVE "+ AND MOVE IN INDI
    CATED DIRECTION.";PRINTTAB(19)"SIGNS ARE USED TO DEFINE D
    ESIGNATED FIGURE."
1910 PRINT"ENTER";TAB(18)"RECORD FIGURE DESIGNATED BY '+ SIG
    NS. (ALSO:PRINTTAB(19)"PRINTS OUT STRING DEFINING DESIGNATE
    D FIGURE)"
1920 PRINT"CLEAR";TAB(18)"ERASE '+ SIGNS AND START AGAIN TO"
    :PRINTTAB(19)"DEFINE DESIGNATED FIGURE."
1930 PRINT"SPACE BAR";TAB(18)"GO TO REGULAR MODE.";PRINT:PRIN
    T"THESE ARE 3 WAYS OF USING '+ SIGNS TO DESIGNATE A FIGURE:
    ":GOSUB2010
1940 PRINT"METHODS OF DEFINING DESIGNATED FIGURE:"PRINT"1) COV
    ER ENTIRE FIGURE BY '+ SIGNS, LEAVE FLASHING CURSOR ON A '+
    , SIGN, HIT 'ENTER'."
1950 PRINT:PRINT"2) PLACE '+ SIGNS ON DIAGONALLY OPPOSITE CORN
    ERS OF AN
    IMAGINARY RECTANGLE, LEAVE FLASHING CURSOR INSIDE RECTANGLE,
    HIT 'ENTER'. THE PART OF THE SCREEN DEFINED BY THE RECTANGLE
    WILL BECOME THE DESIGNATED FIGURE."
1960 PRINT:PRINT"3) COVER BOUNDARY OF FIGURE BY '+ SIGNS, MAKI
    NG
    SQUARE CORNERS AT TURNING POINTS, LEAVE FLASHING CURSOR INSIDE,
    HIT 'ENTER'. THE '+ SIGNS WILL SPREAD OUT FROM CURSOR UNTIL
    THEY REACH BOUNDARY. THE DESIGNATED FIGURE";
1970 PRINT" WILL BE THE PART OF
    THE SCREEN COVERED OVER WITH '+ SIGNS.";GOSUB2010
1980 PRINT"DESIGNATE MODE (ADDITIONAL COMMANDS)":PRINT"*
    INT DESIGNATED FIGURE MAGNIFIED FOUR TIMES.";PRINT".
    NT FOUR ROTATED VIEWS OF DESIGNATED FIGURE."
1990 PRINT"-
    PRINT TILTED VIEW OF DESIGNATED FIGURE.":PRINT
    "9 GO TO INSTRUCTIONS."
2000 PRINT:PRINT"SUGGESTION: READ INSTRUCTIONS, PRACTICE FIRST
    COMMAND, GO
    TO INSTRUCTIONS, PRACTICE SECOND COMMAND, ETC.. NOTE: IN
    EITHER MODE, HITTING '9' WILL GO TO INSTRUCTIONS.";GOSUB2010 :R
    ETURN
2010 PRINT:PRINT"90, HIT ENTER TO CONTINUE";:INPUTA:CLS:RETURN
    
```

The Making of a Computer Program

Mike Schmidt
Editor

The following is a story about the conception, birth and growing pains of a real life computing project. It has reached maturity now, and we can sit back and reminisce about the rough spots. Although some of you may not know it, you were unwittingly involved in its development. For that, we almost owe you an apology; you will see why as we go along.

Rather than go through this in fine detail, we will step through chronologically. Keep in mind that not all programs develop this way. There are thousands of programs out there, this is just one of them...

The Need

It is obvious that to run a mail order business, you need your customer's names and addresses. You need them not just to fill orders, but to keep track of your repeat customers and to send promotional information or catalogs. Assume that your customers are in every Zip code. A Model I TRS-80 with two or three disk drives and 48K of memory sounds like the real thing, so you use it to keep your mail file. Let us say you started small, and the system handled all your customers without problems.

The Problem

Although it may be a nice one to have, the problem is growth. Your widgets, or whatever you sell, are catching on. People want them, and your orders begin to increase. One fine day you find that your mail/label program has reached its capacity. Now what?

You can look for another program that handles more names than yours does. Or, you can buy a bigger computer. But you are busy making bigger and better widgets, and don't have time to write a program for yourself.

The First Solution

Being an expedient person, you say "Well, all we have to do is segment the mail file into Zip codes. Put the 0xxxx on one disk, the 1xxxx on the next, and so on." So you do it, and it gets you off the hook for a while. This solution has a little problem connected with it though. Now, you need to presort your mail into Zip code stacks before you can enter it into your system. Also, when one of your customers changes address, especially from one zip to another, well, it gets sort of messy.

Time passes, and you suddenly find that the people in the 0xxxx zip must *really* like your widgets, since the 0 zip has reached its maximum size. Now you need to break it into two disks, one from 00xxx to 059xx and the other from 06xxx to 09999.

The same thing happens to a few other zip codes, and you have about sixteen disks to hold your files. Now the mail must be presorted even finer. Since your employee gets so good at sorting mail, she leaves you for a better paying job with the Post Office. (They immediately put her to work as a second level supervisor, but that's not part of this story.)

The Requirement

This problem finally gets your attention (now that your help is gone and you have to do it all yourself). So you decide to get a Model II, with those big 8 inch floppy disks. That ought to do it, right?

Next, you start looking for a program that will work with it, and you find one that handles 2500 names in one file! Now you can consolidate more than a dozen disks into just three, and have some space left for expansion. It works fine, but you have just pushed the problem a little further into the future, and you still have the presort problem and the address change problem.

What you really need is a program that will span four drives on your Model II and look like one virtual file. That's a pretty big file, since you now have the capacity of about 2 million bytes for disk storage. How long is it going to take to sort, or to simply find someone in a file that big?

The Confession

In case you haven't guessed by now, widgets are the 80-U.S. Journal, the Model II program is the Mail/File from Bill Schroeder of Galactic Software Ltd, and the girl who left is a figment of my imagination.

In about February 1980, I contacted Bill about this problem, and he did some homework. He found that about 90% of publications in this country have a circulation of between 2 and 10 thousand. Amazing, isn't it? You would think with *Better Homes & Gardens*, *Playboy*, *Time* and other biggies, that this couldn't be, but they are only the top ten percent. The rest are smaller and many are regional. He also found the Post Office was thinking of a 9 or 10 digit Zip code. He then set to work, making a Mass/Mail system. Since 80-U.S. had the requirement, we became his test site.

I wish I would have recorded some of our telephone conversations -

"Send me a backup of your three Mod II Mail/File disks", he said, "and I'll transfer them over to the new system and send it to you to try."

He said the new Mass/Mail system

would hold 10,500 names, spanning three drives, and the drive 0 disk would be program and control files. It also had instant recall, using a control number. The key fields (name and Zip code) were under constant sort, and it even had a "batch mode processor". With the batch mode you simply do all the updates and additions and deletions and it accepts them immediately. It puts them into a "holding file" on drive 0 and when you are done, you tell it to batch them in while you go on about other business. You don't even *have* to batch them after the session. You can simply turn the system off, and the next time you load the program, it will tell you that you have still "n" items to batch from last time!

Working out the Bugs

"Format a new disk and put it into drive 2", he said. "Then add names until you get to 3499 and stop and call me back."

Well, we did. And while on the 'phone with Bill we added enough new names to cross over to the second drive. It worked, and we were elated. But not for long. We ran a directory listing in Zip code order and found some recent additions to the file were there twice. Also in a few cases, Oregon popped up in New York and West Virginia showed up in Washington.

We spent hours on the telephone, and for the second of many times, the disks were backed up and sent to Bill via Federal Express. A few days later they came back, followed closely by a call from Bill.

"They're going to work now, just keep using them the way you normally would", he said.

Once a week I updated the file, and once a week I updated the other Model II Mail/File for backup. The difference between the two was astounding. The Mass/Mail system took about two hours, where the Mail/File program took almost all day, mostly due to the fact that there were many changes of address from one disk to another. I even got confident enough to announce in the pages of the Journal that we were going to start using real expiration dates on the labels, - but it wasn't to be.

Hardware Bugs

It turned out that TRSDOS 1.2 for the Model II had some problems which we were not aware of. According to Bill, it "went out to lunch" at unexpected times. More specifically, it lost its device control, especially when there were more than three files open at once. The Mass/Mail system had as many as five files going at one time, and every now and then it simply

I'M A BELIEVER !!

"I Love it !! . . . It's really a incredible O/S. It' just great!
Now I see why people who have seen it say they are now
believers. I know I am."

LANCE MICKLUS

- 1) Large (8") drive support.
- 2) Double Sided drive support.
- 3) Double Density drive support.
- 4) 80 Track drive support.

*NOTE all above drives may be mixed on any one system and can be configured at Sysgen time or during any Backup!

5) Winchester technology fixed drive support.

6) Supports any combination of the above drives up to a max. of 8 drives.

7) Supports double-speed processor clock modifications. (Archbold for example)

8) FASTER! ——— Improved overlay structure using ISAM accessing techniques improves loading times by up to 1400%.

9) General purpose output spoolers of a true, symbiont design provide simultaneous output and program execution without any user intervention.

10) Keyboard Type-Ahead feature permits you to enter keystrokes before your programs need them.

11) User definable keys, all 26 letters.

12) Built in Graphic string packer lets you enter graphic symbols into a BASIC program from the keyboard through the use of the (Clear) key. The (Clear) key is simply held down (just like the (Shift) keys) during other keystrokes and viola...graphics!

13) Dated files. — All files are accompanied by the date of their last modification (creation or write).

14) Marked files. — All files are accompanied by a 'mark' if they have been modified since they were last backed up. This permits the BACKUP utility to copy only those files which have actually been updated since a previous backup.

15) File transfer by class. Allows transferring of all files of a similar directory classification such as /CMD, /BAS, /PCL, etc.

VTOS 4.0

VTOS 4.0

Operating System
Diskette with
Operator's Guide

\$99.95

VTOS 4.0

Master
Reference Manual

\$29.95

VTOS 4.0

Combination -
4.0 disk,
Operator's Guide,
and Master
Reference Manual

\$125.00

16) Built-in SYSTEM command contains lower case display driver, screen print, break key disable, blinking cursor, disk drive stepping rate and motor-on delay modifications, and more.

17) User may SYSGEN a custom VTOS system configuration containing special I/O drivers, device LINKing and ROUTing, SPOOLing and DEBUG tasks, etc. which will be automatically loaded during the BOOT process without requiring a more lengthy AUTO and CHAIN procedure.

18) Non-BREAKable AUTO and CHAIN commands.

19) Wild-card DIRectory. Permits you to locate all files of a certain classification such as '/BAS'. Uniformly indicates file size in K (1024 bytes) regardless of drive type. "DIR D" would give you all your files that start with "D".

20) Dynamic file name defaults in APPEND, COPY, and RENAME commands allow you to specify only minimal information about file names.

21) COPY and APPEND commands execute up to 300% faster.

22) ALLOCate command for pre-allocation and non-releasability of file space. File space will never shrink if this option used.

23) MEMORY command for directly setting upper memory limit.

24) Variable Length file support is incorporated which automatically blocks short user data records both within a sector and across sector boundaries thereby taking maximum advantage of disk file space.

25) No security disk needed to make backups or to run the system!

26) Though many O/S bear his design and code VTOS 4.0 is the only Fully Approved Operating System by Randy Cook! And it is FANTASTIC!

27) Endorsed by Scott Adams and Lance Micklus!

VTOS and VTOS 4.0 are registered trademarks of VIRTUAL TECHNOLOGY, INC. - Dallas, Texas 75234

**Available from the following distributors or
your local computer store. DEALER INQUIRIES INVITED.**

5% Discount Just For Mentioning This Ad. (Valid month of this publication ONLY)



**QUALITY
SOFTWARE
DISTRIBUTORS**

11234 Park Central Pl Suite C
Dallas Texas 75230
(214) 692-1055
Micronet - 70130,203
SOURCE - TCC293



ADVENTURE INTERNATIONAL
Box 3435, Longwood, Fla. 32750
(305) 862-6917 - Voice
after 8:00 - same number
as FORUM 80. (SOURCE - TCC957)



**SMALL BUSINESS
SYSTEMS
GROUP**

Main St. and Lowell Road
Dunstable, Mass. 01827
(617) 692-3800 - Voice
(617) 692-3973 - FORUM 80
Micronet - 70310,236

Prices subject to change without notice

Background

quit.

By now most of us know what happens when you lose I/O with a random file open. You simply lose the file. Although the problem is rather well defined now, at that time we were unaware of the "holes" in TRSDOS 1.2, and couldn't identify the problem as hardware or software.

Federal Express got some more of our business because of this. The files were sent back to Bill and he had to reconstruct them. Then, when the system problem was identified, he had to program around it and provide error trapping to prevent loss of the files again. I don't think the problem with 1.2 was ever resolved, but that it doesn't exist with the TRSDOS 2.0 version.

During this process we lost 32 names (I told you that you were maybe involved!), without even knowing it. It was Bill who realized this loss because the numbers in the various control files didn't add up right.

By running directory listings on both the Model II Mail/File and the Mass/Mail system, we were able to find the lost souls and return them to the flock. The problem of why they were lost was serious enough to justify a trip, and Bill flew out from Milwaukee to check his programs on our system. Again it looked like we were home free, and again, it didn't happen..

Operator Problems

With any large, sophisticated system there are numerous details which may escape the attention of the programmer. One of the last problems we ran into happened as follows: The file is being updated and a subscriber has just sent in a renewal. The operator called the subscriber from the file by control code and changed the expiration date and then filed the record for update. This update goes into a holding file on drive 0, waiting to be batched in later. Now the operator sees that the subscriber also had a change of address when he renewed, and calls him back up by control code and changes the address, and again files him for update.

The net result was chaos! The record now gets into the main file twice, and also displaces another record already there!

Long before this, Bill created a "recovery" program which reads the data files and reconstructs the control files. This is something we could run, but he wants to see the actual files to determine the exact nature of the failure and correct it. The files go back to him again. This time they come back all straightened out, plus the ability to do more than one update before batching. We finally have a working Mass/Mail system.

User Additions

"So great", I said to Bill. "Now that we have it working, how can we get a Postal Zone count for 2nd Class circulation?"

"No problem", he says, and whipped up a neat little utility which reads the data files and totals the number in each zone.

This is a custom program, since postal zones vary depending on where you are located. Bill has included this option with the Mass/Mail system. You simply send

him your particular zone chart (available from your post office) and he customizes your program to give zone breakdown from your location.

Bells and Whistles

The program has complete error trapping. You can now hardly do anything wrong. The ESCAPE key can be used anytime to breakout of the program and return you to the main menu. It will accept only legal entries, other keys are locked out. Since the original versions, the speed has been increased significantly. The batching speed is 60% faster than before. It is now possible to update records by control code, switch to find someone by name or zip and then continue by control code.

Searching by control is so fast that by the time your finger leaves the ENTER key, the record is on the screen - even with a 10000 name file! Searching by zip code takes a few seconds as does searching by name. Searching by any other item in the record takes time, depending on where they are in the file. Any item in the record may be used as a search code.

There is a Mass/Purge capability, wherein you may kill all records with a certain code. This feature is password protected (your choice of password) to prevent inadvertent mass murder in the file. There is also a Mass/Update feature, which allows you to change an item in every record. This is also password protected.

"I think the system will stand the test of time...it is probably the most sophisticated system around for micros today"

When printing or searching, you have a choice of up to 19 different selection codes. For example, only the people in Michigan who expire in May 1980 and whose names start with "B". There is also an exclusion code which is unique. You may print all records *except* those with a certain code.

Updating the file is easy and fast. Searching is chained, which means you do not have to return to the main menu after each search. Once the record is on the screen, selecting any field in it automatically puts the cursor at the end of that field. You may then add to the end of the field, use the left arrow to transparently move through the field and change a letter or more, or you may use the F1 key, which wipes out that field and puts the cursor at the beginning of the field. This feature will be even more appreciated when we have to start adding four more digits to the zip codes starting in February 1981.

Speaking of zip codes, there is space allowed for an eleven digit, alphanumeric zip, so no matter what they finally decide to do, this system should handle it.

Another interesting and very useful feature is called "pre-defined input keys".

Say you are updating the file, and each record you add today will get the same code in some field. You may use up to four pre-defined inputs, and when you get to the field you want you simply use the control key and D, E, F or G and your pre-defined input will automatically appear in that field. Very handy..

Printing

The system has a directory listing, which may be used for an 80 or 132 column printer. It can be ordered by zip code, control code or alphabetically. All the selection codes may be employed here as well.

Labels may be printed one-up or up to six across. There are two ways to print labels. One is in standard label format. The other is a *User Defined* label, in which you decide what gets printed and where it gets printed. The user defined mode is slightly slower than the standard label format. Again, all selection codes are applicable to printing labels, including the exception capability. There is also the option to print from 1 to 99 of each label.

Labels may be printed in a 3 or 4 line format. A message line is also available, which can be used for "ATTENTION ADVERTISING DEPT" or "SERVICE MGR" or "MERRY CHRISTMAS". What you say here is up to you.

Since up to four disks can be used in a set, it may be possible to get the wrong disk into a drive. Mass/Mail checks for this possibility, and lets you know about it quick. It also keeps track of each session, and writes this information to the control disk. This also checked, just in case you may have your work disks mixed up with your backup disks.

Conclusion

Field testing Mass/Mail was a long, sometimes frustrating experience. It was worth the effort though, since we now have a working system. Our end of it was nothing compared to what Bill and his people went through. It took them 6 man-months to write and debug. This represents a considerable investment, and indicates a break-even point quite a way down the road. I think the system will stand the test of time. In its capability range, it is probably the most sophisticated system around for micros today.

Anyone currently using the Model II Mail/File who moves up to the Mass/Mail system may have his files converted by Galactic. Also, if a four drive Model II user of Mass/Mail decides to go to hard disk, the conversion capability is there. Incidentally, Mass/Mail can handle up to 32,000 records provided the storage medium is there.

Working with Bill Schroeder on this has been a pleasure. It is apparent from the support we got on the Model I and II Mail/File systems that he services what he sells. His documentation is clear and complete, and designed to reduce service calls in the middle of the night. Even then, I have called him, and he was there with an answer. ●

OWN A TRS 80*?..... SELECTRA-PRINT Puts You Just A Cable Length Away From A Word Processing System!

...just a simple hook-up with the supplied cables and your SELECTRA-PRINT is ready to run.

For Word Processing on a TRS 80, for example, just command "LPRINT" and SELECTRA-PRINT automatically outputs clear, clean, high-fidelity, hard copy....and of course you can use it to print-out any other information you might need.

SPECIAL NOTE:

SELECTRA-PRINT is a Selectric typewriter and

although it has been modified for computer print-out, it may still be used as a standard office typewriter.

SELECTRA-PRINT is VERSATILE!

SELECTRA-PRINT is compatible with most micro and mini-computers including ☐ Apple ☐ Heath H8 ☐ IMSAI ☐ Cromemco ☐ Alpha Micro-systems ☐ Space Byte ☐ North Star Horizon ☐ SWTP ☐ Vector Graphic ☐ Sol ☐ Polymorphic ☐ Digital Group ☐ Ohio Scientific ☐ Altair ☐ Sorcerer ☐ Xitan ☐ Rex ☐ KIM ☐ EXORcisor

Already own a Selectric I™, II™, or III™?

We can convert it to Selectra-Print

dealer inquiries invited

MICRO COMPUTER DEVICES, INC.

3156 East La Palma Avenue, Suite H, Anaheim, California 92806 - (714) 630-8206

*TRS 80 is a Trademark of Radio Shack, a division of the Tandy Corporation

TRS-80***MULTI-TASKING OPERATING SYSTEM***TRS-80^(R)

TRUE TIMESHARING WITHIN A TRS - 80

ADDS A NEW DIMENSION TO YOUR MODEL I SYSTEM

The first system utility to allow TWO USERS or programs to operate independently in a TRS-80.

.. .. .

- TSHARE V 1.2 is an interrupt driven executive which patches itself to NEWDOS or TRSDOS.
- Allows TRS-80 to be interfaced to a second terminal thus providing for an additional operating user in your EXPANDED SYSTEM. Additionally, a printer can be used to service both users.
- SIMPLEX mode for non serial-port users. Requires only a printer as the second "screen". Jobs share the keyboard under user control and detach to run separately but simultaneously. This mode allows non - interrupt driven timesharing.
- CONFIGURE allows segmenting of available memory above '7700 HEX in any proportion between the two users. Selects communication mode and port type for second terminal.
- Options for parallel port, RS232, TRS232, and HUH as the connection for your second terminal. All software drivers are included.
- Communicate between USERS or PROGRAMS using peek and poke. The experienced programmer can now create a new generation of multi - terminal operated games or business software.
- Execute BASIC or MACHINE LANGUAGE. Full use of disks. Requires 32K plus one disk drive.

INTRODUCTORY OFFER on easy loading 5¼ diskette
Full Documentation

\$89

*California residents add 6% tax.

COMSOFT

1124 N. Brand Blvd.
Suite 201
Glendale, California 91202
213/649-0369

TRS-80, TRSDOS tm Radio Shack/Tandy Corp.
NEWDOS tm Apparatus, Inc.
TRS232 tm Small System Software
HUH tm HUH Electronics



PROFILE II for the Model II

Bill Schroeder

About two months ago Radio Shack released a software package that will change the course of the micro-computer industry. PROFILE II has come to the market, and is the closest thing yet to true "DATA BASE" management for a micro.

The TRS-80 MODEL II is without a doubt the most versatile and powerful micro but it's acceptance has suffered from the lack of professional software available to run on MOD II TRSDOS. Although many special interest programs and many CP/M packages are available, few professional software houses have made high level business utilities available under TRSDOS. This fact was basically due to the new concepts put forth in MOD II TRSDOS and the state of flux the operating system seemed to be in. Now this seems to be solved with the just released and "set in concrete" TRSDOS 2.0 for the MOD II.

