80 U.S.

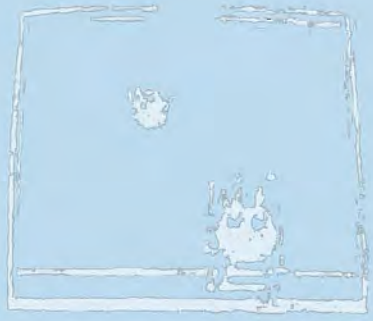
Vol IV Number 3 - $3.00 per Copy

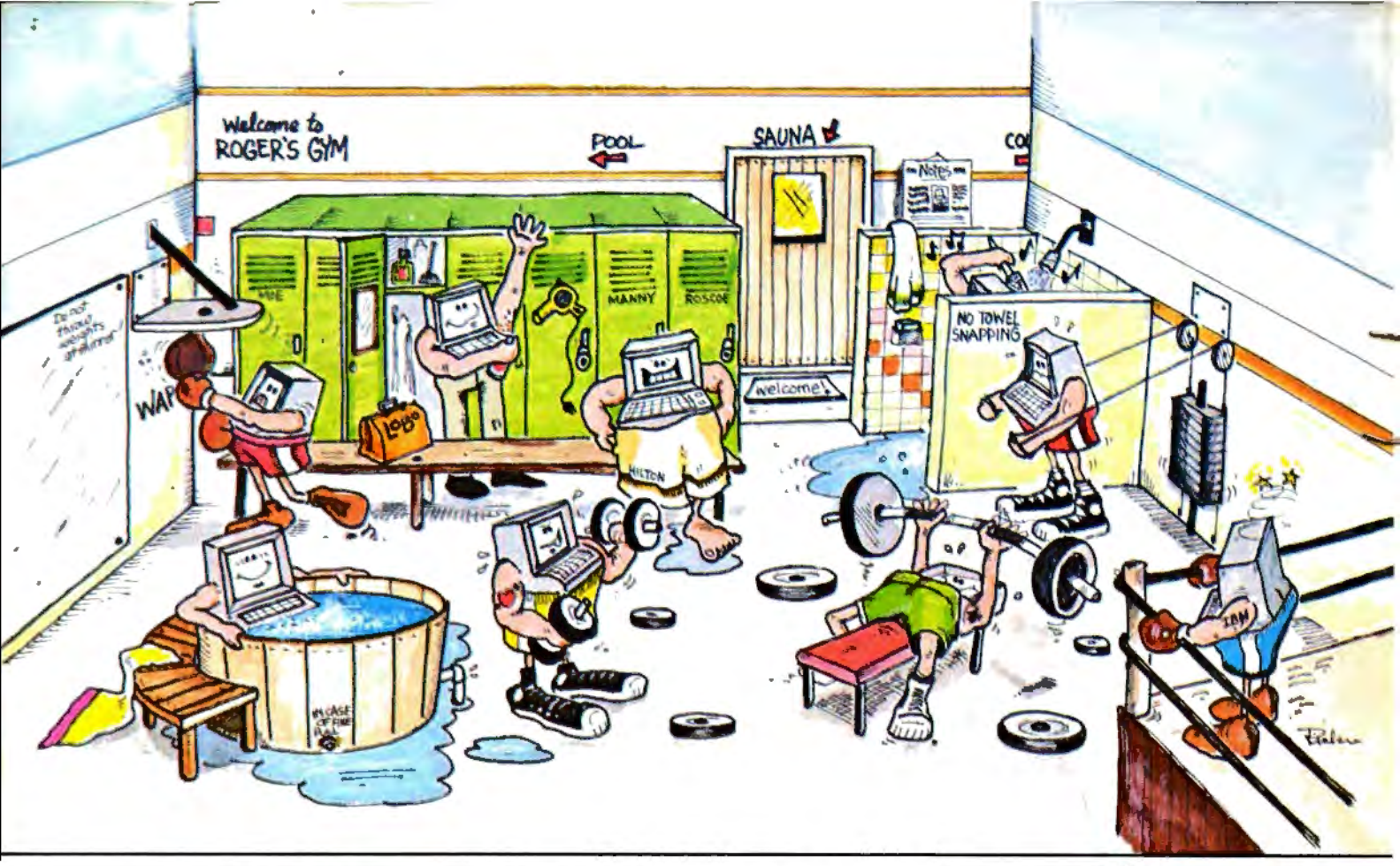
THE TRS·80 USERS JOURNAL

May/Jun 1981

Leo Christopherson:

The "Disney" of TRS-80 Animated Graphics, tells you how to "LINE PACK" a program!