With the advent of PROFILE II, we not only have a very sophisticated operating system for the MOD II, we also have a dynamic DATA BASE management system.

I will try to describe some of the concepts and implementations of PROFILE II and hope to give details on actual applications in future issues. Profile is like having a whole new computer language, aimed straight at letting the businessman talk to his computer in an efficient manner!! Yes VIRGINIA, my computer CAN do something useful.

The program was written mainly by Howard Wolowitz with the aid of his partner, Bill Prady. Howard is an ex-IBM employee and Bill is an ex-Radio Shack computer center employee. Bill joined Howard in the spring of 1980. Based on the results coming from their company (The Small Computer Co.) they make quite a team.

Technical Confusion!?

DBM, FDC, SECTOR, BYTE, FILESPEC, RAM, MEMORY MAP, VIDEO DRIVER, SPOOLER, RS-232, VID RAM, PORT, ROM, COMPILER, HEX, ASSEMBLER, ASCII, BIT. Now that I have installed in this article all the technical terms that would be

conspicuous by their absence, the rest of this article will be in plain English.

What is a Micro Data Base Manager?

A Data Base Manager must provide certain minimum features to be of value to the applications of the end user. I feel that these are the minimum requirements.

1. The entire system must be user designed to the point of being mastered by the end user in no more than one day.

2. The system must make total use of the hardware that it is designed for and the upgrades to that hardware. This is essential so that when you buy that additional add-on drive, the software running your existing files can be told that more space is now available for it to use for added data records. This is the reason for adding those new drive(s) in the first place.

3. The set up of a new data base, including needed screen displays and print-out formats must be accomplished in less than 3 hours. This assumes an experienced operator with a good understanding of the system and of the data to be placed in the system for you.

4. The system must be void of errors that would cause the loss of your data. If errors are found, the vendor must be very responsive to the problem and its correction. (problems with computer software can never be solved fast enough, but the vendor must make every effort to do so).

5. The system must have different levels of operation built in. This is so that the "bright new guy" (person) in the office can develop data management systems to suit your exact needs, and then turn these systems over to regular office help to operate from a "clerk" level.

6. The system must allow for the setting up of new video displays and new print-outs for the existing data files with a minimum of effort.

7. Output to the printer should be available in report format, with proper headings, titles, pagination and dating easily set up. Label formatting must also be available.

8. All display, report or label formats that relate to a set of data files should be saved by the system for later use, and be able to be changed or updated easily by the user.

9. As the data base grows, the system must provide the user with methods of removing records by groups or class. The system should allow the user to move these records to an "archive" type file or, at the users option, destroy them to make room for new data records in the system. This function should also allow for the extraction from a large data base of specified subsets of that data base, to become a new file containing the specified set of records.

10. Fast, easy to use sorting of data must be provided. The entire data base should be able to be sorted on several different items in each record, as specified by the user.

11. Updating of the records in the system must be straightforward, easy for clerical personnel and fast enough so as not to be frustrating to the user. The longest it should take to search for any record in the system should be 15 seconds.

12. The system should be able to handle several of the jobs intended by the user for his computer. It should also perform these jobs nearly as well as a custom written program for the same job.

13. Basic mathematical functions should be provided so reports can reflect column totals and sub totals, as well as allow basic adding and subtracting of inputs from values in the records.

14. The data files that are created by the system should be in some easy to understand form to allow other languages such as COBOL or BASIC to access them. The data bases generated by the user will often need the attention of a professional programmer to provide certain utilities. The easier the files are to understand, the faster this programmer will be able to deal with your needs; therefore the bill should be lower.

Getting started with Profile II

Getting started with PROFILE II is fairly simple for the average person. The system is menu driven and well set-up for the non-programmer.

EDAS EDAS

A sophisticated Editor & Assembler setting the standard for the '80 Model 1 & Model 111. All EDAS commands and SOURCE text can be entered in either upper case or lower case. Direct assembly form memory or disk by means of *GET assembler directives. This gives text buffer capacity equal to your drive configuration! 30,000 bytes of symbol table.

Direct assembly to disk or memory for faster debugging operations! DOS "system" command functions KILL, DIR, FREE, and LIST are available from within the environment of EDAS.

The Editor, with renumber, maintains command syntax identical to the BASIC editor. Global change permits you to alter a string throughout a designated range of lines while block move relocates lines of text.

EDAS is priced at \$79 plus \$3 S&H. A 72-page manual included.

cmdfil

Now you can append two or more CMD files and/or SYSTEM tapes. Perform transfer to & from disk/tape of SYSTEM/CMD modules with offset capabilities. Read VTOS ISAM overlays. More! \$20

« MISOSYS » :serious software (tm)

VTOS 4.0, the system you have been waiting for is here. No ad could adequately describe the capabilities inherent in VTOS. MISOSYS provides full technical support for this system. You owe it to yourself to explore VTOS 4.0. Available for \$125 with the Reference Manual or \$99 without. Call or write for all the details.

VTOS 4.0

dsmbler

Complement your assembly language tools with this Z80 disassembler which produces screen, printer, cassette, or disk file output. A two-pass process provides SYMBOLS for 16-bit address and 8-bit relative references. EQUates & ORG are generated. Read SYSTEM programs & display load address range. \$20 (DSMBLR I for non-disk use is \$15)



MISOSYS - Dept U
5904 Edgehill Drive
Alexandria, Virginia 22303
703-960-2998 MicroNET 70140,310
Dealer Inquiries Invited



diskmod

Turn your Editor Assembler into a disk package. This 32K patch modifies EDTASM for DOS operation. Features? Add full disk I/O, block move, global change, printer pagination with optional prompting, sorted symbol table, print memory utilization, correct DEFM expansion, protect memory, and recover after BOOT. From within the EDTASM you will have DIR, KILL, & FREE. Upgrade your EDTASM today! Version for EDTASM+ coming soon. \$20.

THE B00K

THE B00K must be a part of your Z-80 language tools. Volume I gives you access to all math operations in your Level II ROM including ASCII-Binary conversions. Included is a symbol table of the entire machine noting over 500 addresses. Volume II tells you everything you wanted to know about the Level II I/O - printer, keyboard, video, and cassette routines are fully explained. Each volume has a fully-commented listing of all the routines discussed. THE B00Ks will save you hours of assembler program development time. Don't start programming without THE B00Ks. Each volume is priced at \$14.95 + \$1.50 S&H or buy both for \$24.95.

After booting your computer, a single character entry will cause you to quickly be at the main menu, ready to go.

This menu contains selections for entering all main areas of the system. Here are the selections available:

- 1 - Define Data Formats
- 2 - Define Screen Formats
- 3 - Define Report Formats
- 4 - Define Label Formats
- 5 - Define Selection Formats
- 6 - Expand Existing File

- 7 - Inquire, Update, Add
- 8 - Print Reports
- 9 - Print Labels
- A - Select Records
- X - Exit

Let's take a look at each of these options and see what function they will provide.

1 - Define Data Format

This function allows the user to specify a name for the files that will contain the data (in effect name the system). After naming the system you will now have to tell the system how each record is to be structured and the length of each item in the record. You will be specifying this for what the system calls "SEGMENT 1." This is a very important concept as all the pieces of data, by which you will want to locate a record, must be defined in this segment.

This segment of your data base may be

up to 85 characters in length and contain up to 36 different pieces of data. Each of these pieces of information, such as name, phone #, eye color, date of birth, etc. is called a "FIELD." Each of the fields you define is numbered by the system, for its internal use, and then given a name by you.

After defining what information will be contained in SEGMENT 1, you may then define data in up to 3 additional segments. Each of these additional segments may contain up to 256 characters of data. These segments are for storing information that you wish to have connected with the "KEY" data in SEGMENT 1. As you can see, one record may be 1, 2, 3, or 4 segments long, depending on the amount of data you wish each record to contain. The maximum record size for the system is 853 characters.

2 - Define Screen Formats

This powerful feature allows you to set up how the data fields (specified in menu option 1) will appear on the screen and how they will be labeled.

This is accomplished with the use of what can be termed a "full screen editor." With this editor you simply place the cursor where you want it and enter the labels you want for your data, followed by the number of the data field you want to be displayed.

You may have these labels in reverse video or regular video. You may also place bars and lines on the screen as well as

titles and headings. When you are satisfied with the display you have created, you simply tell the system to save the screen. That screen format will then be available for displaying your input data.

The data that will be displayed on your screen may be changed at the time a record is displayed and, of course, while adding new records. To let the system know what type of data will be accepted in a field you must tell it. You have the choice of alphanumeric (letters and numbers), numeric, decimal and protected. If you define a screen with one or more protected fields, those fields will not be able to be changed. In essence you have created a limited access screen.

There are two other field types which may be requested on a screen. They are for adding or subtracting to a data field. These input fields would follow a label like "AMOUNT RECEIVED:". The entered amount would then be added to or subtracted from the defined field.

You may have up to 5 different screens for each set of files you create. Each screen may be password protected so that unauthorized persons can only use the screens that you have allowed them access to, such as those with protected fields or screens which only display portions of the data record for clerical use. Complete, unprotected access can be given to persons with the proper password.

3 - Define Report Formats

This function allows you to set up printed reports that the program is to produce from your data files. Dating, headings, titling, pagination, page numbering and columns are all supported. This report formatter uses a video screen type editor, similar to that of the screen formatter. The entire 132 columns of a wide print out are viewable 80 columns at a time. All you need to do is move the cursor right and the rest of the form is pulled in as the leftmost portion begins to disappear. Setting up fancy looking reports proves to be a simple task on this report formatter. All you do is enter your title and heading data, then just place your field numbers in the appropriate places under the column headings.

The system allows you to store up to 5 different report formats for each data file set you create. Any one of the formats can be called for use to print selected data or be recalled for editing, to change that report.

4 - Define Label Formats

Label formatting is very similar to report formatting, except that headings and titles are not available, but are not needed. Label formatting is simply a matter of positioning the fields on the screen in the places that you wish them to appear on the label. Multiple labels across are supported but are configured when setting up for label print-out.

Again, you may make and save up to 5 label formats for each of your file sets.

5 - Define Selection Formats

This option allows for the extraction of specified data from specified records. This part of the extract system allows you to select the fields in the data that will be pulled out to make a new record. For example, if you have a file where each record contains fields 1, 2, 3, 4, 5, 6, 7 and 8, you may set up a selection format to take each record, meeting your criteria, and take out fields 3, 5 and 7 and place them in the new file as fields 1, 2 and 3. This option is what could be termed a "record restructuring formatter."

The resultant files from this portion of the system are created to be used by Model II SCRIPSIT, but are of a conventional structure, with an imbedded header defining the internal layout of this file (a nice touch from the programmer's standpoint). Any competent programmer will have no problem putting them to use in BASIC, COBOL, FORTRAN, or ASSEMBLER.

6 - Expand Existing Files

This is a very interesting feature. This option allows you to set the number of records that are available for use in any set of files you create, *and expand those files anytime you wish to allow for more records.* This is a very important option because it allows you to keep your files at the smallest usable sizes. Files that are properly sized to the amount of active records can be searched faster and therefore managed more efficiently.

7 - Inquire, Update, Add

This is the day to day activity option. It allows adding, searching, editing, and deleting. All editing of the records is done through a very nice screen editing system. Simply place the cursor on the data to be updated and change it. Then hit one key to put away the updated record and continue. Locating records for reference purposes is simple and fast.

8 - Print Reports

9 - Print Labels

These two options are the same except for the final output that will go to the printer. To get printed output you must first specify the file from which the data will come, and then the number of the label or report format you wish to use.

Now, to get a report or labels which are based on a certain group of records within your data base, you will have to answer the selection questions. These questions will let you select the records to be used in the report by any of the fields in SEGMENT 1 of your files. For selection purposes, full relational control is supported including; less than, greater than, equal to, less and equal, greater and equal, plus full "anding" and "oring" of different fields. Field selections may also contain a "I don't care" or a "wild character." So, if you had a date field (MM/DD/YY) and you want all records in which that field contains "79" dates, then you would enter "===== " on that selection, in effect masking the leading part of the field.

Sorting is also specified at this point. Sorting can be performed on any field in SEGMENT 1. The system will scan the entire data file to find all the records that meet the selection criteria that you have set up. Then it will proceed to sort the selected records and print those records in the format you have selected.

If all this sounds awfully powerful, it *should*, because it is!

A - Select Records

This option is used in conjunction with option 5, which was described earlier. It allows for selection criteria similar to the set up involved in printing, including the sorting selections. The system will extract the selections, sort them, reconstruct them, and place them in the new file.

E - Exit

This option is selected when you wish to leave the program and return to the operating system "TRSDOS READY."

K - Kill File Set

This option is not shown in the MENU but is available. After selecting this option you will be asked the name of the file set you wish to destroy. After giving the file name, the system will check with you once more to make sure that you wish to destroy this data, as this option has a very permanent function and is not reversible. After confirming your intentions the system will remove the data from the disk and the space will then be usable by other files.

Conclusions

This system is one of the best data management systems I have ever seen running on a Micro. Yes, there are systems that are better, but they cost thousands, not hundreds.

I strongly suggest that if you feel you may have the need for a computer, check out the TRS-80 Model II and its powerful data management partner PROFILE II. If you already are the owner of a Model II, pick up your phone and order the program from Radio Shack. You'll never regret it.

One parting note of thanks, to Howard and Bill at The Small Computer Co., in Ridgewood, NJ., for making this outstanding software package available to the TRS-80 user, also to Radio Shack, whose mass marketing capabilities have made the price so attractive, and lastly to Joe Sigler and Don Stanfield at Tandy, who made it possible.

Should you have or get PROFILE II, there is one thing you must do to get proper performance: the manual makes no reference to the TRSDOS 2.0 command called "VERIFY DETECT." At TRSDOS READY you should type (VERIFY DETECT OFF) and press enter. This will turn off the constant "disk changed? checking." PROFILE II will then be able to scan records about 5 times faster. But be aware that you *must* do the "I" for INIT, every time a disk is changed. When this checking is turned off, failure to do the "I" will most certainly result in disaster (lost files)! Be sure to have backups, just in case.

Below you will find ratings on PROFILE II as compared to the many other "data managers" I have seen for the TRS-80.

DOCUMENTATION	FAIR
(is not detailed enough for the novice)	
CONCEPT	EXCELLENT
EASE OF USE	EXCELLENT
RELIABILITY	EXCELLENT
FEATURES	EXCELLENT
SPEED	GOOD

Scans 1000 records:

in 14 seconds (VERIFY DETECT OFF)
in 66 seconds (VERIFY DETECT ON)

CAPACITY

up to 20,000, 85 character records with 4 drives, less with fewer drives or larger records.

SUPPORT	GOOD
PRICE	\$179.00
AVAILABILITY	GOOD

Meets Requirements

of a data base manager	GOOD
overall rating	EXCELLENT

AUTHORS NOTE:

It must be noted that there is a "PROFILE" product for the Model I TRS-80, also sold by Radio Shack. This entire article pertains ONLY to PROFILE II for the MODEL II TRS-80 and any similarities between the two are purely coincidental. They were not written by the same people and they do not function the same.

Bill Schroeder

Once upon a time

in the Land of Adventures, there was a King named Adams. King Adams said to himself, "My kingdom has Adventures for adults but what about my younger subjects?" So the mighty King went to his wise Knights of the C R Table with his plight. Lo and behold Sir Talley had the answer. Then the King asked, "But does it have sound effects, and graphics and can it be used by readers and non-readers alike?" King Talley replied, "Oh yes Sire and even more, it has both a story mode and a quiz mode."

"So be it!" said the King, we will spread the news throughout the kingdom and call it...

**Kid
Venture**

FOR 4-10 YEAR OLD ADVENTURERS

Now available from your
local Computer Lord
or send for prices
and additional
info to:

DEALER
INQUIRIES
INVITED

Adventure
INTERNATIONAL

Box 3435
Longwood, Fla.
32750
(305) 862-6917

© Copyright 1980 Adventure International

IF YOU LIKED SANTA PARAVIA,
ENJOYED GALACTIC EMPIRE,
YOU'LL JUST LOVE PROJECT OMEGA

TRS-80™



A PROGRAM PAINSTAKINGLY RESEARCHED
FOR A TRULY AUTHENTIC & STIMULATING
SIMULATION . . . YOU'LL JUST LOVE IT!

NEW

TRS-80 16K TAPE \$14.95 (Single Player)
TRS-80 32K DISK \$24.95 (Multi-Player)

NOW AVAILABLE AT
YOUR LOCAL
COMPUTER STORE
IF NOT, CALL
OR WRITE TO:

DEALER
INQUIRIES
INVITED

ai

Adventure

INTERNATIONAL
A DIVISION OF SCOTT ADAMS, INC.

BOX 3435, LONGWOOD, FLA. 32750 (305) 862-6917

The finest Data Base Manager Available

NEW *Maxi* **NEW**
**Micro
Manager**

MAXI MICRO MANAGER for TRS-80 Models 1 & 3
Requires 48K of RAM and 1 Disk Drive Minimum.

JUST CHECK SOME OF THE FEATURES

- * Supports six different relational search techniques.
- * Comes with programmer's interface.
- * Over 93 pages of documentation.
- * Supports up to 20 user defined fields.
- * Each field records up to 800 characters.
- * Files can be up to four disks in length.
- * Compatible to 35, 40 & 77 track drives.
- * Has calculated equation fields.
- * Complete report generator.
- * Data can be merged into letters.

And much, much More!

REGULAR
PRICE
\$99.95

SPECIAL
INTRODUCTORY PRICE **\$79.95**
Offer Expires 1/31/81

NOW AVAILABLE AT
YOUR LOCAL
COMPUTER STORE
IF NOT, CALL
OR WRITE TO:

DEALER
INQUIRIES
INVITED

ai

Adventure

© Copyright 1980

INTERNATIONAL

BOX 3435, LONGWOOD, FLA. 32750 (305) 862-6917

Playmate of the Year

Don Scarberry
D & M Software, Tacoma, WA

A few months ago I experienced the pleasure of playing with one of the many games presently available. I was never very enthusiastic about playing games on my computer but, after seeing one of the latest versions in action I decided to give it a try. To my delight, I quickly discovered the one I had chosen came complete with sound. All that was necessary was to purchase an inexpensive audio amplifier and connect it to my TRS-80. Instantly my computer came to life. The sound effects brought a new dimension to game playing. I could now, not only see and touch my computer but, hear it also. As everyone knows the human maintains contact with the real world through its five senses - sight, touch, hearing, smell and taste. Well, my TRS-80 was really teasing three of these senses now. Sooner or later all good things must come to an end. I got bored. Sound effects were simply not enough of a challenge for my computer. So, armed with the knowledge that Radio Shack was offering a Voice Synthesizer I began a long journey in search of one of these nearly extinct creatures.

Many days and stores later I still had not found a synthesizer. All stores offered to special order one for me but none had one in stock and even fewer had any meaningful information pertaining to specifications and operation. Enough of this I thought. My travel costs already had me invested in a synthesizer and I hadn't even seen one yet. I might as well have been searching for Sasquatch. My desire for a talking computer quickly turned to lust. Finally I got smart. I let my fingers do the walking.

Success! A little ten square foot store down by the railroad tracks had one in stock. The clerk told me I had better hurry if I wanted to see it because the manager was going to send it back. It seems that no one was interested in a nearly extinct little box that did nothing but talk all the time. So I ran down as quickly as I could (actually, I took my car) fearing that I may never get another chance to observe one of these rare critters before they left the face of the earth. "Where is it?" I panted to the clerk as I rushed into the store. He pointed a crooked, skeleton-like finger at a little silver box clinging nervously to the TRS-80 on display. I tip-toed, quietly, toward it. After all the trouble I had locating it I didn't want to scare it away. It didn't make a sound. I poked a finger gently in its side. Not a peep. "It's dead," I shrieked. "Not really," the clerk replied. "It isn't hooked up and no one

here knows how to program it. Besides, there isn't any demand for these things so we're going to send it back and if anybody wants one we'll special order it." I was no fool. This thing wasn't responding to my attention because it hadn't been taken care of properly. All it needed was someone to give it the proper care and before long it would be jabbering away. I bought it and took it home.

The hook-up procedure was a snap. After plugging in the electrical cord to a 120 volt outlet and the ribbon cable to the parallel interface jack on the expansion interface my computer and I were ready to converse with each other. There was but one minor problem. We didn't speak the same language. All the BASIC language I had mastered was for naught. If I was going to carry on a conversation with my new pet I would have to learn its language. Naturally it wasn't going to learn mine. Well it didn't and I did.

About two months later I was speaking to my computer and by golly, it was talking back. It took a lot of effort on my part, to learn the phonetic language but I can honestly say it was well worth it. I no longer had a nearly extinct creature on my hands but an almost human S-80, with one big eye in the middle of its forehead, an oversized set of vocal cords, and a whole lot to say. This thing just wouldn't shut up. Radio Shack has the marvel of the century here and hardly anyone is even aware of it.

A little about phonetic programming might prove helpful at this point. Practically any word in the English language can be produced using phonemes which the voice synthesizer is capable of producing. Phonemes are simply small units of sound which when combined properly will form the proper sound of a complete word. There is a phoneme for virtually any part of any word. Specific words, sound durations, pauses, pause durations, stop plosives, fricatives, nasals, vowels and semi-vowels are all phonemes the synthesizer is capable of reproducing.

The phonemes themselves are not used to produce the sounds. These are merely symbolic representations. For each phoneme there is a ASCII character on the keyboard of the computer that actually creates the sound. So to cause the synthesizer to say a word one must create a set of phonemes, representing that word, then type in the ASCII characters, which represent the phonemes used. The ASCII

characters are then sent to the synthesizer and the word is spoken. The whole process is fairly straight-forward except for one thing. The phonemes, in order to get to the synthesizer, must first pass through a little window.

The window is located at the last 32 print locations on the video display. Before sending phonemes into the synthesizer this window must first be opened. We can open it by printing or poking a "?" anywhere in the window. Once open we can send our phonemes to the synthesizer and the word will be spoken. When the transmission is complete we close the window to prevent accidentally sending meaningless information through it. The whole process, as described, sounds rather complex but if one perseveres for a few hours, then intelligible conversation will be the reward. Practice, however, is needed to ultimately have the computer carrying on intelligent, articulate, conversation.

There are some disadvantages to Radio Shacks' method of sending information to the synthesizer. If whole phrases are passed continuously, the synthesizer may begin to stutter and make sounds like a dying frog. For this reason the words, phrases or sentences must be broken up into shorter ones and timing loops inserted between them. Sentences that are too long may overflow the buffer and the characters that cause the overflow will be lost. Ideally one should limit the sentences to not more than 32 characters. Longer sentences must be broken up into smaller blocks using the timing loops to avoid discontinuous speech.

As stated earlier, the synthesizer will pronounce "most" words of the English language. After extensive practice I have discovered a few words that are very difficult for the synthesizer to pronounce clearly. For example, words involving nasal sounds, such as those that end in "ing" are very difficult to understand even though the proper phonemes might have been utilized. This type of problem is not a major drawback however, since the English language contains many words with like meanings but different spelling. So, if a word is particularly difficult for the synthesizer to utter clearly, one can simply choose another word with like or similar meaning.

One additional disadvantage I have found is the lack of a tone control. The sound radiating from the speaker is rather deep and lacks presence. Because of this, and differences in hearing ability between

Our new program package for the TRS-80™ sounds terrific.

So does the price.

There are lots of programs with sound that are worth about a dollar. Trouble is, they cost a lot more.

But at Basics & Beyond we've just developed Microcosm III, 20 programs with sound—each just as good as our competition's \$15 and \$20 programs—for \$24.95. That's a 20-program package for \$24.95.

It includes "Pinball," replete with ringing bonuses, spinners, buzzers and flippers; torpedo-firing "Submarine" that explodes with underwater excitement; and the right/wrong buzzer in "Long Division" teaches step by step.

At Basics & Beyond we underscored our point that most other program packages are overpriced with Microcosm I and Microcosm II, \$19.95 each. Now a lot of people will start hearing about our third package and stop listening to high prices.

You see, it's not that our program packages for the TRS-80™ microcomputer are so cheap. It's just that theirs are so expensive.

BASICS & BEYOND, INC.

Box 10 • Amawalk, N.Y. 10501 • Or call 914-962-2355
Mastercharge and Visa accepted.

No charge for postage or handling. N.Y. residents add 5% sales tax.
TRS-80 is a trademark of the Radio Shack division of Tandy Corp.

QUILL DRIVER is here!

QUILL DRIVER is the latest in text formatting on the TRS-80. Allows use of your Basic Editor, the EDTASM Editor, or several others to create text files with various control commands for:

- Right justification
- Pagen
- Right justification
- Pagination
- Headers and Footers
- Alternate input files to handle lengthy documents and form letters
- Processing of printer control codes
- Centering of titles and subtitles
- Accommodates 'problem formats' such as 'hanging indents' and outlines
- Conditional paging to allow room for tables and charts
- Vertical and horizontal formatting
- AND MORE! 33 commands to provide the versatility you want and need!

QUILL DRIVER comes complete with a lower case, keyboard printer driver. Text can be prepared without a lower case mod (although not visible on screen). DOS, 32K and one drive required. Includes 50+ page manual (available separately for \$5.00). QUILL DRIVER comes on diskette packed with sample files and info! Sorry, diskette is formatted only, so make arrangements to transfer to system diskette if you have but one drive.

SPECIAL OFFER: DISK*MOD from Misisys provides dozens of fixes to Radio Shack's EDTASM, including: DIR, KILL, FREE — all from EDTASM command mode; improved disk I/O routines and the 'Find and Replace' option! Order with QUILL DRIVER, and we'll provide FREE another diskette containing more patches, including protection for Drivers that 'protect themselves' (such as the one included with QUILL DRIVER!) and one to allow lower case input into EDTASM! **DISK*MOD is \$20.00.**

Sounds reasonable to me. Send me the following:

☐ QUILL DRIVER @ \$39.95 ☐ DISK*MOD @ \$20.00

MAIL TO:

The Alternate Source
1806 Ada Street
Lansing, MI 48910

SHIP IMMEDIATELY TO:

Name _____
Address _____
City _____
State, Zip _____

☐ What else you folks got?

Phone Orders: 7 days, 24 hours, 517/487-3358, Visa and Master Charge
MICHIGAN RESIDENTS ADD 4%.

☐ By the way, I'm hearing more and more about your 'Alternate Source' newsletter. Sign me up for that too. Here's \$9.00 for six issues.

people, it may be difficult at times to "catch" exactly what is being said. I have noticed however, that the longer I use the synthesizer the more "tuned in" I become and the easier it is for me to understand. Understanding the synthesizer is analogous to understanding someone with a different dialect. The longer one is exposed to the others manner of speech the easier it is to understand.

Probably the most significant disadvantage is the rather skimpy instruction manual included with the synthesizer. One shortcoming that becomes almost immediately apparent is the meager 33 phonetic word samples and phrases that are provided. Many new computer owners are still struggling with learning basic programming skills and hardly have the time (or patience) to learn the phonetic language required to get their synthesizer up and talking. A dictionary of English to phonetic conversions would prove extremely helpful in this area.

The disadvantages I've mentioned should not overshadow the fun, excitement and practicality of owning a voice synthesizer. Eventually, I anticipate all computers will talk. A computer without voice is like silent movies are to television.

With a voice, the computer seems to come alive. A very curious thing has come to light while using my synthesizer. My computer

tends to take on my personality through my programs. I have my own skills, habits and idiosyncrasies and so my personality traits are exhibited when the programs are running. I became more acutely aware of this while running one of my talking programs and my wife began laughing hysterically.

After gaining control of herself and wiping her eyes she exclaimed, "It sounds just like you!" You see I have a southern accent and when I write the words for the synthesizer I first sound them out. When the computer says them the way I feel they should sound I put my stamp of approval on them. The end result is a talking computer that has a slightly robotic voice and a southern drawl. Can you imagine the voice of a computer using a program written by a German Professor?

I could go on and on describing my overall delight with my computer since I purchased the synthesizer but, this article must end somewhere. Add a synthesizer to your system and you'll open up an entirely new dimension in computer-human relationships. Is your eyesight failing? Are you blind? These handicaps need no longer prevent you from interacting with a computer. Your computer can now become your new playmate - almost. Too bad it doesn't have a female voice!

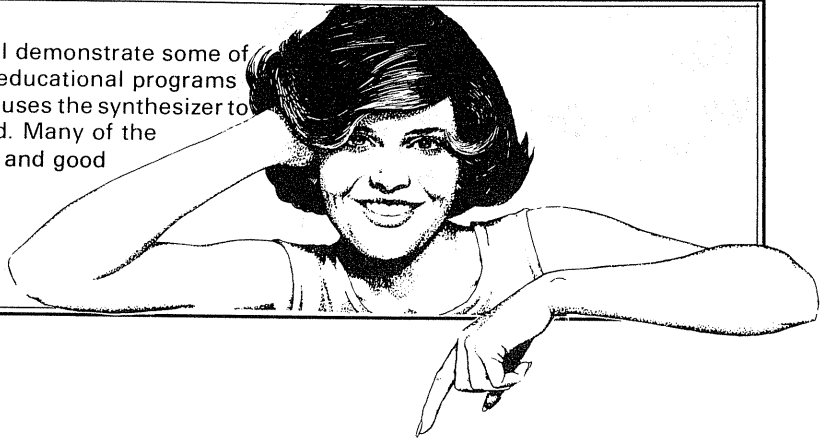
Benefits of using the synthesizer are many. Programs can be written for educational use, where the computer

can give quizzes orally. In fact, the use of the synthesizer allows one to write programs for any area that demands oral output. Most spelling programs, for example, are actually feeble attempts to simulate the "old fashioned" spelling bee. With the synthesizer, however, a true spelling bee program can be written. The computer might offer a group of study words for practice, first on the screen, then orally administer a quiz. This seems to be the best approach by far.

How about a talking alarm clock? Can you imagine having your computer persistently nagging you awake each morning. Need a talking card partner? Tired of telling bedtime stories to the kids? Let the computer do it. How many times have you turned away from the computer to wait for a program to finish execution? Wouldn't it be nice to have it tell you, orally, that it was finished. Imagine how much more fun all those "adventure" games would be with actual voice output. And if the computer begins talking too much, you can simply pull the plug.

This article is directed primarily to those individuals (like myself) who have wondered if buying the synthesizer would be worth the rather high selling price. I can honestly say, "you better believe it." It's like buying a whole new computer. And, I might add, you'll find your computer has a great deal more to say than what you've been seeing on the screen. ●

Here is a very simplistic program which will demonstrate some of the techniques used to obtain high quality educational programs using the Voice Synthesizer. The program causes the synthesizer to repeat the ABC's over and over without end. Many of the techniques needed to obtain clear verbiage and good program flow for educational use are demonstrated here.



```

5 CLEAR 1000
6 CLS
8 'WRITTEN BY D & M SOFTWARE
10 DIMQ$(26),P$(26),ZY$(26)
20 B1$="H38L8CU"
21 B2$=";EAMNOUM99D"
22 B3$=";E33MMYORTET>RR"
23 B4$="L9TSL/NN ;UR @EEBEESEES"
24 B5$="HERGOUSN6=IN"
36 BA$="BEEP BEEP"
50 FOR X=1 TO 26
55 READ Q$(X)
60 NEXT X
65 FOR Y=1 TO 26
70 READ P$(Y)
75 NEXT Y
100 GOSUB25010
1000 GOSUB10000
5000 END
10000 CLS:PRINT CHR$(23)
10005 ZY$=B1$:GOSUB21000
10010 PRINT@474,"HELLO"
10015 ZY$=B2$:GOSUB21000
10020 PRINT@532,"I AM NOMAD"
10025 ZY$=B3$:GOSUB21000
10030 PRINT@590,"I AM YOUR TEACHER"
10040 FOR Z=1 TO 1000:NEXT:CLS:PRINT CHR$(23)
10050 PRINT@4,"LET'S LEARN OUR A B C'S"
10060 PRINT STRING$(31,"-")
10070 ZY$=B4$:GOSUB21000
10075 ZY$=B5$:GOSUB21000:FOR Z=1 TO 500:NEXT
10080 FOR X=1 TO 26
10100 ZY$=P$(X):GOSUB21000
10105 PRINT@542,Q$(X)
10110 NEXT:FOR Z=1 TO 300:NEXT
10115 CLS
10120 GOTO 10040
19999 END
20000 POKE 16383,63:POKE 16383,32
20010 FOR ZX=1 TO LEN(ZY$)
20020 POKE 16383,ASC(MID$(ZY$,ZX,1))
20030 NEXT ZX
20040 POKE 16383,32:POKE 16383,63:POKE 16383,32
20050 RETURN
21000 FOR Z=1 TO 1000: NEXT: GOSUB 20000: RETURN
22020 END
25010 CLS
25020 FOR X=1 TO 5
25030 GOSUB 25080
25035 ZY$=BA$:GOSUB21000
25040 PRINT @ 453,CHR$(23)"TURN ON SYNTHESIZER"
25050 GOSUB 25080
25060 CLS
25070 NEXT X
25080 FOR Z=1 TO 150:NEXT
25090 RETURN
26000 END
27000 DATA A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z
27050 DATA @EE,BEE,SEE,D!EE,EE,9FF,DJEE,@ETC,;YY,DJ@E,K@E,9LL,9MM,9NN,OUU,0PEE,KY
((U,1R,9SS,0TEE,Y(UU,VEE,D18LY(U,9KSS,W;E,ZEE

```

**For Model I, Level II
16K and up with
Synthesizer.**

Byte Miser Software

Presents

DATORG by James F. Williams

author of ASPTCH

DATA ORGANIZER is an extremely flexible, fast, and memory efficient file keeping system designed especially for tape based TRS 80 users. (A Disk or Stringy Floppy version that has all the features of the tape version plus Disk or Stringy Floppy I/O is also available.) It combines the speed of machine language routines with the ease of program modification of BASIC.

The machine language routines are:

- Write, Verify, Read, and Merge tape files (many times faster and more reliable than BASIC's)
- Sort forward or backward by any occurrence of any delimiter (space compression codes and line feeds included)
- Search the file for strings of any length with any number and combination of "don't care" characters
- Key Debounce
- Line Input
- CLEAR (more than the 32767 bytes BASIC allows)
- String Packing (averages many times faster than BASIC)

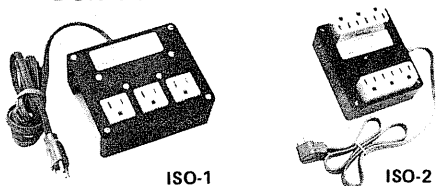
The BASIC program allows you to control these routines and:

- Set memory usage
- Edit (Global editing included)
- List
- Keep a running subtotal of one numeric field per line
- Have optional printer output from Lists or Searches
- Display remaining memory

DATORG can be the core of almost any file keeping requirement you have. Checkbook, Index, and mailing list just scratch the surface. Send only \$20 for tape or Stringy Floppy version (L2 16K and up) or \$25 for Disk versions (32K and 48K included) to Byte Miser Software, 720 W. Haven Blvd, Rocky Mount, NC 27801.

Also available-->CPYALL (--Make backup copies of almost any L2 format tape. Only \$7.00!!!

DISK DRIVE WOES? PRINTER INTERACTION? MEMORY LOSS? ERRATIC OPERATION? DON'T BLAME THE SOFTWARE!



ISO-1

ISO-2

- Power Line Spikes, Surges & Hash could be the culprit! Floppies, printers, memory & processor often interact! Our unique ISOLATORS eliminate equipment interaction AND curb damaging Power Line Spikes, Surges and Hash.
- *ISOLATOR (ISO-1A) 3 filter isolated 3-prong sockets; integral Surge/Spike Suppression; 1875 W Maximum load, 1 KW load any socket \$56.95
 - *ISOLATOR (ISO-2) 2 filter isolated 3-prong socket banks; (6 sockets total); integral Spike/Surge Suppression; 1875 W Max load, 1 KW either bank \$56.95
 - *SUPER ISOLATOR (ISO-3), similar to ISO-1A except double filtering & Suppression \$85.95
 - *ISOLATOR (ISO-4), similar to ISO-1A except unit has 6 individually filtered sockets \$96.95
 - *ISOLATOR (ISO-5), similar to ISO-2 except unit has 3 socket banks, 9 sockets total . . . \$79.95
 - *CIRCUIT BREAKER, any model (add-CB) Add \$ 7.00
 - *CKT BRKR/SWITCH/PILOT any model (CBS) Add \$14.00

PHONE ORDERS 1-617-655-1532

ESP Electronic Specialists, Inc.

171 South Main Street, Natick, Mass. 01760

Dept. 8U

WHEN OUR PROGRAMS TALK PEOPLE LISTEN

Radio Shack has the machine to make your computer talk. We have the software!

Until recently there has been little support for the TRS-80® Voice Synthesizer. At D & M Software we feel that within two years virtually all computers will talk and for this reason we have instigated a crash program to develop as many quality programs as possible to fulfill the projected explosive demand by educators and businesses.

Sound effects are nice but a talking computer is sensational! So, if you own a synthesizer or are considering purchasing one then consider our support. We offer a whole new dimension in interactive programming for education, business, or fun.

SPELLING BEE (A 3 PART PROGRAM)

The computer administers a lesson consisting of ten words. When the student has learned them he is given an oral quiz in true spelling bee fashion. When the lesson has been completed student may advance to next lesson. The program is structured so that an unlimited number of lessons can be given with progressively greater degrees of difficulty. Excellent for any grade or age level. Includes ten sample lessons of ten words each. The second part of the program allows one to create his own spelling lessons on cassette for much needed flexibility. A third part enables one to practice creating new words using the phonetic spellings needed by the synthesizer. Error trapped and bug free.

MULTIPLY MOD-1 16K L-II \$19.95

An interactive, vocal, adventure in arithmetic. The program offers a list of four programs which include multiplication, division, addition and subtraction. Lessons are printed on the screen and oral tests are administered. Positive reinforcement is given the student as he progresses. Scores are kept and grades given orally as the student and computer work together toward a better understanding in mathematics.

L-II 16K \$19.95

CROSS REFERENCE DICTIONARY (OVER 5000 LISTINGS)

The largest known cross reference manual for use with the TRS-80® Voice Synthesizer. No need to spend more long hours learning another computer language. Want your computer to say a word? Look up the English spelling. Right next to it you'll find the phonetic spelling that your synthesizer needs to say it correctly. Included is a large list of commonly used phonetically spelled phrases and sentences ready to type into your program. Invaluable as a time saving device.

D & M SOFTWARE \$14.95

1510 SOUTH 97TH

TACOMA, WASHINGTON 98444

DOES YOUR TRS80* DESERVE THE VERY BEST SOFTWARE? EDUCATIONAL

MATH-PAK-1 [] **MATH-PAK-2** - Interactive math drill programs that let the user enter the answer digit by digit, just like paper and pencil. With user selected difficulty levels, carryovers, remainders, simplification, reducing, games as rewards, scoring, and more. The MATH-PAKS are used throughout the U.S. and Canada in schools and are recommended by the 80 U.S. Journal, S-80 Bulletin, and The Computing Teacher. (L2-16K) Order MATH-Pak-1 FOR integers, MATH-PAK-2 for fractions. \$14.95 ea.

HOME/BUSINESS

H-O-R-K-S - Low cost, single entry accounting system that works. With 66 user assigned account codes, auto audit trail, search with totals, 32 or 48K, 1 to 4 drives, credit and debit summaries with 3 formats, up to 9200 complete entries, plus 8 pages of documentation. (32K 1 disc minimum) \$24.95/cassette \$29.95/disc

INVENTORY + - Why settle for just an inventory listing? Get aging reports with 2 options, reorder reports, total listings with purchase dates and amounts, total cost of inventory, items sold, total profit, and more. Do day to day updates, change and delete items, and pack files. Includes printer routines and documentation. (32K-1 disc minimum) \$24.95/cassette \$29.95/disc

CASH REGISTER 80 - Use your TRS-80* as a point of sale terminal with auto inventory lookup, auto pricing, auto inventory update, optional discount pricing and taxing, print sales slips with user adjusted formats, and get end of day reports with all cash, check, and charge sales by salesperson and a listing of all non-inventory items sold. CASH REGISTER 80 requires INVENTORY +, 48K, and 1 disc minimum. (2 discs recommended) \$24.95/cassette \$29.95/disc

GAMES

PYRAMIND - A challenging game of logic and deductive reasoning. Can you solve the computer generated code? The computer will give you clues on each entry but it will take all your skill to break the code. With 3 difficulty levels, sound, high speed graphics, and save a game on cassette or disc. (L2-16K) \$11.95/cassette

See your local dealer or
send check or M.O. to:

EDU-WARE
P.O. BOX 366
MAYNARD, MA 01754

Dealer and education inquiries invited.

Mass. residents add 5% tax.

Ask about our cassette duplicating service.

*TRS-80 is a registered trademark of the
Tandy Corp.



Byte Miser Software
720 West Haven Blvd.,
Rocky Mount, NC 27801

Disk (\$25.00)

Tape or Stringy Floppy (\$20.00)

Data Base Management systems have recently become a dime a dozen, literally. Nearly everyone is getting into the act. It's a sign that the TRS-80 user is growing up. At first, the personal computer was a "Gee Whiz" phenomena. Games were popular and little was done to make the system really useful. The few practical programs were forgotten in the desire to play new and better games.

Since games have begun to grow stale for many people (how many times can you play Star Trek without growing bored stiff?), practical programs have begun to improve in popularity. DATORG is such a program.

The problem with most data base manager programs intended for the small system, such as the Level II 16K, is that they are too slow to be really useful and they take forever to store and load

information. DATORG moves beyond all of this to give us a truly useful small computer system in a very simple package.

In order to improve the response of the program, the critical portions of the program are coded in an Assembly language module. These functions include write, verify, read and merge tape files; file searching and sorting; memory Clear; compressed space code input and output; and string packing. Pretty impressive for a small package.

One of the really unique features of this package is that the high level control functions are in a BASIC program that the user is free to modify to customize the package as needed. A page of instructions is provided to help the user modify the program.

Let's see what the package can do.

Functions

1. Append - Creates or extends the file.
2. Edit - Edit, delete or replace a line or range of lines. The editor has some powerful features such as a replacement editing feature that will replace a given string with another you specify.
3. List - List a range of lines or the whole file. Using the system's printer toggle, you can direct output to the printer and with the Subtotal toggle, a subtotal of the numeric values in the first field of every record output is displayed.
4. Sort - Sort the field or part of it into order.
5. Search - Search the file or a part of it for the occurrence of a string.
6. Write - Write the file to cassette. A verify option is included that, like CLOAD? will verify the dump to tape.
7. Read - Read in the data file from cassette.
8. Merge - Combines files up to the limit of your memory.
9. Mem - string handling functions.

The good and the bad

Like every other program, there are good and bad points. To the good, I found the system relatively easy to use and fast compared with programs written all in BASIC. During the testing period, no errors in the handling of the program were observed.

To the bad, I didn't like the prompts for information that well. More importantly, it has always been my feeling that the computer should do all the real work, so I feel put upon when I have to do the computer's work by padding out numbers with zeros so they will sort correctly.

For many people, a more important limitation is that there can only be one numeric subfield for subtotalling and it must be the first on a line. Even this isn't such a terrific limitation for small applications.