LOBO'S NEW LDOS™ Puts muscle in your TRS-80*

LOBO DRIVES' new LDOS™ Disk Operating System is loaded with outstanding features that will enable you to realize the full power and potential of your TRS-80.* With LDOS, you can support up to eight drives (5¼ and 8-inch drives, double-sided drives, double-density drives, 80-track drives), including the new 8-inch and 5¼-inch Winchester fixed disk drives, in any combination.

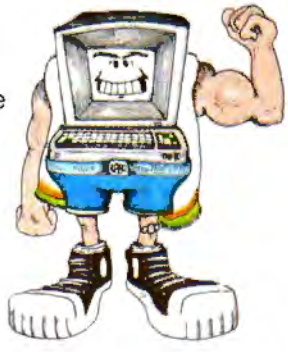
Other LDOS muscle building features include: ISAM accessing techniques; keyboard typeahead; Graphic string packer; Dated files, Marked files; File transfer by class; Built-in lower case display drivers; Non-breakable AUTO and DO commands, and many, many more.

LDOS is the perfect operating system to use with your LOBO DRIVES LX-50 or LX-80 expansion interface and disk drive subsystems. There's even an 800 number for instant service. To find out how you can put more muscle into your TRS-80, contact your nearest LOBO Drives dealer or call or write.

LDOS is available:

A. Operating System Diskette with Master Reference Manual
$139

B. Master Reference Manual Only
$25



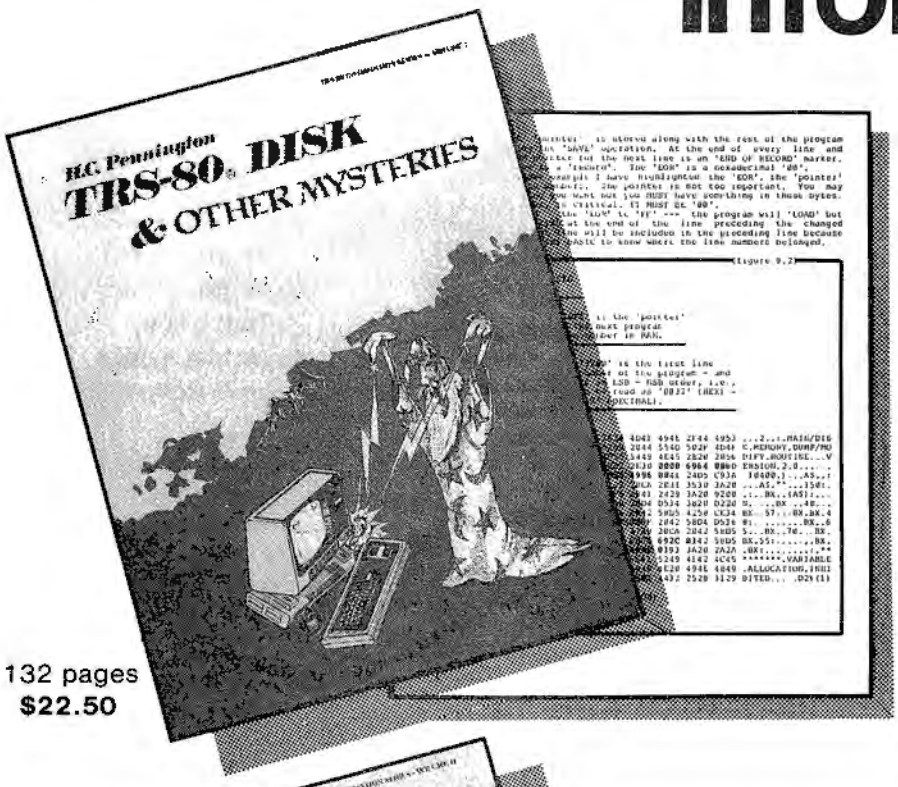
* TRS-80 is a registered trademark of Radio Shack, a Tandy Company.