The system is limited in what it can handle by the limitations of strings in BASIC. Since everything is apparently stored as a string by the program, it is necessary to keep your information to a size smaller than this.

Overall, I was impressed by the ease with which I learned the system, the speed with which it operated, and the fact that I didn't run into any errors during my use of the system. I will have to point out to be fair that I have not put the program into day-to-day use since it is too limited for the rather complicated data base management tasks I use my computer for, but two years ago, I would definitely have taken this compared to Radio Shack's In-Memory Management Program which I did buy and almost never use because it's just too cumbersome for my needs.

DATORG is available from Byte Miser Software, 720 West Haven Blvd., Rocky Mount, NC 27801. It comes in versions for Disk (\$25.00), or for Tape or Stringy Floppy (\$20.00). T.R.Dettmann

AT LAST!

Mass production prices on this high quality software. Buy direct and save 50%. Now, also available for CBASIC on CPM and MBASIC on HEATH HDOS.

DATA BASE MANAGER

Mod-I \$69 Mod-II \$199

You can use it to maintain a data base & produce reports without any user programming. Define file parameters & report formats on-line. Key random access, fast multi-key sort, field arith., label, audit log. No time-consuming overlays. 500 happy users in a year. Mod-II version has over 50 enhancements including 40 fields max. 'IDM-M2 is great!' - 80-US.

A/R

Mod-I \$69 Mod-II \$149

Invoices, statements, aging, sales analysis, credit checking, form input, order entry. As opposed to most other A/R, ours can be used by doctors, store managers, etc.

WORD PROCESSOR

Mod-I \$49 Mod-II \$49

Center, justification, indentation, page numbering. Mod-I version features upper/lower case without hardware change!

MAILING LIST

Mod-I \$59 Mod-II \$99

The best! Compare and be selective. Form input, 5-digit selection code, zip code ext., sort any field, multiple labels. Who else offers a report writer?

INVENTORY

Mod-I \$99 Mod-II \$149

Fast, key random access. Reports include order info, performance summary, E.O.Q., and user-specified reports. Many have converted their inventory system to ours!

GL, A/R, A/P, & PAYROLL

Mod-II \$129 each

Integrated accounting package. ISAM, 100+ page manual, Uses 80 column screen, not 64. A \$1,000 value. Dual disk required.

L216, a cassette package of 10 business programs for Level II 16K systems, \$59. Includes word processor & data base. Poker game \$19.

Most programs are on-line, interactive, random access, bug free, documented and delivered on disks. Mod-I programs require 32K TRSDOS. Don't let our low prices fool you! If still not convinced, send SASE (28¢) for catalog.



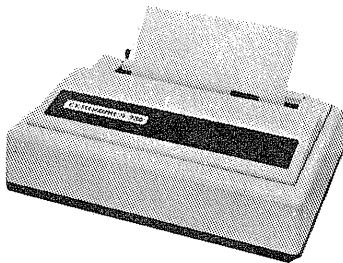
MICRO ARCHITECT, INC.
96 Dothan St., Arlington, MA 02174

MODEL II



26-4002
64K 1 Drive
\$3440.00

26-4160 1 Drive EXP . \$1035.00
26-4161 2 Drive EXP . 1575.00
26-4162 3 Drive EXP . 2115.00
26-4501 Gen. Ledger . . 180.00
26-4502 Inventory . . . 180.00
26-4503 Payroll 360.00
26-4554 Acct. Rec. . . . 180.00
26-4701 Fortran 270.00
26-1157A Daisy Wheel . 2495.00
26-1158 Daisy Wheel II 1799.00



CENTRONICS

Fast 100 CPS Centronics
730 Printer \$659.00
Text Quality Centronics
737 Printer \$819.00

Model II Cobol Compiler
\$360.00
Cobol Run Time Package
\$36.00

\$ DISCOUNT \$ TRS-80® DEALER

COMPUTER SPECIALISTS

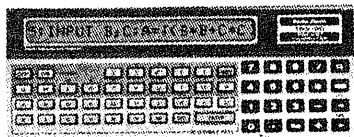
26-1155 Quick Printer II \$187.00
26-1145 RS-232 Board 84.00
26-1140 "O" K Interface 249.00
26-1141 "16" K Interface 359.00
26-1142 "32" K Interface 469.00
26-1160 Mini Disk - Drive O 419.00
26-1161 Mini Disk - Additional 419.00
26-1154 Lineprinter II 699.00
26-1156 Lineprinter III 1799.00
26-1159 Lineprinter IV 859.00
26-1166 Line Printer VI 1080.00
26-1563 Scripsit - Disk 79.00
26-1566 Visicalc 83.00
26-1562 Profile 72.00

NOTE: Call for availability of VIDEO TEX, Model III, Color, and other new products.

ALL OTHER R.S. SOFTWARE
FURNITURE, STANDS, CABLES
AND ACCESSORIES AT
DISCOUNT FROM
CATALOG PRICE.

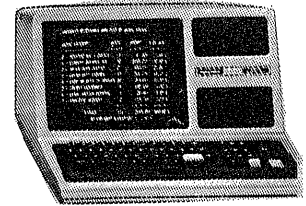
Novation Cat Modem . \$149.00
CCA Data Management
System 72.00
Adventure Games
Games 1-9 each 14.00

Pocket Computer



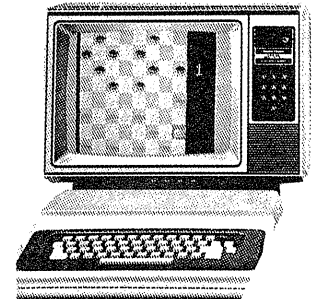
26-3501 1.9K P.C. \$225.00
26-3503 Cassette I/F 45.00
14-812 Recorder 72.00

MODEL III



26-1061 4K I \$630.00
26-1062 16K III 888.00
26-1063 32K III
2-Drives, RS232 2225.00

COLOR



26-3001 4K \$360.00
26-3002 16K 540.00
26-3010 Color Video 360.00
26-1206 Recorder 54.00
26-3008 Joysticks 22.50

 **Acorn**
Software
Products, Inc.

GAMES:

Alien Invasion \$9.00
Stock Market 9.00
Star Trek 9.00
Block 'Em 9.00
Ting-Tong 9.00

UTILITIES:

System Savers 14.00
EDUCATION:
Language Teacher 18.00

**FREE: COMPUTER CATALOG
UPON REQUEST**

1-800-841-0860 Toll Free Order Entry
MICRO MANAGEMENT SYSTEMS, INC.

No Taxes on Out Of
State Shipments

Immediate Shipment
From Stock on Most Items

DOWNTOWN PLAZA SHOPPING CENTER
115 C SECOND AVE. S.W.
CAIRO, GEORGIA 31728
(912) 377-7120 Ga. Phone No.

R.S. 90 Day Limited Warranty
F-48 Form Provided

Largest Inventory
In the S.E. U.S.A.

*TRS-80 is a registered trademark of the Tandy Corp. 80-U.S. JOURNAL Jan/Feb 1981 105

View (from the Top of the Stack)

Until this point, we've concentrated on learning how to use a few of the fundamental statements in assembly language, but now it's time to start really doing something with it.

Believe it or not, you now know enough to start making interesting programs using your knowledge of assembly language. However, to really be able to put together some interesting things, we need to do better at input and output.

In this column and the next, we'll learn how to get information into and out of your assembly language programs in a very simple way. This issue, we're going to cover the use of a few of the Level II ROM routines in order to input and output information.

Next issue, the topic will be using BASIC along with machine language to get useful things done. The objective in the next issue will be to make your assembly language programming skills useful to you when you are programming in BASIC.

After we cover input and output using these techniques, we will concentrate on learning more assembly language programming while we put together projects which do more than add two numbers in a register. Emphasis from now on will be on doing something. We'll add to our knowledge a little at a time as we put together simple projects.

Eventually, we'll come back again to input and output techniques, learning how to do them without either BASIC or the ROM. But first, let's get something going that will do something.

Why use the ROM?

A very good question to ask at this point is why use ROM? To that I can give a very short answer, because we can get input and output faster this way than any other *and* we can keep our programs very short by using the ROM to handle some of our more frequent needs such as inputting and outputting information.

ROM CALLS

The routines we will be using are accessed from the ROM by using the CALL instruction we learned about last time. As you remember, by telling the computer to CALL an address, we are putting the address in the Program Counter on the stack and putting the address CALL'ed in the Program Counter. When a RET instruction is found, the return address is POP'ed off the stack and into the Program Counter and execution continues from the point right after the CALL.

We'll 'Piggy-Back' on Microsoft by using some of their more useful routines to get what we want. At some later time, we'll talk about other CALL's to ROM (hopefully including CALL's that will be applicable to the MOD III).

The CALL's that will be of most use now are the following:

Input Routines

002BH

A CALL to this location will return the ASCII value of the key pressed in the A register. It does not wait for a key to

be pressed and if there is none, it will put OOH in the A register. The AF and DE register pairs are used by this routine.

0049H

A CALL here waits for a key to be pressed and otherwise works the same as the CALL 002BH.

005D9H

If HL points to an input buffer, and B has the maximum length of that buffer, then a CALL 05D9H will input to the buffer and echo the characters to the screen.

00361H

Inputs a line from the keyboard until either [BREAK] or [ENTER] are pressed. Keys echo to the screen. Uses a buffer whose address is at location 40A7H and whose length is FOH bytes. The string is terminated with a zero byte after entry is complete.

1BB3H

Outputs a "?" prompt to the screen and then jumps to 0361 to complete the input.

Output Routines

0033H

A CALL to this location will display the character corresponding to the ASCII value in the A register at the current cursor position.

01C9H

This CALL clears the screen.

28A7H

The location of the string buffer is put into HL, then a CALL is made to this location. All characters in the string (until either a 00H byte or a 22H byte) are put on the screen starting at the current cursor location. If the byte at location 409CH is -1, then the output is redirected to the tape output port. If the byte is +1, then the line goes to the printer port. If the byte is 00H, then it goes to the screen.

Now that we know where to find these ROM routines, let's try and make them work.

A Short Application

Let's write a short program as follows:

1. Clear the screen
2. Prompt for the input of a name
3. Input and display the name
4. Write the name back to the screen
5. Prompt for [ENTER] to do it again or [CLEAR] to end the program
6. Wait for an input character
7. If the character was 13 (ENTER), then go back to step 1
8. If the character was 31 (CLEAR), then return to DOS or Level II
9. If the character was neither of the two, then repeat again from step 6

Try to convert this simple design into a program. Next issue, we'll print a program that does this using the ROM calls, and then go into some ways of getting information to your program with USR CALLS from BASIC. ●

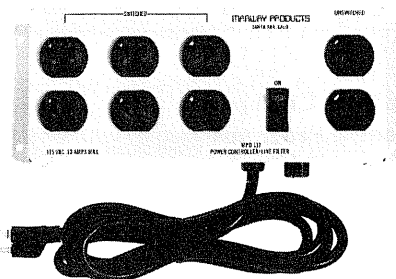
The 80-INDEX

- * Unlock the reference cabinet on your shelf. Find programs, tips, reviews, ads in seconds.
- * 700 item index for 80-US, 80-Microcomputing and KILOBAUD. Selected for the disk system owner.
- * Two superfast programs. SCAN by keyword, and SEARCH by description.
- * Complex enquiries linked by AND, OR, NOT are entered in seconds - and answered in seconds!
- * Requires 48K Model 1 with 2 disks.

Programs + Index + Manual US \$29 (2 Diskettes)
Manual available separately US \$8

HEXAGON SYSTEMS

P.O.Box 397 Stn. A,
Vancouver B.C. Canada
(604) 582-7646



MPD 117

NO "GLITCHES", SURGES OR INTERFERENCE

Clean power distribution for your:

- Microcomputer system
- Audio Visual system
- Lab instrumentation

Marway Products' new MPD 117 AC Power Control upgrades a standard AC utility outlet into a convenient noise-free source of AC power. The MPD 117 features:

- High energy EMI filter
- Transient voltage suppressor
- 10 amp circuit breaker
- 6 switched outlets
- 2 direct (unswitched) outlets
- Illuminated "on/off" switch
- UL recognized components

At \$89.00 (plus tax and shipping) the MPD is the low-cost solution to your power distribution needs, and the best dollar value available in AC power distribution and noise suppression products.

Marway can solve your power distribution problems and save you money.

MARWAY PRODUCTS INC.

2421 S Birch St., Santa Ana, CA 92707 (714) 549-0623

THEY ARE HERE!!!

CONQUEST

A strategy simulation based on the famous board game of world conquest. Cass. \$14.95

MATCHBOOK FOOTBALL W/SOUND

Slide your "matchbook" to score your touchdowns. Not your average football game. Great graphics and sound. Cass. \$9.95.

TERROR AT SELACHII BAY

Find and destroy the killer shark which has been prowling the waters. Cass. \$7.95

MONSTER'S LAIR

Loosely based on the game of dungeons and dragons. Written totally in machine language with graphics. Cass. \$14.95 Disk \$19.95

BOSS VERSION 2.1

Multi-purpose single stepper for basic programs. Excellent for the amateur or serious programmer. Cass. \$29.95 Disk \$34.95

SUPER UTILITY

Multifaceted utility with many functions too numerous to mention here. A must for the serious programmer. Write for full details on Super Utility. Disk \$49.95

All cassette programs require Min. 16K LEV. II Disk-32K

BLANK CASSETTES

\$.85 ea. 10/\$8.00 100/\$70.00
C-20 5 Screw Housings. Excellent for computer use.

MEMORY EXPANSION CHIPS

200 Nanosecond Add \$2.50 for dipshunts and instructions \$45.00 per 16K
We also have other games and utilities, as well as cassette labels, cases, etc. Write for FREE Catalog.

TERMS: Check, Money Order, or C.O.D. Add \$1.50 for shipping and handling. Ohio residents add 4 1/2% sales tax. C.O.D. orders add \$1.25. If 2 or more programs ordered, we pay shipping and handling.

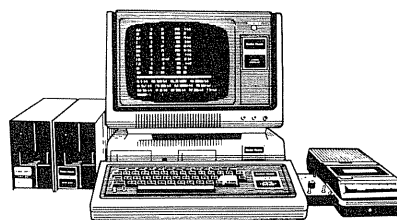
DEALER INQUIRIES WELCOME

LAKEFRONT SOFTWARE

P.O. Box 5240 Willowick, OH 44094 216-944-6722

If you're serious about the stock market, you need

Tickertec™



Watch 48 to 400 of your favorite stocks
without a 15 minute delay.

Tickertec™ is a computer program that displays the NYSE or AMEX tickertape on your TRS-80™ Model I or both exchanges as an option on the Model II. You see every trade as it is reported by the exchange and track the last ten trades, tickertape reported volume, and high and low limits on the stocks you are watching. Tickertec prices start at \$1,000.00 with many optional features available including hard copy and portfolio management systems. Programs may be purchased for cash (i.e., hard dollars) or payment can be arranged in the form of discounted brokerage commissions (i.e., Soft Dollar Software™). Exchange fees are extra. Call for FREE brochure TOLL-FREE at (800) 223-6642; in New York call (212) 687-0705; or mail the coupon today.

Max Ule & Company Inc.

6 East 43rd Street, N.Y., N.Y. 10017

Name _____
Address _____
City _____ State _____ Zip _____
Bus. Phone _____
Pers. Computer? ☐ Y ☐ N Model _____

©1980. Max Ule & Company Inc.

**Lemonade or Champagne
Nepenthe Programs
3014 Biggs Court
National City, CA 92050
\$2.95 + \$0.50 postage**

As more and more businesses are turning to the microcomputer for assistance, the demand for systems designers is drastically increasing. The task of becoming a professional systems designer may upon first inspection seem rather simple. But when one actually analyzes what this endeavor entails, well, those who are weak of spirit beware. Systems designing is for the most part a difficult but yet rewarding occupation. The booklet *Lemonade or Champagne* delves deeply into the realm of the development of microcomputer systems.

At first glance paying \$2.45 may seem a little overpriced for this seemingly scanty 52 page booklet, but this is not so. *Lemonade or Champagne* has many ideas and solutions worthy of a volume many times its size. The information in this booklet is very clear and gets right to the point. The main point, which is stressed throughout the entire booklet, is professionalism. The author feels that one must treat all potential business situations in a professional and efficient manner. The book does not attempt to teach one how to program, it assumes a moderate working knowledge of programming, although a non-programmer can easily understand most of the non-technical information.

The booklet is divided into four sections. Section one is a theory oriented chapter. It provides a structured approach to systems development and points out some of the pitfalls that might be experienced. It also deals with such aspects as problem definition, system analysis, system design, program development, unit/system testing, documentation and system implementation and maintenance.

Section 2 addresses that portion of the systems development effort which pertains to the analytical requirements. The main idea in section two is the system design manual. It tells which forms are needed to compromise an effective manual, and as a consumer service, free system development forms are available upon request from Nepenthe Business Software.

Section three looks at some of the finer points of programming the TRS-80 and in particular, the subject of random disk files. It explores several of the many alternatives open to the programmer for

the construction of random data files on disk, and shows several examples of how these alternative file structures might be employed in business applications.

Section four deals with software solutions and programming aids. One of these aids is a "global" program modifier which allows the programmer to change every occurrence of a given command or variable by changing one occurrence of the variable. The second programming and analysis aid is a line number cross-reference utility. This utility allows one to trace the logic of a program and find out just exactly where those GOTO's, GOSUB's, NEXT's, ELSE's or RETURN's go.

Other aids mentioned are program compression utilities, a file manager program and a text editor. Some of the above programs are available directly from Nepenthe Business Software. Each section of *Lemonade or Champagne* ends in a comprehensive summary and then a brief introduction to the next section.

If all of the previous information seems to be a bit much for a fifty-two page booklet, then I suggest you write and order a copy. If you do, you will find that *Lemonade or Champagne* is one of the most informative books ever to be offered on the subject of systems design for the TRS-80.

W W Harper II

**Membership Billing
Microdata
8791 S Tacoma Way
Tacoma, WA 98499**

\$269.95

It's always nice to look at good software for the TRS80. MICRODATA of Tacoma, Washington surprised us recently with a new system being introduced to run on the TRS80 Model II using the CP/M operating system. Our first look at the system is definitely favorable and we are working with our accountant to try the system out here on a full time basis.

The first package available, and the one that we have looked at for this review is the Membership Billing. Membership Billing is a complete Accounts Receivable Package which includes some really sophisticated features:

- Menu selection entry
- User-defined Customer and Service Codes
- Interactive data entry and editing
- Extensive error and improper entry checking
- Annual, Semi-annual, Quarterly, or Monthly Billing

Up to six different charges at nine different rates

- Any number of one-time charges
- User defined standard fee/service codes
- Automatic Calculation of finance charges including the ability to have exceptions.
- Automatic Account Aging
- Ability to change past due amounts
- Automatic input to the systems General Ledger

The following reports are listed for the system:

- Monthly Customer Statements, ready for mailing
- Master Customer List with aging
- Last Payment Information
- Master List of service codes and activity
- Select Customer Listings
- Daily & Monthly Transaction Registers

The system requires a TRS80 Model II or other computer with 8" drives running CP/M with CBASIC2, a minimum of 48K RAM, and a printer.

The System in Use

No amount of just testing can really tell whether a system is really any good. It's the day-to-day use of a system for a real business that pulls out the bugs in a program and highlights both the errors and the limitations of the system. The Membership Billing program has been tested on dummy files at this point, and we are sufficiently impressed to want a closer look.

In running the program, we found it very convenient and easy to use. Prompts for input were clear and sufficient for someone with a moderate knowledge to use effectively. No program such as this should be run without consulting an accountant. Even an experienced businessman could get lost in here if he didn't understand accounting. The manual doesn't try to overcome that difficulty. It provides only the simplest directions for use of the system and provides no examples or display aids. In fact, it's only 9 pages long!

The most serious fault that I found with the system was the manual provided. Clearly, this is not a system you want to try to set up without an accountant looking over your shoulder.

As far as the rest of the program goes, I have to say that the program worked well with no errors. Each advertised feature worked as specified without noticeable error and with reasonable speed. That says a lot for the system right there.

The program's displays were clean and meaningful, the reports provided useful insight into the status of the data files, and no errors were noted, even when attempts were made to force them.

The sum total of all of our efforts to date is that the system so far comes out with flying colors. In short, it works as well as can be tested for a limited amount of time. But we were impressed enough to try it for real, in our own office.

T R Dettmann

THE PROGRAMMER'S GUILD MEANS ADVENTURE

THUNDER ROAD ADVENTURE

It is late in the evening. The moon has just risen over the top of the mountain. Grandpa McKee has just loaded the last jug of White Lightning into the car. Everything is ready. Then as if on cue, the RIDGERUNNER appears. He is cool, confident, as he slides behind the wheel of the fuel injected '57 Chevy. He cranks the engine and it leaps into life with a throaty roar, 427 cubic inches of raw horsepower. Grandpa McKee hands him the destination slip. It says, simply KNAWBONE. A thin smile comes over the RIDGERUNNER'S face. KNAWBONE is Sheriff Bubba Clemmons' territory. There would be a lot of action tonight. He flips the Hurst shifter into low gear and powers away in a hail of dust and a shower of stones. Grandpa McKee shakes his head. The boy will be in for a rough time tonight, of all the routes in these hills this one is the toughest. That's why they call it --- **THUNDER ROAD!**

THUNDER ROAD ADVENTURE is so real you almost expect to see Burt Reynolds cruise by in a black Firebird. The RIDGERUNNER is your eyes and ears as you maneuver the country roads around Hazzard county. There are Revenooors, hungry bears, a good looking lady hitchiker, treacherous roads, burned out bridges, roadblocks and puzzles to solve before you can deliver your cargo of pure Kentucky Corn to the thirsty old Boys in KNAWBONE. Can you dodge the long arm of the law, and all the pitfalls of Hazzard County? Or will you wind up in Sheriff Bubba's pokey? **THUNDER ROAD ADVENTURE** is classic adventure style for ADVENTURE fans of all ages.

Tape \$14.95 Disk \$19.95

DEATH DREADNAUGHT

This mind chilling thriller will freeze the blood in your veins. You are trapped aboard an ancient alien battlecruiser whose entire crew was murdered brutally. You are the scout team's only survivor. Your ship's been smashed, but one alien shuttlecraft remains intact. As you move about the wreck the danger increases. You must secure air, food, fuel and batteries. Now, as precious seconds of your life tick away, you suddenly realize **YOU ARE NOT ALONE!** The creature that ripped this ship apart is **ALIVE** and waiting for you to make a mistake. **CAN YOU ESCAPE?** Or will **YOU** be the creature's next victim?

WARNING: DEATH DREADNAUGHT IS RATED R due to **EXTREME** descriptions of **VIOLENCE!**

\$14.95 on tape \$19.95 on disk



SPIDER MOUNTAIN ADVENTURE

Beyond the desolate outback of GRAYLOCKLAND lies the lair of the spider SHELOB. Once a wizard's cavernous fortress. SPIDER MOUNTAIN is now a deadly trap from which only the most skilled adventurers emerge. Treasure awaits the fortunate. **DEATH** stalks the foolish.

Can **YOU** solve the riddle of the SPIDER MOUNTAIN ADVENTURE and come back alive?

\$14.95 on tape \$19.95 on disk



LOST DUTCHMAN'S GOLD*

\$14.95 on tape
\$19.95 on disk

Deep in Arizona's Superstition mountains under the watchful eye of Apache Indians, the fabled **LOST DUTCHMAN'S GOLD MINE** awaits the adventurer bold enough to challenge the fates. With only a mule and a Winchester, and the ghost of an old gold prospector named **BACKPACK SAM** to guide you, do you dare to venture into the misty mountains, where it is said that the Devil himself, protects the **LOST DUTCHMAN'S GOLD?**

TRIPLE ADVENTURE DISKETTE
\$35.00—Save \$10.00

SPECIAL

DEALER INQUIRIES INVITED

DOUBLE TAPE
(ADVENTURES OF YOUR CHOICE)
\$24.95

The Programmers' Guild

P.O. BOX 66, PETERBOROUGH, NH 03458 (603) 924-6065

SEND CHECK OR M/O (SHIPMENT WITHIN FIVE DAYS OF RECEIPT) C.O.D. ACCEPTED

Files & Foibles

T R Dettmann
Associate Editor

Most programs of any real significance are concerned with storing information for later use. Business programs store accounting information. Engineering programs often store data. Even Adventure programs sometimes store information needed about the scenario.

The way we store information is in data files. These are collections of information on some storage medium such as disk or tape. We are going to take a look at simple disk files and find out how to set them up as well as some hints on how to use them.

This article is not all-inclusive. You should be familiar with the material in your DOS manual, or at least have it handy to check on things we say and do. Make sure you read about how each Basic statement is written for file handling. If you take the time, files can be a useful addition to your programming skills.

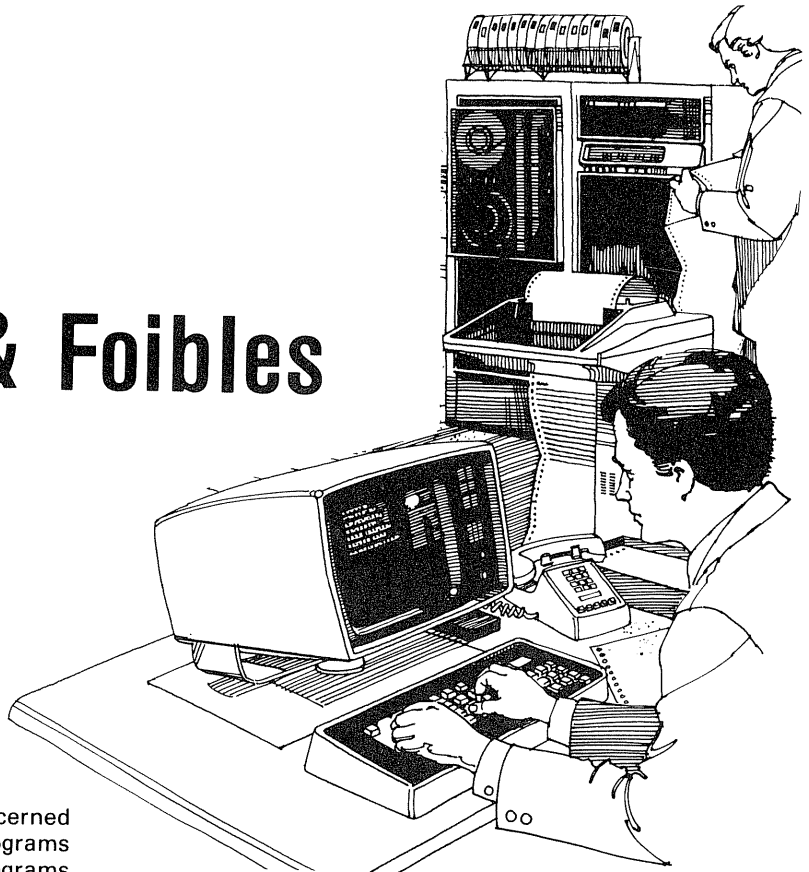
We will not be covering Random Access files here. We will cover them in a later article. This article will serve as a foundation for that later work.

What is a File?

We all have some idea about what a file is, but let's be a little more precise so that we are all talking about the same things.

In the early days of computing, everything was done on punched cards which were stored in large filing cabinets designed specifically for them. (In fact, I still have one in my office, right behind my desk!). Gradually, the usage of the word "file" transferred from the cabinet to the decks themselves.

With the introduction of tapes, it became natural to refer to information on tape as a file. Most often, the information was stored as a sequence of "card images" meaning that the tape record was built exactly like a card deck.



Both of these methods of storage had one major quality in common, they were both "sequential" storage methods. That is, both cards and tapes were read one "card" (or data record) at a time. In order to look at the 1234th record, it was necessary to first read the 1233 records before it. Woe to the programmer who dropped a card box and lost the order!

With the introduction of "disk" storage, access to files could be done much faster than before. At first, files were still stored sequentially. But, the unique feature of the disk which makes it a really powerful tool is that another, better type of file can be designed.

Disk storage has the ability to rapidly get at *any* record on the disk with only a little head movement and possibly a small wait for the disk to come around to the proper point. This means that if you know the exact location on the disk, you can go directly to an item without looking at everything before it in the file. This is Random Access.

On the first computers (and again on the first microcomputers), it was the responsibility of the programmer to keep track of the locations of his files. This required a great deal of programming (as well as a lot of worrying and pulling of hair) to make sure that files were kept separate and that the location of each bit of information was known.

It wasn't long before someone recognized that the computer could just as well (probably better) do the work. This gave rise to the concept of the Operating System, or Disk Operating System (DOS). The operating system takes care of the where and the record keeping, we only need to worry about the files once again.

Many people use sequential access data files because they are simple to use and do not require you to learn many new techniques; you just have to read and write from a new place.

Before we start talking about sequential files though, we have to talk first about file "buffers".

File Buffers

What is a file buffer? Why do we need them? How are they made? What do they look like?

A file buffer is a region of memory that is set aside to receive information from your program until an efficient sized packet is made for writing to an output file. Vice-versa, a buffer can be used to hold information from an input file for your program to access.

Why have a buffer? The problem of buffers came up years ago to speed up access to information and make it more efficient from both the program side and the external storage side. The buffer exists to match the radically different access speeds of an input/output device to the computer itself.

It would be very slow indeed if every time we printed a string, each letter had to be transferred to disk one at a time! To make the most efficient use of the disk, we should have much information ready to go to the disk all at once. To make programming clear however, we generally want to get only certain small bits of information at a time.

The buffer matches these two radically different requirements as efficiently as possible. To do this, an area of memory is set aside to be an image of a section of the disk. For disk read and writes with the TRSDOS operating system, this is 256 bytes (plus some system space for file information).

Every time you write information to disk by using a PRINT statement, you are actually putting the information in the buffer assigned to that file. Only when the buffer is full will the information be written to its appropriate place on the disk.

To see this in action, try the experiment in program 1. It inputs lines you type in at the keyboard and writes them to disk. By waiting for the disk to stop between each line, you will find that if you keep your lines short, the disk will not start of every line, even though each line is PRINTed to disk when it is read in.

Further, after 10 lines are in, the system will read in each of the lines one at a time. If you wait for the disk to stop between each read and each line doesn't fill a buffer, then the disk will not start for every line you ask to be displayed!

But where do the buffers come from? Would you believe that something *you* do creates them, even though you may not know it! In TRSDOS, when you answer the "HOW MANY FILES?" question with ENTER, you are creating the space necessary in memory for three file buffers, *even if you do not use them!* In NEWDOS, entering Basic without specifying any files portion of the Basic command does the same thing. In Model II TRSDOS, entering Basic without specifying a number of files creates *no* file buffers at all.

You can specify any number of up to 15 file buffers for use by your program when you go into Basic. To create 8 buffers, you would answer "8" to "HOW MANY FILES?" in TRSDOS, type "BASIC 8" to enter Basic in NEWDOS, or "BASIC -F:8" in Model II TRSDOS to enter Basic.

PROSOFT

Dept B / Box 839 / No. Hollywood, Ca. 91603 (213) 764-3131

Your TRS-80[®] Model 1 and Line Printer IV or Centronics[®] 737 can easily produce documents with this typeset look. All you need is

PROP.

Notice how the letters (not just the words) on each line have been evenly spaced, resulting in a professional, rather than a computerized appearance. (You are reading PROP-created, 737-printed text right now.)

<*> <*>

If you have been looking for an IBM[®]-like EDITOR and WORD PROCESSING text formatter, then you have been looking for

SUBEDIT and SUBSCRIPT

both based on the highly-regarded components of CMS ... software normally found on mainframe time-sharing systems ... and now available for your TRS-80 Model 1. SUBSCRIPT supports:

readable control words
737 printer output
appended files
form letters
indentation
PROP
centering
disk files
embedded files
right-justification
top and bottom titles
video output (checkout)

(and, of course, more)

<*> <*>

All software is distributed on diskette for 32K and 48K TRS-80 Model 1's. Documentation and sample programs are included. At least one disk drive is needed.

PROP \$22.95

SUBEDIT + SUBSCRIPT \$22.95

SPECIAL: Both packages \$39.95 postpaid

Calif. residents please add 6% sales tax
Payment by check or money order

Dept B / Box 839 / No. Hollywood, Ca. 91603 (213) 764-3131

35→40

TRACKS IN 15 SECONDS

That's how long it takes to open up those unused five tracks on your old 35-track diskettes with

XTEND40

If your TRS-80[®] operating system and disk drives support 40 tracks, but you still have some 35-track diskettes around, this is the quickest and easiest way to expand their capacity.

XTEND40 is distributed on diskette for TRS-80 Model 1's with NEWDOS40 or NEWDOS/80[®].

INTRODUCTORY PRICE: \$14.95 postpaid

Calif. residents please add 6% sales tax
Payment by check or money order

WRITE OR CALL FOR INFORMATION
ON OUR CURRENT SOFTWARE LINE

In order to use any file, sequential or random, the file must first be "OPENed". The open statement is provided to do this in Basic. By opening a file, we take one of the available file buffers and use it as a connector between the file and your program. The open statement looks like this:

OPEN "mode", filename, "filename"

Where "mode" is "I" if the file is sequential and being used for input, or "O" if the file is sequential and being used for output. ("R" if the file is random, but we will cover that later). Filename is the number of the buffer to assign to the file, and "filename" is the name of the file stored on disk.

When a file is opened, the file named in the open statement is connected to the buffer and file information is funneled through there. When the file is used, either to put information into the file or to get it out, that information is first put into the buffer and then picked up by the program in INPUT statements or written to the disk when the buffer is full.

When you are done with the file, you use the CLOSE statement to flush all information out of the file buffer and onto the disk. If you do not close a file properly, some information may still be in memory, including updating information for the directory if the file was increased in length.

Sequential access files work just like files on tape, except that a few extra commands are needed to use them. We first OPEN the file for use, then INPUT or PRINT information from or to the file as desired.

To create a sequential file, we first OPEN the file for output. Let's say we want to create a file called "MAIL/SEQ". If we assign it to buffer 1, then we use the statement:

OPEN "I", 1, "MAIL/SEQ"

This makes the file available to our program. To write to the file, we simply use the standard PRINT statement, modified to indicate which file we want:

PRINT #1, NM\$, ADR\$, CTY\$, ST\$, ZP\$

Easy right? Well, we goofed (did you notice?). A PRINT statement like the one above writes each item in the print list to the file the same as if it were going to the printer or the screen.

No problem you say? But there is when we try to input the items later. Look at Figure 1. Here is how the record looks for the strings given. Each of the strings we printed to the file will be separated from the others by spaces. *But when we input the strings later, Basic will think this is only one string!*

In an INPUT statement, Basic will add to a string until it finds a comma or the end of the record. But there are no commas here! To get around this, when using strings, we have to put the commas in the file like this:

PRINT #1, NM\$+"," , ADR\$+"," , CTY\$+"," , ST\$+"," , ZP\$

Notice that there are two ways to put the comma in, either add it to the string that needs it, or put it in separately.

With numbers, we don't need the commas since a number is automatically stopped when a blank space is reached. But a number after a string still must be separated by a comma, like this:

PRINT #1, NM\$+"," , AGE

Without the comma, the number that should be AGE

would be read in by an INPUT statement with string NM\$. That would cause an error when the system tried to read AGE since there is nothing left in the record.

After a file is created, it must be CLOSED to get all the information out of memory and to the disk. Then we can OPEN it for input.

To OPEN a file for INPUT with sequential access, we use the statement:

OPEN "I", 1, "MAIL/SEQ"

To bring in a record of information, we use the statement:

INPUT #1, NM\$, ADR\$, CTY\$, ST\$, ZP\$

This will bring in one record from the file and put the information in the strings in the input list (remember, this will only happen correctly if the commas are in the record on disk).

It often happens that we wish to bring in things from disk with commas. For example, in writing a text editor, we could make each line a string variable. But then there might be commas in the lines. To get everything in a record as one string variable, we use the command:

LINEINPUT #1, LN\$

Everything in the record will be put in the string LN\$. You can try this easier than you think by saving a program on disk in ASCII format (put an "A" after the last quotation mark when you save it). Program 2 illustrates how to do this with program files. This works because programs are stored as sequential files on disk.

On input from a file, we have a special problem. Where is the end of the file? If we read in more records than are in the file, our program dies with an error from Basic. To prevent that, we could put the number of records to be found in the first record of the file, but that gets a bit cumbersome.

Basic provides us with a simpler way to find the end of a file. We simply check for the end of file with the EOF function. You can imagine this as a true-false function. It is false until we reach the end of the file, then it is true. To use it, we check it just *before* we are about to input a record. If the file is at the end of file, EOF is true. In that case, we skip to some other processing, otherwise we can read the record. Program 2 illustrates this use of the EOF function. Once we are done with a file for input, we close it, as we did the output file.

Because of the way these files are put together, there are some fundamental problems in using them efficiently. For example, unless you are using NEWDOS, you cannot open a file for output at the end of the file.

Whenever you open a file, a pointer is created which points to the first record in the file. If we opened the file for INPUT, this is fine, since we will read the records in order until we get to what we want.

If the file is opened for OUTPUT though, the only way to move the pointer is by PRINTing new records to the file. By doing this, we are writing over the old records. In order to add to the end of a file, we either have to bring the whole thing into memory and then write it back to the file, or we have to copy it record for record to a new file which we keep open for output. Either way, it is inconvenient.

With random access files we will see no such problem. We can read or write to or from any part of the file. But to get that ability, we have to add in some additional complexities. This will be the subject for a future article on file handling.

With these strings:

NM\$="80US JOURNAL"

ADR\$="3838 S. WARNER"

CTY\$="TACOMA"

ST\$="WASH"

ZP\$="98409"

FIGURE 1

the file will look like this:

80US JOURNAL 3838 S. WARNER TACOMA WASH 98409

```
10 REM*****
20 REM
30 REM   DEMO PROGRAM 1
40 REM   SEQUENTIAL FILES
50 REM
60 REM   TERRY R. DETTMANN
70 REM                                     FILENAME: PROG1/BAS
80 REM*****
90 CLS: CLEAR5000
100 SC=63:GR=140:MD=20
110 REM  ON TRS80 MODEL II USE THESE INSTEAD
120 REMSC=79:GR=95:MD=35
130 REM - - - - -
140 PRINTSTRING$(SC,GR):PRINTTAB(MD)"FILE BUFFER DEMO"
150 PRINTSTRING$(SC,GR)
160 PRINT:PRINT
170 PRINTTAB(10)"OPENING FILE TEMP/DAT"
180 OPEN"O",1,"TEMP/DAT"
190 PRINTTAB(10)"NOW INPUT 10 LINES"
200 PRINTTAB(10)"WAIT UNTIL THE DISK STOPS BEFORE ENTERING A NE
    W LINE"
210 PRINT
220 FORI=1TO10
230   PRINTI;" ":";:LINEINPUTLN$
240   PRINT#1,LN$
250 NEXTI
260 PRINTTAB(10)"CLOSING FILE"
270 CLOSE
280 REM - - - - -
290 PRINTTAB(10)"OPENING FILE FOR INPUT"
300 OPEN"I",1,"TEMP/DAT"
310 CLS:PRINTSTRING$(SC,GR)
320 PRINTTAB(MD)"BUFFER DEMONSTRATION":PRINTSTRING$(SC,GR)
330 PRINT:PRINT
340 PRINTTAB(10)"PRESS ENTER TO READ IN A LINE"
350 PRINTTAB(10)"WAIT FOR THE DISK TO STOP BETWEEN LINES"
360 PRINT
370 FORI=1TO10
380   C$=INKEY$:IFC$=""THEN380 ELSE IF ASC(C$)<>13 THEN 380
390   LINEINPUT#1,LN$
400   PRINTI;" ":";LN$
410 NEXTI
420 PRINTTAB(10)"CLOSING FILE"
430 CLOSE
440 END
```

```

10 REM*****
20 REM
30 REM   PROGRAM 2
40 REM   SEQUENTIAL FILES DEMONSTRATION
50 REM
60 REM   TERRY R. DETTMANN
70 REM                                     FILENAME: PROG2/BAS
80 REM*****
90 CLEAR5000:CLS
100 SC=63:GR=140:MD=20
110 REM   ON TRS80 MODEL II USE
120 REMSC=79:GR=95:MD=35
130 REM   - - - - -
140 PRINTSTRING$(63,140):PRINTTAB(20)"READ AND DISPLAY FILES"
150 PRINTSTRING$(63,140)
160 PRINT:PRINT
170 PRINTTAB(5)"ENTER THE NAME OF AN ASCII FILE (EITHER A"
180 PRINTTAB(5)"PROGRAM FILE SAVED WITH THE 'A' OPTION OR A"
190 PRINTTAB(5)"FILE WRITTEN TO DISK BY ANOTHER PROGRAM"
200 PRINT
210 PRINTTAB(10);:LINEINPUT"FILENAME: ";FF$
220 OPEN"I",1,FF$
230 I=0
240 IF EOF(1) THEN 290
250   I=I+1
260   LINE INPUT#1,LN$
270   PRINTI;" : ";LN$
280 GOTO240
290 CLOSE

```

WordMagic II™

**DOLLAR FOR DOLLAR, FEATURE FOR FEATURE
THE MOST COST-EFFECTIVE, PRACTICAL, EASY TO USE
WORD PROCESSOR FOR THE TRS-80™ MODEL III****

Field Tested & Proven - WordMagic II™ includes:

Total TRSDOS™ compatibility - no need to pay extra for a separate operating system!
Automatic WRAP-AROUND • Automatic MARGINATION • TABBING • PRINT FORMATTING •
Automatic PAGINATION • VERTICAL MARGINS • PAGE NUMBERING • LINE NUMBERING •
INDENTATION • SPLITTING LINES • REJOINING LINES • UNDERLINING (Most letter quality
printers) • PAGE BREAKS • MEMORY BUFFER "SCRATCH-PAD" • VARIABLE LENGTH FORMS •
Full Edit Commands - GLOBAL SUBSTITUTION • INSERTION • FIND • OVERWRITE • KILL LINES
• HACK • DELETE CHARACTER(S) • FULL CURSOR CONTROL - UP, DOWN, LEFT, RIGHT, TAB,
END OF LINE, BEGINNING OF LINE • Automatic Generation of a Table of Contents • Smooth Right •
Centering • Left Justify • Merge Data Blocks • Create "Personalized" Form Letters • Merge Repetitive
Paragraphs • Mail List • Label Generation included at no extra charge! • Documents up to Diskette
Size • Easy to use Command Structure • Complete User's Manual

**Requires 64K (WordMagic II may be used with one disk drive, but its installation requires two drives).

COMPLETE PACKAGE (DISKETTE & MANUAL)

\$195.00

(California residents please add 6% tax)

(Overseas purchasers please add \$15.00)

CalData Systems

P.O. Box 178446
 San Diego, CA 92117
 (714) 272-2661



TRS-80™ & TRSDOS™ are Trademarks of Radio Shack, a division of the Tandy Corporation.

The Model III makes its Debut

Our 32K, two drive Model III finally arrived just before we put this issue to bed. Even though time is a precious commodity near press time, we felt you ought to hear about our first impressions of this machine.