LOBO drives INTERNATIONAL

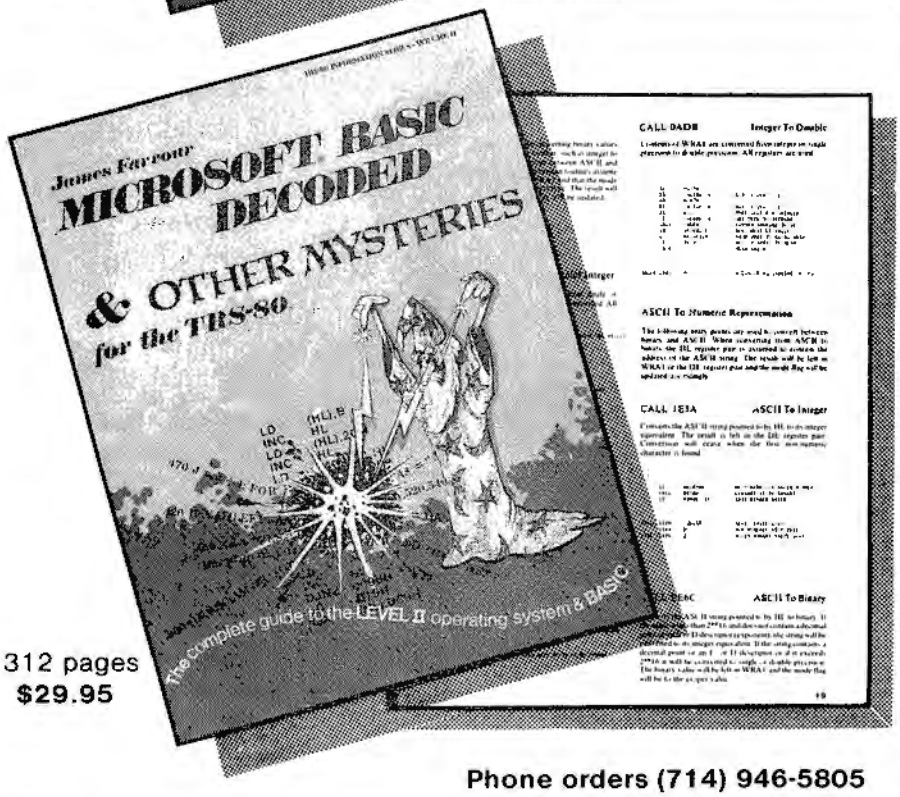
LOBO DRIVES, INT'L
354 South Fairview Ave.
Goleta, CA 93117
(805) 683-1576

'.. this is not only a worthwhile book but a great book. My advice is to get it and USE it!' - **William Barden Jr.**

'HOW TWO' information!



132 pages
$22.50



312 pages
$29.95

TRS-80 Disk & Other Mysteries is the definitive **fixit** book for disk users. More than 130 pages of easy to read, entertaining and immensely useful information - find out how to recover disk files, the layout of information on disks, memory maps, problem solutions .. the list goes on!

Many readers have saved days of work by recreating disk files that were unreadable. With more than 14,000 copies sold so far, **TRS-80 Disk** & Other Mysteries has received favourable reviews in several magazines. Yours for only **$22.50** (plus $1.50 shipping, CA residents add $1.35 sales tax).

Microsoft BASIC Decoded & Other Mysteries is the definative guide to your Level II ROMs. With more than 7,000 lines of detailed comments, and 6 additional chapters packed with information, it is easily the biggest and best book about the Level II ROMs available.

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

Exploit the full power of Microsoft BASIC, with the aid of more than 300 pages of tested examples, explanations and detailed comments. Only **$29.95** (plus $2.00 shipping, CA residents add $1.80 sales tax).