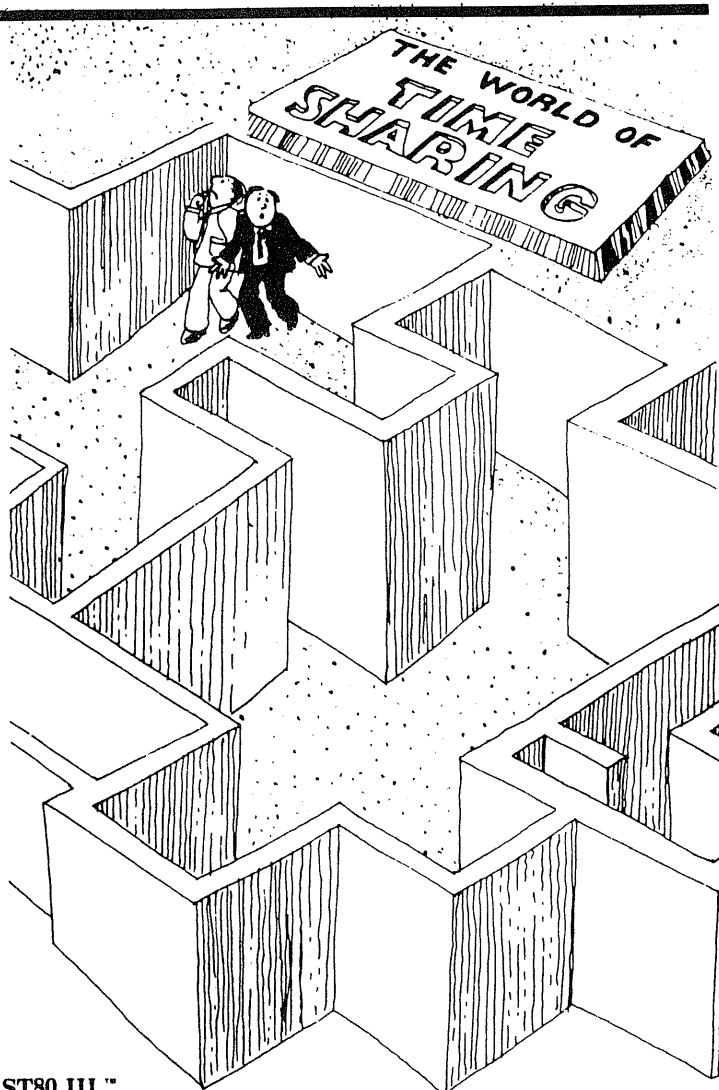
There is just one way to describe the Model III - it's cute. Compared to the Model I, the Model III is just one nice compact package. It has one power cord and no extra boxes sitting around with flaky cables connecting them. It has considerably more disk storage than the Model I, is faster, and purrs like a well oiled electric coffee grinder when the drives are active. There is no more "bang, bang" when the head loads. Cute and powerful are the two words that first come to mind after you first make eyeball contact with it.

The TRSDOS manual which came with the machine is still in the preliminary stages, but in spite of that, is very complete. But who reads manuals? The first shot out of the barrel was to see if one of the cassette recorder cables from the Model I would fit. No, they don't. The connector is almost the same, but not quite.

Remembering some of the features we wrote up in the last issue, we tried the LIB command from DOS READY. Sure enough, there they are. The HELP command is especially useful, although not nearly as complete and detailed as those in the Model II TRSDOS 2.0. The HELP command keeps you from running for the manual every time you try something different. If you don't understand the notation used, there is even a HELP SYNTAX, which explains it for you.

After playing around with the library commands for a time, it became apparent we would have to crack the book after all. The first thing was to get some Model I programs loaded and running. Enter the CONVERT command; here is how it worked:

Insert your Model I diskette in drive 1, and from DOS READY, enter CONVERT. It asks for the source drive (in this case 1), and the destination drive (0). It then reads the files on your Model I diskette and converts them to the Model III format on drive 0. Yes, there is plenty of space on drive 0. It's a 40 track, double density drive, remember? After the conversion is complete, you remove the Model I diskette from drive 1, insert a blank diskette and format it. Then you can copy the converted files from drive 0 to 1. After this, you can use PURGE to



ST80 III™

The Ultimate Communications Utility

The *Smart Terminal Communication Package* from SBSG, Inc., can turn your TRS-80* Model I or Model II Microcomputer into a very intelligent distribution processor. Easy to use commands and a built-in HELP function insure successful operation even by the most inexperienced personnel. Full user control of all communication options insure that whatever your communication requirements, ST80 III™ can provide for them. We'll get you there.

ST80 III™ can test your communication hardware and notify you of hardware fault. ST80 III™ can transfer files from memory to other computers and process received information or store it on disk. ST80 III™ can support prompted or unattended modes of operation, or remote control from a host computer. ST80 III™ can take full printer control. User definable *control tables* can be used to establish special control functions. User definable *function keys* can also be used.

SBSG, Inc., provides full user support and markets three other ST80 products. Any computer with communication capability can be accessed by ST80 III™ via your TRS-80*

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

**SMALL BUSINESS
SYSTEMS GROUP**
6 Carlisle Road
Westford, MA 01886
(617) 692-3800



FORTH is an advanced language and system for advanced programmers. MMSFORTH is a professional version tailored to the Radio Shack TRS-80 Model I and supported by MMS.

Interpreter, Compiler, Assembler, Full-Screen Editor Structured & Modular Programming, Expandable Instruction Set w/Source Code, Graphics, Strings, Arrays, Double-Precision, Very Fast and Compact, Virtual Memory includes 5 Demo Programs w/Source Code

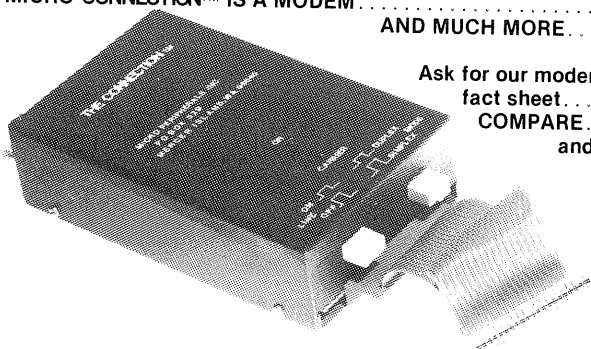
MMSFORTH System Cassette (16K, 1 drive) \$64.95
 MMSFORTH System Cassette (16K, Level II) \$44.95
 microFORTH PRIMER (required for above) \$15.00
 THE DATAHANDLER in MMSFORTH (32K, 1 drive) — an exceptionally fast, flexible and easy to use Data Base Management System \$49.95
 P I M S MANUAL (required for DATAHANDLER) \$ 9.95
 FLOATING POINT MATH and FULL Z80 ASSEMBLER \$29.95
 Diskette (16K, 1 drive) \$25.00
 Other FORTH literature: \$19.95
 USING FORTH — beyond microFORTH PRIMER \$ 6.95
 URTH TUTORIAL MANUAL — very readable introduction
 CALTECH FORTH MANUAL — good on FORTH internal structure
 Shipping is \$2 plus \$1 per extra book, Mass. orders add 5% tax.

mmsFORTH
 MILLER MICROCOMPUTER SERVICES
 61 Lake Shore Road, Natick MA 01760 (617)-653-6136
 Dept. U181
 Send SASE for free information

SAVE 50% AND

GET MICRO CONNECTED Plugs directly between ALL S80'S (Level I,4K to MOD II,64K) and the telephone network. To compliment the DIRECT CONNECT MODEM, a separate RS-232 jack ALSO operates your serial printer on OR off line. With THE MICRO CONNECTION™ you have accurate 300 baud two-way and simplex communications between you and the computer data base of your choice. In ADDITION, you have radio communications and extensive software options and packages. **NO NEED** for the expansion interface, acoustical coupling modem, RS-232 board or cabling. That is why **THE MICRO CONNECTION™ IS A MODEM**.....

AND MUCH MORE....!



Ask for our modem fact sheet....
 COMPARE...
 and.

Then get your MICRO CONNECTION™ for only \$249⁰⁰!

the microperipheral corporation

PO Box 529 Mercer Island WA 98040

206 454 3303



kill off the files on drive 0 to make space for another batch. There is only one hitch here: Convert will copy *all* the files on the Model I diskette - you have no choice. So what we did was format a clean diskette on Model I and copy only those files from the Model I library that we wanted on the Model III.

The converted programs ran well on the Model III. In some cases we had to change the CLEAR statement downward to account for the 16K less memory than our 48K Model I. We stayed away from programs heavily loaded with PEEKS and POKES, but would you believe that SUPERZAP 3.0 converted over and ran! There are a whole lot of questions about it accessing past track 34, but we'll soon find out all about that.

There are so many NEWDOS-like features in TRSDOS for the Model III that it makes you feel right at home - almost. The syntax is just a little different in some cases.

To get a directory from Basic, you can do a CMD"D:d" (where d is the drive), and come back with a directory and a Basic READY prompt. CMD"E" will give you the previous TRSDOS error, just as before. CMD"L" will load a Z-80 subroutine or program file into RAM. CMD"P" will check for printer status. CMD"J" converts a calendar date (both ways, from 00/00/00 to what day of the year and vice versa). CMD"Z" will duplicate the video output on the printer (this is in addition to the DUAL command, but is easier to use inside a Basic program). CMD"C" gives you Basic program compression and deletion of remark statements. (This one is nice!). Another real neat one is CMD"O", which will alphabetize (sort) a string array. We didn't put this one to a real test, only the sample in the manual, but it has some real potential. CMD"X" will give you a cross-reference of reserved words, string variables or strings in a Basic program. Very nice indeed.

Model III Disk Basic now has the up-arrow to list the previous program line, down-arrow for the next line, period to list the current line, comma to edit the current line, shift-up arrow to list the first program line and shift-down arrow to list the last program line. You can also do an "L10" to list program line 10. "Exx" or "Dxx" will edit or delete program line xx, and Axxx,xxxx will give you AUTO beginning at line xxx, incrementing by xxxx.

CMD"I" will let you execute DOS commands from Basic, and return you to Basic. There is a "wild-card" feature which can be used with COPY and KILL. For example, COPY/BAS :0 :1 will copy all files with extensions of /BAS from drive 0 to drive 1.

There is a Basic command to disable/enable the BREAK key, it is CMD"B" (ON or OFF). There is also a Basic program renumbering command, which allows you to specify the newline, startline and increment.

Disk Basic features variable length records, by specifying the length, and the letter "V" after the HOW MANY FILES? question. You can also append to the end of a sequential file by using the OPEN "E" statement.

There is no doubt about it, the Model III is a serious contender, and it looks like the Biggest Name in Little Computers has done it again.

TWO computers with ONE printer

Gary Rittenbach
Auburn, WA

You don't have to switch 40 lines of ribbon cable - two will do it.

I teach computer science at a private school. We had two computers and only one printer. The problem was to be able to reach the printer from either computer. Here is the equipment we had:

One 16K Level II

One 32K Level II with expansion interface and disk drive.

One line printer II.

A 40-line buffered cable ran from the CPU to the printer. The expansion interface has its own 34-line buffered printer port, although it still uses a 40-line ribbon cable. In order to hook up both computers, I made a switchbox (See Figure 1). On the 16K machine, I found line 35, a power line, when disconnected, severs all communication to the printer. This one line virtually clips all 40 lines. The 32K machine reacts in the same way when line 1 on the cable is disconnected.

From the CPU all 40 lines go directly to a rotary switch in the switch box. The expansion interface on CPU 2 has all lines going directly to the printer through the switch box except line 1 which is connected to the same rotary switch. When the switch is in the 1 position, (see Figure 1), line 1 from the expansion interface is disconnected. Thus, line 35 from CPU 1 is connected, allowing CPU 1 to use the printer but not CPU 2.

When the switch is in the 2 position, line 35 from CPU 1 is disconnected and line 1 from the expansion interface is connected, allowing CPU 2 to use the printer but not CPU 1.

Using the manuals provided with the printer, interface or computer, you must determine which positions on the edgecard, and thus which lines on the ribbon cable are #1, #2, etc. Wire wrap all the number 1's together, the number 2's together, and so on. **You only switch as many lines as you have computers.** In this case two lines, the #35 line coming from the buffered cable and the #1 line from the buffered port on the expansion interface. If you had two 16K machines and no expansion interface, you would need two buffered printer cables and then you would hook line #35 from both cables to the switch.

We already had the buffered cable, so our costs were:

\$10 for ribbon cable

\$12 for three sockets

\$15 for three 90 degree wire wrap headers.

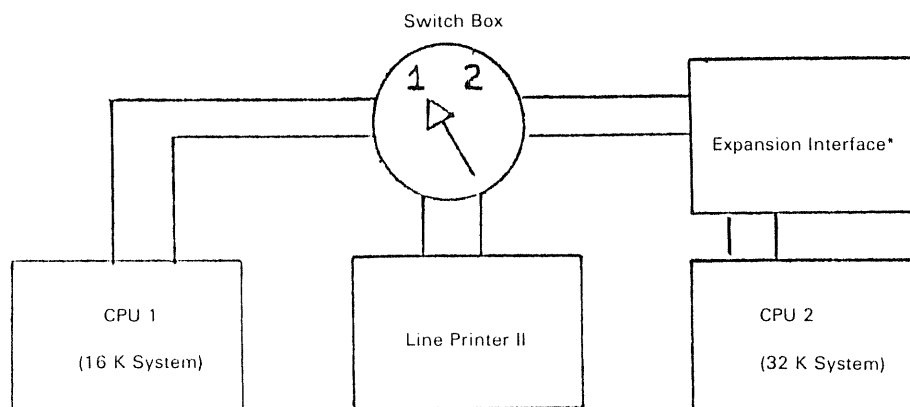
\$12 for three 40 pin edgecard connectors.

\$1 for perf board.

\$2 for a rotary switch.

\$5 for a metal box.

Our students have used this \$60 hook-up for several months of the school year. It definitely is student-proof. ●



*The expansion interface is unnecessary. Both systems could be 16K. Then line 35 from both computers will be connected to the switch.

Figure 1

TRSDOS version 2.0 for the Model II has been released. This operating system is a significant improvement over the previous 1.2 version. Here, courtesy of Bill Schroeder, are some patches for TRSDOS 2.0:

```
PATCH SYSRES/SYS A=0CFB F=1E C=1D
PATCH SYSRES/SYS A=0D12 F=1E C=1D
PATCH SYSTEM/SYS R=24 B=199 F=2020 C=4853
```

2. This patch is to allow the DEBUG in TRSDOS 2.0 for the Model II to function from 0000H to FFFFH so you can work with the system. As written the DEBUG will not allow you to view, modify or monitor addresses below 2800H or above F400H.

3. This patch is to make the "VERIFY DETECT" function work properly on early releases of the 2.0 DOS for Model II:

As written, the system is merely going through the motions of providing disk change detection, but not actually doing anything about it (except taking up time on every disk I/O operation). This patch will make the system almost "user proof" (in regard to not having to perform an "I" after each disk change). However, for every plus there is a minus, as the trade-off for this feature is speed.

The "VERIFY DETECT" function causes the operating system to slow down by 3 to 5 times. The cure for this is to simply TURN IT OFF. To do this, type "VERIFY DETECT OFF" at TRSDOS READY. If you turn the function off, it is completely off and the operator *must* do an "I" for INIT after every disk change. Failure to do the "I" will most certainly result in lost data. Be careful, and have backups of every disk!

You have to separate the string variables that won't change (data base, etc.) and those that will change. Load the strings that won't change first, usually from disk or tape. Here is where the sneaky stuff comes in. The TRS-80 uses three pointers for string allocation, the beginning or 16561 & 16562, the end or 16616 & 16617 and the next available string byte or 16598 & 16599. PEEK locations 16598 & 16599 and POKE the results into 16561 & 16562 respectively. You have now protected that string area which is above that point. When the "garbage man" comes, he will only repack the unprotected part of the string area. If a sizable part of the area has been protected, you should notice the difference in speed. One final note, if you stop your program and try to rerun it, you may get an "out of memory" error. To relieve this condition automatically, POKE 16561 & 16562 with 255, or whatever you wish to unprotect, before the normal exit from the program.

```
100 DEFINT I-J
110 FOR I=1 TO 5: PRINT "I="; I
120 IF I=3 THEN 200
130 NEXT I: STOP
200 FOR J=1 TO 5: PRINT "J="; J
210 FOR I=1 TO 5: NEXT I
220 NEXT J: STOP
```

118 80-U.S. JOURNAL Jan/Feb 1981

signal points at the innocent variable J in the 200 block.

One solution is to routinely provide a dummy FOR...NEXT loop at each jumpout. For example, replacing line 120 with:

120 IF I = 3 THEN FOR I = 1 TO 1: NEXT I: GOTO 200
will fix up the program. (Thanks to Harry McAndrew, Seattle WA for this one.)

When working with disk files (or even with tape files), it's easy to forget what name we are storing a program with. This can cause a problem, particularly on disk, since we might accidentally store a program we are changing over another of the same name - but not the one we want.

This generally happens when the names are close to each other. In order to overcome this problem, a good idea is to put the program name in a remark statement at the beginning of the program in a standard line. Then all you need to do to make sure you have the right filename is to look at that line or just list the beginning of the program.

Along the same lines, it's a good idea to make it a practice to always include a remark label at the beginning of a program that indicates when the program was last modified, who wrote it, what it is called (including the filename) and any other information. This can help at a later time in reviewing your programs.

Do you need to have variable field formatting in your program? Have you noticed that the string describing the field layout for a PRINT USING statement can be a variable? This means that you can change the formatting in response to changing needs or keyboard input simply by building a string that contains the proper formatting information.

A note to SCRIPSIT users and other text editor users. Watch out when you have a large text in memory. As you get more and more text in, the system will slow down. This is particularly apparent when you put in a word that is shifted to a new line during editing.

What happens is that it takes longer and longer to shift the word to the new line during editing. If you are typing at all fast (even two finger fast!) you may lose some letters.

Don't forget the calculator model! Remember that Basic can execute most instructions directly from the keyboard. This gives you a powerful tool for debugging or testing programs.

When you have built a subroutine or a short program segment, type in the values of dummy variables you wish to use, then GOTO the program segment or GOSUB to the subroutine, directly from the keyboard. This will execute the section you want and then return to the keyboard.

You can then print the values of variables you want to check to see if everything went right. This is one of the most powerful features of the interpreter system for Basic, and one you can use daily. ●

MODEL I to MODEL II

Radio Shack* provides a means to transfer programs. Now Unique Software provides a means to **CONVERT** them. PRINT@, CMD, some PEEK, even SET and RESET, and more are rewritten for execution on a Model II. What **CONVERT** cannot rewrite, will be listed by type of error and line number. **CONVERT** will execute under Model I or Model II Disk Basic with no changes. Useful 'Summary of Conversion' included.

(Available on formatted disk)

on Model I diskette (upload yourself). . . . \$16.95

on Model II disk. . . . \$19.95

Price includes postage and handling Texas residents add 5% sales tax



Unique Software
700 Executive Plaza Building
Fort Worth, Texas 76102

If you're looking for unique software look for U.S.

* Radio Shack is a division of Tandy Corp. and not affiliated with Unique Software

OUTPUT 3

For the TRS-80™.

EXPAND-O-BOARD

\$29⁹⁵

As your system grows, the need for additional I/O ports becomes obvious. Solve the problem with the new three port extender from S.C.P. The EXPAND-O-BOARD may be connected to the keyboard unit or E.I. A bus cable is required and may be ordered separately for \$11.95.

TO ORDER CALL:

(313) 264-5704

OR SEND YOUR ORDER TO:

Sterling Computer Products
36811 Lodge Drive
Sterling Heights, MI. 48077

Add \$1.50 for shipping - Mich. res. add 4% sales tax

Orders may be paid by check, money order or COD.
VISA & MASTER CHARGE ACCEPTED

DEALER INQUIRIES INVITED!

*TRS-80™ is a trademark of Tandy Corporation

BREAK - with JKL

Stop Printing All Those Blanks..

Al Domuret
Fair Oaks, CA

One of the very useful features of Apparat's NEWDOS for the TRS-80 is the "JKL" option which dumps everything from the screen display to the printer. However, some of us are a bit disappointed about the lack of a functional BREAK key: Once the screen printing starts, it continues until the entire screen has been printed - even if it is mostly blank - or until the printer is turned off. What is needed is a way to stop the printer with the BREAK key. And so, here is the fix.

Two methods are provided to suit the preferences of individual users. One method puts the BREAK check routine directly to the SYS1/SYS file so that the user never knows it is there and is not required to load it. It will be loaded automatically with normal disk operations. The second method requires the BREAK check program to be loaded manually whenever the user wants to use it. My preference is the first one, and that is the one which will be discussed first.

For the first method which requires direct-to-disk software changes, it is assumed that the user has possession and a working knowledge of that handy tool called SUPERZAP. As usual, the user is cautioned to first perform the changes on a backup disk. If you are not familiar with using SUPERZAP, this is a good project with which to practice because the change is rather short and simple.

Before proceeding with the actual SUPERZAP changes to NEWDOS, perhaps some readers would appreciate some explanation of what is being accomplished and how.

The "JKL" screen printer code can be found at memory location 43B1 to 43E2 Hex, and can be examined with DEBUG, SUPERZAP, RSM2D, or whatever. The routine is quite short, and the problem is to somehow fit a BREAK KEY check routine somewhere in the middle of the code. I arbitrarily inserted a jump out of the "JKL" routine at 43CA Hex, the jump going somewhere to a BREAK check and RETURN routine.

Problem: Where to put the BREAK check routine so that it will load automatically from disk? It should not require a memory size protect in high memory and it must not get clobbered at wherever it is to be situated. In other words, the user should not be bothered with having to load it and should not even be aware of its presence.

There are a few small memory "holes" in low RAM at 4060 and 40C0 Hex which would serve our purpose, but I am already using these holes for my Heath printer software driver and other similar short machine language

routines. In hunting another safe haven for the BREAK check routine, I learned that:

1. The SYS1/SYS file resides almost continuously at 4E00 to 51C8 Hex.

2. At times SYS1/SYS gets overwritten by the other system files, but the memory area from 51C9 to 51FF is pretty well left alone by these other system files.

3. Although SYS1/SYS does get overwritten by other system files at times, it gets reloaded back to its usual memory location after every disk operation.

So a logical solution appeared to be to put the BREAK check routine at 51C9 Hex and to have it reloaded everytime SYS1/SYS gets loaded in the normal course of system operations. To make a long story short, it works, and the fuss of making the software change is negligible.

Enter SUPERZAP

To make the necessary disk software changes, bring up SUPERZAP. The changes to SYS1/SYS begin at track 10, sector 4, byte 8A. Starting at 8A, you will find "02 02 00 4E etc."

Change 02 02 00 4E etc., by typing from SUPERZAP, MOD8A

The changes to be made starting at byte 8A with SUPERZAP are listed in Table 1.

The Second Method

The second method of implementing the BREAK check routine is presented as a source listing in Table 2. If you prefer this method over writing directly to your DOS disk with SUPERZAP, just program it as shown in the table. When implementing the program, simply run it from DOS as needed.

Notice that some of the Hex code listed in Table 1 is not included in the Table 2 source listing. This additional Hex code is used by the TRS-80 in manipulating disk files. As a brief explanation of what the additional disk code in Table 1 means, the first characters that read: 01 21 C9 51 are interpreted as follows:

01 = notification to DOS that a new record of code follows.

21 = this new record of code consists of 21 bytes (hex), including the "memory pointer".

C9 51 = "memory pointer": start loading the code to memory location 51C9 (note the reversal of the two bytes. This is typical).

F5 C5 = the code follows up to, but not including, the next 01 or 02 encountered.

The next record starts with the next '01'.

01 05 CA 43 C3 C9 51: 01=pointer, 05=number of bytes, memory location is 43CA, followed by code to be loaded there (code=C3 C9 51).

The last record starts with the '02'.

02 = substitute for '01' to indicate End of File.

02 = EOF consists of 2 bytes, and these next two bytes represent the file's execution address.

00 4E = execution address for SYS1/SYS (4E00 Hex).

Every SYSTEM file and every CMD or similar object code file is structured on your disks in a similar way. Using SUPERZAP, take a look at some other CMD or SYS files and try to trace the '01' pointers from one record to the next, and where each record is loaded into memory. This

can be a very useful learning experience. It will help you to determine where in memory the disk data is to be loaded. Or, working in the opposite direction, if you want to find where certain memory code is located on disk, this is how to go about finding it.

The user should be advised of one minor glitch if method one is used (i.e., writing the code directly to SYS1/SYS): If DEBUG is called up, the "JKL" screen printer option will not work. If this is not acceptable to the user, it will be necessary to use the second method discussed above which requires the BREAK check routine to be loaded manually as a program. With this second method, it is possible to get a screen print dump of the DEBUG display. ●

```
.....
01 21 C9 51 F5 C5 E5 21 7F 38 7E FE 04
28 09 E1 C1 F1 CB 74 23 C3 CD 43 3E 0D
CD 3B 00 E1 C1 F1 C3 DA 43 01 05 CA 43
C3 C9 51 02 02 00 4E
```

Table 1

Write these codes to SYS1/SYS with SUPERZAP. The changes start at track 10, sector 4, byte 8A. Start the changes by entering "MOD8A" with SUPERZAP.

```
.....
00010      ORG  4049H      ;POINTER TO TOP OF MEMORY
00020      DEFW FIXP-1    ;SET MEMORY SIZE PROTECT
00030      ORG  43CAH      ;JUMP OUT OF SYSO/SYS
00040      JP   FIXP      ;TO CHECK FOR BREAK KEY
00050      ORG  OFFEOH
00060 FIXP  PUSH AF      ;SAVE REGISTERS
00070      PUSH BC
00080      PUSH HL
00090      LD   HL,387FH  ;IS BREAK KEY PRESSED?
00100      LD   A,(HL)    ;SAVE KEYBOARD INPUT IN A
00110      CP   04H      ;04=BREAK KEY CHARACTER
00120      JR   Z,BRK     ;GO IF BREAK PRESSED
00130      POP  HL      ;BREAK KEY NOT PRESSED, SO
00140      POP  BC      ;RESTORE REGISTERS
00150      POP  AF
00160      BIT  6,H      ;THESE 2 INSTRUCTIONS WERE
00170      INC  HL      ;OVERWRITTEN IN SYSO/SYS
00180      JP   43CDH     ;RETURN TO SYSO/SYS
00190 BRK   LD   A,ODH    ;BREAK KEY IS PRESSED, SO
00200      CALL 003BH     ;OUTPUT A CARRIAGE RETURN
00210      POP  HL      ;RESTORE REGISTERS
00220      POP  BC
00230      POP  AF
00240      JP   43DAH     ;JUMP TO END JKL ROUTINE
00250      END  402DH     ;GOTO DOS AFTER LOADING
```

TABLE 2

Use EDTASM to write program. Notice that this routine will automatically protect itself by defining the "Memory Size" at memory location 4049 and 404A hex.

LEARN TRS-80® ASSEMBLY LANGUAGE DISK I/O

Your disk system and you can really step out with REMSOFT's Educational Module, **REMDISK-1**, a "short course" revealing the details of DISK I/O PROGRAMMING using assembly language.

Using the same format as our extremely popular introduction to assembly language programming, this "ASSEMBLY LANGUAGE DISK I/O PROGRAMMING" course includes:

- Two 45-minute lessons on audio cassette.
- A driver program to make your TRS-80® video monitor serve as a blackboard for the instructor.
- A display program for each lesson to provide illustration and reinforcement for what you are hearing.
- A booklet of comprehensive, fully-commented program listings illustrating sequential file I/O, random-access file I/O, and track and sector I/O.
- A diskette with machine-readable source codes for all programs discussed, in both Radio Shack EDTASM and Macro formats.
- Routines to convert from one assembler format to the other.

This course was developed and recorded by Joseph E. Willis, for the student with experience in assembly language programming; it is an intermediate-to advanced-level course. Minimum hardware required is a Model I Level II, 16 K RAM one disk drive system.

REMDISK-1 only **\$29.95**
Dealer inquiries invited



REMSOFT, INC.
571 E. 185 St.
Euclid, Ohio 44119
(216) 531-1338



Includes \$1.50 for shipping and handling
Ohio residents add 5½% sales tax
TRS-80® is a trademark of the Tandy Corp

AT-80 ANNOUNCES A NEWDOS SPECIAL

DDIR80 — Creates program lines of NEWDOS DIR's, adding them to itself. Options include — Search, Re-search, Run, Hardcopy, Display DIR's, Others. Stores up to 175 DIR's. 32K/one disk. \$23, w/demo.

CAT — Tic-Tac-Toe with randomly numbered squares. FAST graphics. Human vs human option. \$12.

FTDEMO80 — Displays the programs, and the keyboard commands, from the NEWDOS/80 Appendix A examples, *WHILE* executing the programs and commands and displaying results. Cycle through the five file types with only the enter key. \$12.

Disk only. Deduct \$3 each for 2nd and 3rd program ordered.

NEWDOS \$45 NEWDOS+ \$95
NEWDOS/80 \$145

Add 4% for MasterCard/Visa

AT-80

3827 Dismount
Dallas, TX 75211
(214) 339-0498

OSI (8K)

APPLE

TRS-80†



Computers & Gambling Products Magazine

PRESENTS:

PROBABILITY HANDICAPPING DEVICE 1 — A BASIC PROGRAM FOR: HORSE RACE HANDICAPPING!

This incredible program was written by a professional software consultant to TRW Space Systems. This is a complex program carefully human factored for easy use. It is a comprehensive horse racing system for spotting overlays in thoroughbred sprint races. Your computer will accurately predict the win probability and odds line for each horse based on your entries from the racing form. The next day overlaid horses can be spotted on the track tote board. The user's manual contains a complete explanation of overlay betting plus much more useful information. The appendix contains a detailed tab run of a 100 consecutive race system workout showing an **amazing 50% return** (\$1.50 returned for each \$1.00 flat wager). Includes many features such as error correction, bubble sort, line printer output, automatic keyboard debounce, archiving, etc. The manual may be ordered separately for perusal for \$7.95 and credit.

CHALLENGER 1P, 2P, or 4P 8K VERSIONS Now Available!

Phd-1 User's manual and cassette for:

Apple II (16K), TRS-80 Level II (16K), Challenger (8K) 29.95
TRS-80 or APPLE DISK 34.95

BRAND NEW FROM SDL: WIN AT THE RACES. This thoroughbred handicapping algorithm is based on a currently popular book on thoroughbred multiple regression techniques. Both sprints and routes. All of the features of PHD-1 plus more. This program incorporates the best data entry technique we've ever seen.

32K TRS-80 or APPLE CASSETTE 34.95
32K TRS-80 or APPLE DISK 39.95

BOOKS:

Winning at the Races 21.95 + .75 P&H
Beating the Races with a Computer 14.95 + .75 P&H

Make checks payable to **JOE COMPUTER DEPT U**

22713 Ventura Blvd., Suite F, Woodland Hills, CA 91364

CA residents add 6% sales tax
PHONE ORDERS: 213-992-0514

*SEND \$2.00 TO PLACE YOUR NAME ON OUR MAILING LIST.

†TRS-80 is a registered trademark of Tandy Corporation

ATTENTION SOFTWARE AUTHORS

From The Company That
Brought You *Adventure*,
by Scott Adams

We are now accepting TRS-80, Apple, and Atari software for review to manufacture under the *Adventure International* label. Join the fastest growing software company in the U.S. and enjoy a money paying hobby as well. Just send a machine readable copy of your program with documentation to: *Adventure International*, Box 3435, Longwood, Florida 32730



**adventure
international**

SOFTWARE CPU™

IF you're learning an instruction set, or analyzing an alien machine code program, or creating your own super software structures, then you are keeping instructional effects of CPU architecture and RAM all together in your head in a complex running mental map. **Whew! Instrument your imagination!** TBUG-linking **SOFTWARE CPU™** series of microprocessor simulations on the Level II 16K TRS-80™ display a complete parallel before/after set of Processor Programming Models with scrolling disassembler, CPU Registers, flags and stack, plus an intelligent RAM Window reacting selectively to RAM-interactive instructions. It's your entire imaginative overhead, clicking away in Single-step or variable speed TRACE modes under your dynamic control. Plus a slug of debugging features you'd never imagine would be available in such low cost development software. Rely program flow with a **SOFTWARE CPU™**.

Super STEP: Animated Z80 Programming Models, Disassembler, Single-step/TRACE modes with intelligent RAM Window, 5 user-selectable Windows, single and cumulative instruction times in microseconds, Reference Space, much more. Big booklet, a **Z80 Software CPU**. 16K Level II TRS-80, TBUG required. No. BL-O \$19.95

EMU 02: Animated 6502 Programming Models, Disassembles to 6502 mnemonics, Single-step/TRACE modes, 6502 counterparts to #B, #J, #F, #F and #G commands, fast Cross-Interpreter, keyboard scan port with p-instructions DB, EB control, paging in virtual address space, more. Big booklet & SYNTEREK card, it's a **6502 Software CPU**.

16K Level II TRS-80, TBUG required. No. BL-1 .. \$24.95

ACCEL: from Southern Software of England, is a **COMPILER** for Level II TRS-80 **INTEGER BASIC**. Properly structured (no dynamic redefinitions, correctly nested loops etc.) error-free BASIC programs are compiled by **ACCEL** to fast Z80 machine code for potentially spectacular speedups.

ACCEL Compiler for 16K Level II TRS-80 ... \$44.95

Include 75 each
postage, CA add 6%

ALLEN GELDER SOFTWARE
Box 11721 Main Post Office
San Francisco, CA 94101

TRS-80, TBUG™ Radio Shack/Tandy Corp.
Software CPU™ Allen Gelder Software

MODEL II GRAPHICS GAMES

You are in the space frigate *Toramo*. You are being stalked by the deadly *A L I E N*. Give your **MODEL II** a break from it's drab business processing. Test your skill with these fascinating **GRAPHICS** games.

ALIEN(\$20)* - Nerve-shattering, electrifying game. Escape the blood-thirsty alien by navigating thru the maze of the spaceship superstructure.

CROSSOVER(\$15)* - Quick thinking, strategic game to trap opponent into making critical mistake. How quick can you respond to your opponent's move?

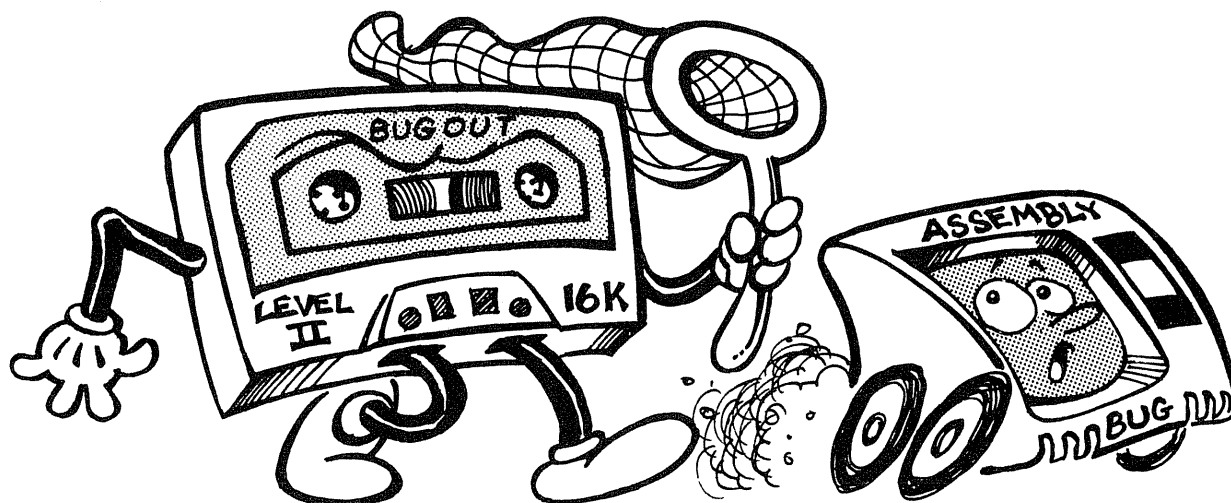
MX(\$15)* - Shoot down incoming aircraft with mobile missiles. Can you reach perfection?

TEX(\$15)* - Old fashioned shootout. Shoot your opponent before he shoots you.

(*) 1 or 2 player logic

!!!!!! Order all 4 games for \$50. !!!!!

L.D.A., 11078 Viacha, San Diego, Ca. 92124
Add \$5 to your order to cover 8" floppy.



THE ULTIMATE BUG CATCHER!

NEW PRODUCT SWEEPSTAKES

Win \$1,000 Cash

That's right! The Software Plantation is growing a whole fistful of crispy green dollar bills for someone out there...and it just might be you!

We are also growing an acre or two of exciting new ideas for your TRS-80*, including BUGOUT, the ultimate Z-80 Monitor/Disassembler/Debugger/and Assembly Language Teaching Package ever developed.

BUGOUT was written to make assembly language programming a snap for both beginners and experts! BUGOUT is the tool you need to get a firm grasp on assembly language programming and debugging, with an extra special peek into the ROMS as an added bonus! Experienced programmers will throw away their present methods once BUGOUT takes over. BUGOUT provides a powerful set of commands which gives the programmer total control over his computer!

SOUNDS GREAT! NOW TELL ME ABOUT THE \$1,000!

Never fear says our company president and resident micro-maniac. "I wrote BUGOUT, so damn the royalties and full speed ahead!" Well, that's where the \$1,000 is coming from, and everyone who requests our BUGOUT INFORMATION PACKAGE gets a shot at it! NO PURCHASE NECESSARY! Just fill out and mail the attached entry form.

Better still, order BUGOUT today and save \$5.00. BUGOUT will pay for itself immediately by saving you countless hours learning how assembly language works, and debugging your own assembly language programs! ORDER NOW! Your time is worth far more than our low introductory price of \$24.95 (regular \$29.95).

*Trademark of Tandy/Radio Shack

Creative Ideas	Quality Products
<div style="display: flex; justify-content: space-between;"> <div style="width: 40%;"> </div> <div style="width: 60%; text-align: center;"> <p>The SOFTWARE PLANTATION, Inc. PO Box 44623 Tacoma, WA 98444 (206) 531-1506</p> </div> <div style="width: 40%;"> </div> </div>	
<p>Grown Exclusively for the 80's</p>	

ROYALTY SWEEPSTAKES ENTRY FORM

Complete sweepstakes rules and procedures enclosed with BUGOUT or BUGOUT INFORMATION PACKAGE.

No purchase necessary Void where prohibited by law.

TO PURCHASE BUGOUT, complete and mail to:

THE SOFTWARE PLANTATION, INC.,

P.O. Box 44623, Tacoma, WA 98444

☐ Check ☐ Money Order ☐ Master Card ☐ Visa
 Card No. _____ Expiration date _____

Washington residents add 5.3% sales tax

☐ Please rush _____ BUGOUT'S @ \$24.95
 and enter my name in the sweepstakes
☐ Please rush BUGOUT INFORMATION PACKAGE
 and enter my name in the sweepstakes

Name _____

Address _____

City _____ State _____ Zip _____

Microsoft's Basic Compiler

Truman Krumholz
Springfield, MO

One of the newer pieces of software for the TRS-80 Model I is Microsoft's Basic Compiler. No doubt future articles on it will go into more depth and explain it in much greater detail than this brief overview. My purpose here is to give enough information to perhaps allow one to decide whether or not to purchase it.

The Basic Compiler comes on two diskettes accompanied by a rather large manual. Most of this manual describes Microsoft's Basic-80. However, the section concerning the compiler as it runs on the TRS-80 is quite adequate. The compiler is easy to use and is surely the most painless way for most of us to realize faster running programs.

The procedure to use the compiler is to either write or load a Basic program and then run it with the interpreter to debug it. Then save the Basic source with the extension /BAS. It must be saved in ASCII form, that is, SAVE "PROGRAM/BAS",A. Now, do a CMD "S" to return to DOS and type BASCOM=PROGRAM. This will perform a syntax check of your source code. If all is well the program can now be compiled either with or without a listing file. Compiling is fast and as soon as the prompt appears, type L80 PROGRAM to start the linking process. At the conclusion of L80 a list of the library routines needed for the program is displayed. Now the second

disk is mounted and we type BASLIB-S and the machine code to perform the Basic program is put together. This is the lengthy process. I have had it take anywhere from three minutes to ten minutes or more. Finally, we type PROGRAM-N-E and a CMD module is written to diskette. Our Basic program is now a machine language program named PROGRAM/CMD.

The manual indicates that programs will run from three to thirty times faster when they are compiled. My experience shows this to be about right. The use of integers seems to be very important in improving speed. A precision multiplication program multiplying two forty digit numbers and using only integers in the program, ran thirty two times faster after it was compiled. A prime number finder only ran two and one half times faster but the variables were single precision. Other programs I have tried have fallen in between these extremes, but all have run faster. Graphics using integers are spectacular. I have had to put delays in some moving graphics programs to decrease the speed.

Up to now, all I have discussed has been on the plus side. There are a few negatives. Probably, number one on this list is the price. It is the most expensive software that I have purchased. Another thing that one will immediately notice is the great

amount of memory used for the CMD module. It is particularly apparent with short programs. A 4K Basic program may generate a machine code program of 12K or more. Generally, the longer the program, the more efficient the machine code. However, a 15K program of mine failed due to not enough memory. This depends on the complexity of the program. (Number of command words and number and length of strings, arrays, etc.

I have heard that there are a number of errors in the system but I have found only two. My compiler does not recognize "RUN" which is supposed to be the same as "CHAIN". "CHAIN" does function correctly. The other possible error is that when using BASLIB, the manual indicates that the -S switch is not usually necessary. I find with my system that it is necessary.

What is my conclusion at this time? If you need more speed in your Basic programs, the compiler will give it to you, spectacularly, if you can stay with integers. A Basic game of Gomoku that did take up to two and one half minutes to move now moves within five seconds. This, I think, is the prime consideration. Do you need more speed with your Basic programs? If the answer is yes then Microsoft's Basic Compiler is one way to achieve that goal. ●

UNCLASSIFIED ADS

\$2.50 per half inch and \$2.50 for each additional half inch per insertion. Ads must be typewritten or printed clearly, and must be accompanied with payment in advance. No telephone orders or "bill me's" will be accepted. Unclassified Ads are always set in the type on this page, with the first line in bold type. Use the ads on this page as a guide to figure your size and payment. Send to: 80-U.S. Journal, 3838 S. Warner St., Tacoma, WA 98409

UNIQUE TRS-80 SOFTWARE & BOOKS

Disassembled Handbook for TRS-80 all postpaid.
Vol 1 5th printing "save one year's study" \$10.
Vol 2 3rd printing "all you wanted to know" \$15.
Vol 3 New 1981 "exploring new horizons" \$17.
Morse code transmit program: cassette/disk \$15. Richcraft Engineering Ltd. Box 1065, 1 Wahmeda Industrial Park, Chautauqua Lake, NY 14722 Phone (716) 753-2654 for COD orders (111)

TRS-80 HARDWARE & SOFTWARE FOR

sale due to system upgrading. Send SASE for list.