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

IJG Computer Services,
1260 W. Foothill Blvd.,
Upland, CA 91786

Please send me a copy of TRS-80 Disk & Other Mysteries for $22.50 plus $1.50 shipping. ☐

Please send me a copy of Microsoft BASIC Decoded & other Mysteries for $29.95 plus $2.00 shipping. ☐

OK, send me both IJG books for $52.45 plus $3.50 shipping. ☐

Name .....
Address .....
City .....
State ..... Zip .....
Charge my
☐ Mastercard Interbank Code .....
☐ Visa Expiration Date .....
Card Number .....
☐ Check enclosed
☐ Ship COD ($2.00 extra)

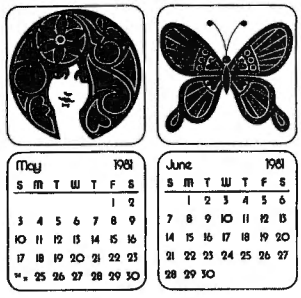
Signature .....

'It has twelve thousand one-liners in it, and every one is great!' - **Dennis Kitsz.**

Phone orders (714) 946-5805

IJG Computer Services,
1260 W. Foothill Blvd.,
Upland, CA 91786

Circle 2



Publications are like any other business. It may not seem so to the casual observer, but it's true. They have the same problem as everyone else - income, expenses and growth.

It is almost a full-time activity of the publisher to worry about cash flow and the state of the business. The business must support those who work there in a style at least somewhat close to what they are accustomed to. (Sometimes you even settle for less.)

Other things not immediately apparent to the casual observer include such things as a reduced price subscription drive. Haven't you ever wondered how some magazines can give you such a deal on a first-time subscription. It almost looks like they give it away for less than cost.

There are many things to balance in running any business. At todays printing and postage costs, the price of a subscription is literally eaten away. The profit, if there is any, must come from advertising revenue.

An advertiser wants to know how many people will see his ad per advertising dollar, and rightly so, since he doesn't want to buy a pig in a poke. Sheer numbers are important. The advertiser not only wants to know how many dollars per thousand readers, he expects to see results, and will measure the success of his advertising campaign by the returns he receives.

The publisher and editor, on the other hand, must insure that the readers are satisfied with the editorial content, so that the publication will continue to sell. The advertising versus editorial content of the publication needs to be fixed at some ratio, determined by the "horse sense" of the editor/publisher. The advertising percentage must be large enough to pay the freight, but not so large that articles of interest to the reader are left begging. The article content must be picked carefully, so that there is something in each issue for the majority. Incidentally, advertising in itself can be an education at times.

Getting the right "mix" is interesting, challenging, fun and frustrating - all at once. It's difficult to get the "feel" of the mood of both readers and advertisers ahead of time. You almost have to screw up before you know about it, and the fun part of it all is trying not to.

Back in the good old days of 1978 and early 1979 I ran the Journal single-handedly from the family room of my residence. The overhead was low, but so was the income, and sometimes I went a month or two without pay. But it was exciting and fun. Almost all the production was farmed out (and it showed, but I didn't know any better), and there were actually a couple of weeks between issues with nothing to do.

Now, we have our own building and production is done in-house (all except the actual printing). We have a very dedicated group of people working full time on the Journal and are growing rapidly - so rapidly that at times it hurts. The individual attention we used to give our old and faithful readers is sometimes not there anymore. Renewal notices are run on a regular schedule and sometimes your payment and our notice cross in the mail.

Have faith gentle reader, we still believe you are the most important commodity we have, and we will continue to customize and streamline our operation to cover the most with the best.

Our recent advertising campaign, featuring a reduced rate, first-time subscription (which ended 31 Mar 81) was a tactic designed to increase our numbers, and thereby increase the value of the product we provide. Our heaviest attrition is during this period, and we needed to bolster our cash-flow and even up both ends. It exceeded our expectations, and the end result is that you who did not benefit directly from the reduced price will have an even better Journal in the issues to follow.

We have seen too many really good publications come and go. They go mostly due to bad judgement or management. It is still a case of dollars in versus dollars out (and make a good saleable product in between). Hang in there folks, we intend to be around for a long time!

Mike

80-U.S.