R Lee, 25 Amaryllis Ave, Waterbury, CT 06710 (011)

DISK REPAIR - MPI, PERTEC, SHUGART

disk drive repair. Flat rate of \$35.00 plus parts, add \$2.00 shipping. ALL SYSTEMS GO, PO Box 915 105 West Plant St., Winter Garden, FL 32787 (305) 351-6527 (011)

RELOCATE T-BUG TO HIGH RAM. RUN

Basic with T-Bug in RAM. Quick return after RESET. Probe the reserved RAM. Level I. Listing \$4.00 David S Robertson 2810 SW Caulder, Des Moines, IA 50321 (031)

DATA BASE MANAGER - IDM - "There are

many data base managers available these days, some may do the job for you, others may not. This one is doing a very good job for us." - 80-U.S. Jul '80. Now, new documentation and fastest machine sort. Mod II \$199. Mod I 48K \$149. Micro Architect Inc., 96 Dothan St., Arlington, MA 02174 (111)

INCREASE WIN PROBABILITY WITH

harness racing for TRS-80. We believe this is the most advanced harness racing handicapping program available anywhere. Years of research have gone into the development of the algorithms used. Correctly used, a greater return on your track investment will be yours. Requires minimum 16K Level II. Only \$24.95 (add \$1. shipping; CA residents add 6 1/2 sales tax). Send check or MO to WelMur of California, Inc. PO Box M-11B, Fremont, CA 94537 **Coming soon - DOG Racing** (031)

FOR SALE: WORKING TC-8 HI-SPEED TAPE

system for \$90. Gone disk. Inquiries answered. G Clark 32 High St., Newport, ME 04953 (207) 368-5295 after 6 PM (011)

LINE PRINTER 1 (Centronics) FOR SALE HAS

tractor feed, manual, no cable. In original condition. \$650. Contact Mike at 80-U.S. for details. (206) 475-2219 (011)

LINE PRINTER II FOR SALE ALMOST NEW

with manual, cable, and carrying case. \$700. Contact Mike at 80-U.S. for details. (206) 475-2219 (011)

DRAW POKER - SOUND AND GRAPHICS

program simulates the quarter game you may have seen in various places of business. But you do not need quarters to play this game. Excellent graphics & sound effects. Required: TRS-80 16K Level II. Only \$14.00 from Michael D Simon, 25 Flagship Road, Baltimore, MD 21222 (031)

HIGH SPEED TAPE UTILITY - USEFUL FOR

fast Basic data storage on tape. A Basic program POKE's the machine language routine used for this utility. Can be added to the beginning of any Basic program. Reserves top 256 bytes of memory for other machine language routines needed by the user. Saves integer, single & double precision, and string arrays. Verifies & reads arrays also. Uses standard 500 baud tape speed, but will save a 100 element integer array in 7 seconds. Basic "PRINT#" statements would take 15 minutes. Routine called from Basic with USR function. Available for Level II - specify 16K, 32K or 48K. \$10. - Small Systems Consultants, 2355 Pilot Knob Drive, Santa Clara, CA 95051 (011)

WANTED: BASIC ADVENTURERS WANTED

for new book on Adventure. Buying first book and first reprint rights plus offering 20% royalty on magnetic reproduction rights. Line listing must be clean and free of packed strings or other unlistables. Themes needed include Dungeons and Dragons, Space Time Warp or other solid Adventure themes. Top dollar offered. Send tape to Captain 80, Box 66, Peterborough, NH 03458 or call (603) 924-6065 after 6 PM EST (011)

CASSETTE DUPLICATING. NEW LOW

rates for small quantities. Top quality cassettes. All tapes guaranteed. Write for rates. EDU-WARE, PO Box 336, Maynard, MA 01754 (011)

BEAUTIFUL PROGRAM LISTINGS USING

IBM electronic typewriter. 8 1/2 x 11" bond paper. Film ribbon. Page numbers. Title (specify) on each page. State single, 1 1/2, or double spacing. Optional copyright notice (specify name/year). Keyboard symbols correctly printed. Indented program continuation lines. Send cassette, \$1. +1 1/2¢ per program line (0-63999 allowed). Robert James, 12010 Cabana Lane, Austin, TX 78759 (011)

BARGAIN! REAL ESTATE & FINANCIAL

analysis program for 16K Level II, \$19.95 includes documentation. Includes complete amortization schedule with yearly interest summary, cash flow analysis, depreciation, and more. Marlin's, 4530 Mission Court West, Columbus, IN 47201 (011)

SOLAR COLLECTOR DESIGN PROGRAM

for 16K Level II. Extensive design and sizing calculations. Field tested results. Documentation, program tape and technical references all for the low price of \$29. Shows you how to design the best collector for the investment. Marlin's 4530 Mission Court West, Columbus, IN 47201 (011)

TRS-80 SUPER GAMES FOR 16K Level II.

10 exciting & challenging games on C90 cassette just \$5.00, cash only. Castlestar Products, Dept CS-40, 8753 Windom Ave., St Louis, MO 63114 (011)

NIGHT FIGHTER: Exciting real-time

simulation of a night interception mission in WWII. The ground controller directs you to the target, you acquire him on your cockpit radar, and the chase is on. Your aircraft can climb, dive, turn, speed up, slow down - but so may the German bomber. Program combines Basic, packed graphics, and machine language to speed action and enhance displays. Three levels of difficulty. On tape (\$10.00) or disk (\$15.00). Specify L2 or Disk, MEMSIZE. Rod Fitzgerald, Box 12709 Dallas, TX 75225 (011)

THE ERCC NEWSLETTER IS PUBLISHED

monthly by Educational, Recreational Computer Club. Membership in ERCC costs \$5.00 including a subscription to the newsletter. Club address is ERCC, Box 325 Owosso, MI 48867. The club meets the second Saturday of each month to discuss ideas, trade programs and to review new computer products. ERCC maintains a newsletter exchange with other clubs. To join the exchange, simply send us your newsletter. Non-profit computer clubs are free to reprint material from this newsletter, giving appropriate recognition to the source. The club meets at the Pines Country House Restaurant & Motel Banquet Room "A". Program time 5:00 PM, Dinner 6:00 PM. Address is 1730 E Main, Owosso, MI (517) 725-5164 OR Fly to the Owosso Airport, then taxi & park right at the Pines. (011)

FOURTEEN TRS-80 DISK BASIC

Programs for a buck apiece: Global Search & Replace utility, Travel Expense Report, Appointment Secretary, Construction Materials Cost Calculator, Programmable Audible Process Timer, Television Guide, Auto Cost Calculator, Magazine Subscription Records, Record Album Timing. Games: Donut Shop, Card Sharks, Phrase Challenge, Star Wars, Market. Supplied on disk: \$13.94+ \$1.00 postage. Busch, 515 E Highland Ave, Ravenna, OH 44266 (011)

TRS-80 TO H-14 PRINTER INTERFACE - NO

software driver required! Print at 4800 baud with full handshaking & printer-status support. Interface connects to parallel printer card-edge of Radio Shack expansion interface. Parallel data is converted to serial form to drive the H-14. No software driver program is needed - the computer "thinks" it is connected to a parallel (Centronics-type) printer. This unit connects externally to your expansion interface so you don't void your Radio Shack warranty. The PTS-3 interface is assembled and tested and is covered by a limited 6 month warranty. Call or write for more info: Multi Media Systems, PO Box 41084, Indianapolis, IN 46241 (317) 839-6520, or send \$89.95 plus \$3.50 shipping & handling. (011)

UNIQUE TRS-80 SOFTWARE/BOOKS Morse code transmit/receive program 5-25 WPM, no ancillary devices reqd: \$15. disk/cassette ppd. Disassembled Handbook all Basic CALLS: \$10. ppd. TV Satellites az-el-distance ur location: \$5. ppd. Decimal/Binary/Hex/Split Decimal/Split Hex both-way conversion pgm to 65K. \$10. disk/cassette ppd. Richcraft Engineering, Box 1065 Chautaugua Lake, NY 14722 Phone (716) 753-2654 for COD orders. (011)

L216 IS A CASSETTE BASED BUSINESS package consisting of: A word processor, Data Base Manager, Inventory Control, Check Balancing program, Label Generator, Statistics, Sales analysis, Sort and Linked access. All for \$59. Super Bargain! Microtek printer at \$770. Verbatim 5" disk \$29 for 10, 8" disks \$41 for 10. Micro Architect Inc, 96 Dothan St., Arlington, MA 02174 (011)

MOD II INTEGRATED ACCOUNTING package. Full functions GL. A/R prints invoices and statements, supports forward balance and open items. A/P helps to select which vouchers to be paid with total cash management. Payroll features user modifiable tax tables. Print W-2, 941, checks, etc. Fast ISAM. As opposed to most other systems, we use 80 col screen instead of 64 and require 132 col printer (not 80), and TRSDOS 1.2, (not CP/M). A similar version using a different tradename is sold for 1000 each, ours is only \$129. Includes disk, postage, 100 page manual, an installation guide, and 1 year updates. Micro Architect Inc, 96 Dothan St., Arlington, MA 02174 (031)

OKIDATA MICROLINE 80 PRINTER 3 PART roll/sheet/fanfold paper. 40/80/132 chars/in. Handles S-80 graphics! Connects directly. Used 3 months. List \$945. selling at \$650. First MO takes it! Val Plavan 9309 Folsom Blvd. Sacramento, CA 95826 (011)

GREATEST BARGAIN! (1) LEVEL II BASIC, 16K software package: word processor, inventory, data base manager, check balancing, statistics, deposit calculator, stock management, label printer, sales analysis, sort and linked list utility. All 11 programs for \$59. (2) Integrated accounting package: A/R, A/P, GL and Payroll. \$129 each. \$20 for manual alone. For TRSDOS I, TRSDOS II, CBASIC and MBASIC on CP/M. For FREE catalog, send SASE (28c) to MAI, 96 Dothan St., Arlington, MA 02174 (051)

FOR TRS-80 RS-232 BOARD. FIX INTER-mittent problems due to intermittent PC board connector with a connector brace. Easy to install. Only \$4.95. Gunn RS-232 board with two 8-bit in & two 8-bit out ports. \$35. Bare board and schematic. Schematic \$1.00 (TX residents add 5% tax). Gunn Industries, 704 Franklin Blvd, Austin, TX 78751 (071)

TWO BUCKS AND UP FOR TRS-80 programs on cassettes. Dozens of unusual programs. Free Catalog. PEC, PO Box 42831, Las Vegas, NV 89104 (011)

HEATH H89 COMPUTER W/32K EXTRA memory. Still in boxes. \$1600 or assembled \$2200. Savings of \$250-600! Steve Larson, 1525 S Lansing, Aurora, CO 80012 (303) 752-3768 (011)

TELETYPE ASR33 COMPLETE W/STAND, paper, tape and manuals. Just refurbished by Teletype. Works perfectly. \$400 or best offer by Mar. 1st. Steve Larson, 1525 S Lansing St, Aurora, CO 80012 (303) 752-3768 (011)



"OUR LAST LEVEL II SOFTWARE directory - and it's more complete than the current Sourcebook. Ask for our new Summer issue, \$4.00 ppd. Future issues will be devoted to software reviews. R E Purser, PO Box 466, El Dorado, CA 95623" (011)

PERSONAL FINANCE - USEFUL FOR maintaining checkbooks on computer, tracking personal expenses, keeping track of tax deductible expenses. Post checks from 6 checking/savings accounts to 40 expense categories for a total of 46 categories specified by name for easy reference. Allows multiple & split posting. Holds up to 800 checks in memory for a single month. Search function finds all checks on a given date, all checks to a given party, etc. & displays checks found. Allows you to enter cancelled check#'s, balance the checkbooks, saves cancelled checks on tape for future reference. Maintains status for each expense category - amount this month, amount last month, monthly avg. this year, total this year. Similar status for checking accounts. All status saved with cancelled checks on tape each month. Any display can be printed on a line printer. Saves 100 checks on tape in 90 seconds using built-in high speed machine language routine. Available for Level II cassette specify 16K, 32K, 48K. \$15. - Small Systems Consultants, 2355 Pilot Knob Drive, Santa Clara, CA 95051 (011)

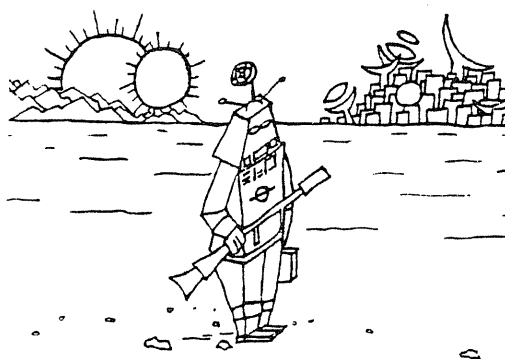
ABACUS[®] software

Maillist	29.95
Tapecopy	19.80
Text Editor	69.95
Inventory	29.95
LowercaseKit	24.95

and much more!!!!
ASK YOUR DEALER OR
Call or Write: ABACUS

PO Box 77  
E. Greenbush, NY 12061
518-477-8222

CYBORG WARS



UP TO 4 PLAYERS COMPETE WITH EACH OTHER
AND/OR COMPUTER TO ACHIEVE WORLD PEACE.
BALANCE OF POWER FLUCTUATES AS PLAYERS
DEVELOP RESOURCES, USE ESPIONAGE, EXPLOIT
ALLIES, UTILISE MILITARY POTENTIAL.

TAPE CASSETTE FOR TRS-80 16K LEVEL II MODEL I

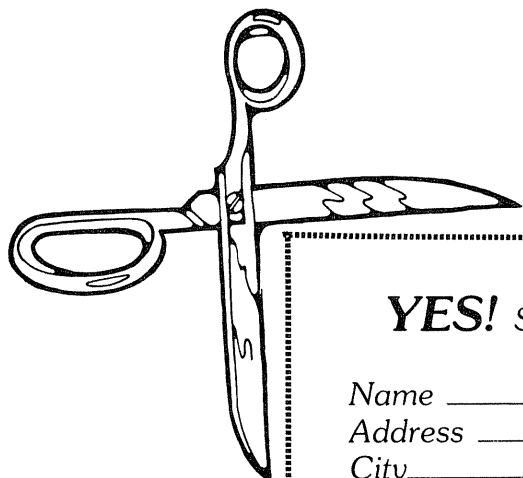
\$15

STRATAGEM CYBERNETICS
286 Corbin Pl., B'klyn, N.Y. 11235

80-U.S.

THE TRS-80 USERS JOURNAL

Now that you've enjoyed this issue of 80-U.S. and seen our new look don't you think it's about time you subscribed? 80-U.S. has proven in the last two years that it can give you more information on the TRS-80 than any other single source. The magazine has grown with each issue, we have better and more innovative ideas for all TRS-80* owners. Don't miss a single issue, subscribe today for only \$16!*



YES! Sign me up as a subscriber to 80-U.S. for \$16 a year!

Name _____
Address _____
City _____ State _____ Zip _____
Visa/MasterCharge # _____ Exp date _____
Signature _____
Bill Me _____

(206) 475-2219

All foreign subscription: \$30 one year only.
Back issues that are available are \$3 each.
Your subscription will begin with the next published issue.

**Mail to
80-U.S. Journal
3838 South Warner Street
Tacoma, Washington 98409**

LUNAR LANDER

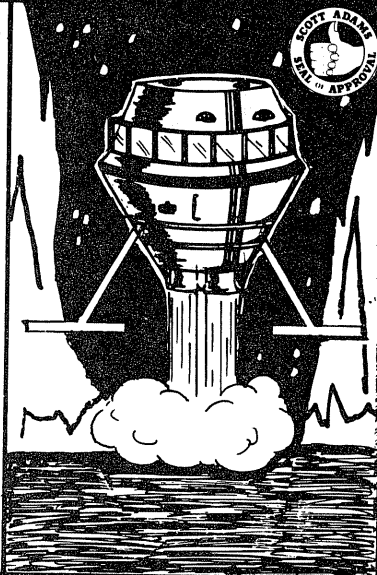
JUST RELEASED! THE MOST FANTASTIC ARCADE GAME EVER FOR YOUR TRS-80!

SUPERIOR GRAPHICS & SOUND EFFECTS

A real time arcade simulation of a LUNAR LANDING

**TRS-80
Models 1 & 3
Level 2
16K TAPE - \$14.95
32K DISK - \$20.95**

JUST LANDING AT YOUR LOCAL COMPUTER STORE OR WRITE TO:



ai Adventure INTERNATIONAL
A DIVISION OF SCOTT ADAMS, INC.
BOX 3435 • LONGWOOD, FLA. 32750

Let Your TRS-80® Teach You ASSEMBLY LANGUAGE

Tired of buying book after book on assembly language programming and still not knowing your **POP** from your **PUSH**?

REMsoft proudly announces a more efficient way, using your own TRS-80®, to learn the fundamentals of assembly language programming --at YOUR pace and at YOUR convenience.

Our unique package, "**INTRODUCTION TO TRS-80® ASSEMBLY PROGRAMMING**", will provide you with the following:

- Ten 45-minute lessons on audio cassettes.
- A driver program to make your TRS-80® video monitor serve as a blackboard for the instructor.
- A display program for each lesson to provide illustration and reinforcement for what you are hearing.
- A textbook on TRS-80® Assembly Language Programming.
- Step-by-step dissection of complete and useful routines to test memory and to gain direct control over the keyboard, video monitor, and printer.
- How to access and use powerful routines in your Level II ROM.

This course was developed and recorded by Joseph E. Willis and is based on the successful series of courses he has taught at Meta Technologies Corporation, the Radio Shack Computer Center, and other locations in Northern Ohio. The minimum system required is a Level II, 16K RAM.

REMASSEM-1 only \$69.95



REMsoft, Inc.
571 E. 185 st.
Euclid, Ohio 44119
(216)531-1338



Include \$1.50 for shipping and handling.
Ohio residents add 5½% sales tax.

TRS-80® is a trademark of the Tandy Corp.

ADVERTISER INDEX

80-U.S. Journal	127
Abacus Software	126
Access Unlimited	9
Acorn Software Products Inc	7
Adventure International	93,99,122,128
Aerocomp	5
Allen Gelder	23,122
AT-80	122
Audio Video Systems	32
Basics & Beyond	101
Byte Miser Software	103
CalData Systems	114
Comsoft	95
Creative Computing Software	47
D & M Software	103
Discovery Bay Software Co	33
Edu-Ware	103
Electronic Specialists	103
Epson America Inc	1
Exatron	Cover 4
Galactic Software Ltd	82
Hexagon Systems	107
Insiders Software	87
Instant Software Inc	24,25,31
Joe Computer	122
Lakefront Software	107
L.D.A.	122
Level IV Products Inc	37
Lobo Drives International	11
Lobo-Galactic	51
Lords	83
Marway Products Inc	107
Max Ule Co Inc	107
Med Systems Software	73
MI-Prog	55
Micro Architect Inc	104
Micro Computer Devices	95
Micro Club	77
Micro Management Systems	105
Micro Mainframes	14
Micro Peripheral Corp	116
Micro Systems Software Inc	84
Miller Microcomputer Services	116
Misosys	97
Pacific Office Systems	91
Percom Data Co Inc	Cover 2
The Program Store	28,29
Programma International Inc	71
The Programmers Guild	109
Prosoft	111
Remsoft	122,128
Rochester Data	14
Simutek Computer Products	50
Small Business Systems Group	115
Snapp Inc	49
The Software Exchange	57 - 64
The Software Plantation	123
Sterling Computer Products	119
Stratagem Cybernetics	126
The Alternate Source	101
Unique Software	119
VISTA Computer Co	Cover 3

Back Issue Availability

The following back issues of 80-U.S. are still available: July/August 1979, November/December 1979, January/February 1980, May/June 1980, September/October 1980 and November/December 1980. Price of back issues is \$3.00 per copy, postpaid. Please allow 3 to 4 weeks for delivery.

When you pick a Daisy... Pick Vista's V300 Printer.



\$1895.

There are Daisies! . . . And, There are Daisies! . . . But Vista has a Peach!

The Vista V300 is exactly that, a "peach" of a daisy wheel printer both from the standpoint of price and performance.

Think of it, a printer at nearly half the price (when compared to models even remotely competitive in quality) combined with the ultimate in reliability, print quality, and flexibility.

Typical Comments: "Superb print quality!", "Highly reliable.", "Definitely letter quality. . . I can't believe the price tag.", "Best use I've seen yet of LSI Technology."

But judge for yourself — look at the V300 features and keep in mind this is a letter quality printer at dot matrix prices.

- **Tractor option available**
- **Print Speed** — 25 CPS (Optional 45 CPS for \$2,195)
- **Print Wheel** — Industry standard 96-character Daisy Wheel (including the extended-life dual plastic wheels)
- **Service** — Prompt maintenance/service agreements available nationwide
- **Interface** — Industry standard parallel (RS232-C optional)
- **Printable Columns** — 136
- **Warranty** — 90 days parts and labor, one year parts only
- Proportional, bi-directional printing • Programmable VFU
- Extensive self-test functions • Hardware and software compatible

Vista does it again! Quality, Price and Performance with a peach of a daisy wheel printer.



IMMEDIATE DELIVERY
For Further Information
Call Toll Free (800) 854-8017

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

AND, Vista Has a Complete V100 Word Processing System for Only \$4995!

The Vista V100 is a complete word processing system that includes:

- | | |
|---|---|
| • Exidy Sorcerer Computer, 48K | • Vista V300 Printer Full Character Daisy Wheel |
| • V200E20 Disc Drive System, Double Density | • Wordstar, CPM 1.4 (Includes E Basic) |
| • Sanyo Data Display Monitor | • Can also be used for Data Processing |



ECONOMICAL

alternative

to disk

systems!

The
exatron
Stringy Floppy

- Load 16K in 24 seconds!
- Load and Save at 7200 baud!
- No Expansion Interface required.
- No volume control to contend with.
- Priced at less than a single disk drive!
- Small, easy to connect to your keyboard.
- Saves/Loads Basic, Machine Language and Data Files.

Ask about our 32K Memory Expansion Unit!

Call our TOLL FREE
number today!
800 538-8559

exatron

excellence in electronics

181 Commercial Ave.,
Sunnyvale, CA 94086

The 80-U.S. Journal
3838 South Warner Street
Tacoma, Washington 98409

Second Class Postage Paid
at Tacoma, Washington 98413
And Additional Mailing Offices
USPS Publication No. 500-470
(ISSN 0199-1035)