JOURNAL

Editor/Publisher
I Mike Schmidt
Sales/Promotion
Margaret Farrell
Managing Editor
Tom Huber
Associate Editor
Terry R Dettman

Contributing Editors
Phil Pilgrim
Larry S Panattoni
Bill Schroeder
Spencer Hall
R C Bahn

Art/Layout/Design
Grace McNamara
Comptroller
Maggie Murray
Circulation
Jill Blackburn
Typography
Anyone who can
Reviewers
Cameron C Brown
Pat Perez
W W Harper II

Printed by
Peninsula Gateway Press
Gig Harbor, Washington

DISTRIBUTORS
Europe
Hofacker Verlag
Tegernseer Strasse 18
D-8150 Holzkirchen/Obb
West Germany
United Kingdom
The Software House
146 Oxford Street
London, W.1 England

The Cover

Our cover this issue shows Leo Christopherson against a menagerie of his creations. Photography by Terry Dettmann.

ISSN Publication #0199-1035

Published bimonthly in the months of January, March, May, July, September and November by **80-Northwest Publishing Inc 3838 South Warner Street, Tacoma, Washington 98409**

Subscription price in the United States is $16.00 for one year, $31.00 for two years and $45.00 for three years.

Foreign Subscriptions (except Canada): $30.00 for one year only, remitted in U.S. funds please.

Canada: $20.00 for one year only.

All issues are sent via surface mail.

Printed in the United States of America

POSTMASTER: Please send change of address form 3579 to 80-U.S. Journal, 3838 South Warner Street, Tacoma, WA 98409

Second Class Postage PAID at Tacoma, Washington

80-U.S.

THE TRS-80 USERS JOURNAL

*TRS-80 is a trademark of the Tandy Corporation

Volume IV Number 3 May/June 1981

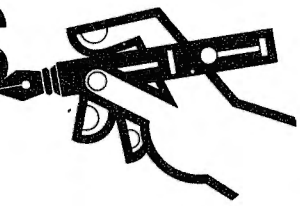
IN THIS ISSUE

Features

Reviews

Departments

LETTERS



Would you please thank Andy and his pal for the calendar program. It works!

I have twin TRS-80 II's with disks, printer, and a Daisy Wheel. I am not displeased which, considering my "mainframe" of mind, is a very superlative statement.

In a back issue, have you ever done an article on the disk drives? I am looking for an article about disk reads and writes. I know nothing about their design, but I am now beginning to believe that the disk drives are not mechanically separate.

I have a series of payroll, check printing, and report producing programs which use the TRSDOS on "0" with the weekly and year-to-date data on Drives 1, 2, or 3. Some of these programs read and write with hardly a murmur while others in the series sound like a Model T. I built FOR NEXT LOOP delays into some of the programs; but even when these data are spread over all three drives, some programs sound "Chunky -- with nuts" -- and bolts?

Do you have any thoughts on the matter?

F. L. Eskholme
S W E D U M
Data Processing
Nutley, NJ

(Radio Shack, in all their wisdom produced two mechanically different drives for the Model I. The first, from Shugart, produces noise by loading and unloading the head. This noise can be silenced by maintaining the head in a loaded condition, but with the size of the 5¼" drives, the solenoid will eventually overheat and cause other problems. The second drives, from Tandon, produce noise when moving from track to track and cannot be silenced. The head is never unloaded.

The Model II uses a Shugart drive in 0 and CDC drives in 1, 2, and 3. Noise is caused by head loading and unloading. However, only one drive is loaded at any one time, so when different drives are selected you will hear a combination of heads loading and *unloading. We know of no solution to this problem as it seems to be common to this type of drive mechanism regardless of the manufacturer. Ed)*

I just received my copy of the March/April issue of 80-U.S. It was badly tattered and several pages were actually missing.

Could you please send me another copy of that issue, this time in an envelope or other wrapper. I permanently bind each year's issues and would like to have a permanent — and readable — set.

For the future, please consider mailing your magazine in a wrapper of some kind. You have a fine publication — you should give it the respect it deserves.

James H. Fox
Afton, MN

(It seems we got a lot more complaints this issue than most. The problem lies with the postal service as the magazines leave bundled and in mint condition from Tacoma, Washington. Complaints should be directed to the post office, your representative or senator, and possibly to President Reagan, suggesting postal employees paychecks be delivered through the mail attached to copies of 80-U.S.

The possibility of wrapping the magazine would result in increased costs which translates directly into high subscription rates. We'd like to hold the price there as this publication has never had an increase However, we are open to suggestions- so put them in writing and we'll continue to explore the possibilities. Ed)

Just read your Jan/Feb issue and wanted to correct a couple of statements made in Bill Vick's article "A Pocket Computer Application." He states the Pocket Computer "can store a program up to 999 lines long." This is not true. The PC stores each line number as two steps of memory, each command as one step of memory, and (ENTER) as one step of memory. Your shortest possible program line takes up 4 steps of memory and might look like this:

100:REM(ENTER)

Since we are limited to 1424 steps of memory, the maximum number of (shortest) program lines would be 356, not 999. However, the line numbers themselves can go up to 999.

Mr. Vick also states "each program step was entered in the abbreviated format to save memory." Since each command word is stored as one step of memory, abbreviating them does not save any memory. It does allow for faster input of your program, though.

Thanks to Mr. Vick for his fine application and my continuing thanks to 80-U.S. for your fine magazine.

Dick Stransky
Madison, WI

(And our thanks to Mr. Stransky for his corrections.

One additional thought... the Pocket Computer will store a command as one step no matter how *that command was entered. Ed)*

RE: NFL-PIX by James Talley

I certainly enjoyed using the above program for this past football season. Would you please let me know what is required for me to get an update of the program before the beginning of next year's football season. Thank you.

Peter J. Broullire III
Albuquerque, NM

On January 21, 1981, I bought one of your NFL PIX Programs, designed for 16K TRS-80 Level II. I don't have the disk drive but instructions say "CLOAD" works for cassette load. This I found to be true.

It seems likely that the program, in basic (I gather from the literature accompanying the cassette that the whole program is in Basic), was developed for a TRS-80 Model I Level II System.

Problem 1

I have a TRS-80 Model III, which was touted to be nothing more than an advanced, improved version of the Model I.

As I mentioned, the program CLOADs fine, *but,* when I run it, the title "page" shows up on the screen for about 10 seconds then I get an "OM ERROR" message in line (X). I check the amount of memory left in the computer by the "?MEM" command. It tells me I have 104 remaining.

Problem 2

I remove the program from the computer and reset...then check memory. It tells me "15314"...

I would like to know if you have come across Problem 1. If so, what is the modification in the "NFL PIX" program to let it "run" smoothly. Or, is it impossible to convert the program. Please let me know somthing about this problem. I am not an experienced programmer, but, whatever you tell me...I can get a clarification from the local Radio Shack Computer Center. By the way, in talking to Radio Shack (New Orleans) they told me that since this is not a Radio Shack program, they couldn't help much but, suggested that if you ran into

WHY IS THE MICROCONNECTION™ A MODEM AND MUCH MORE?

Because it can dial the phone and answer it too, transcribe transmissions, decode busses, run printers, receive/send radio transmissions, answer/originate, and direct-connect to the phone system. Take a look at all the software support. That's why it's a modem and much more.

MODEMS....

MICROCONNECTION
For Model I or III, decodes the TRS-80 buss and creates RS232 port. Eliminates need for expansion box and/or RS232 board. Provides serial printer output. Requires software written for Port 208. Requires adapter cable for Model III—**$249.00**

RS232CONNECTION
Works with virtually any computer or terminal having RS232 serial I/O port. Replaces obsolete acoustic coupled modems. Compatible with all TRS-80 terminal software. Requires RS232 cable—**$199.50**

COLORCONNECTION
Converts RS Color Computer into a timeshare terminal. Provides serial printer output. Does not require additional memory or extended basic—**$199.50**

AUTOCONNECTION
This option permits modems (except color) to answer the telephone and/or dial numbers automatically. Supplied with dialer demonstration program—**$79.95 option**

EUROPEANCONNECTION
This option modifies above units for European Standards—**$20.00 option**

MODEL III ADAPTER
Cable converts Model III 50 pin buss to 40 pins used by Model I peripherals—**$39.95**

RS232 CABLE
Four foot ribbon cable terminated with 25 pin male DB25 connectors—**$24.95**

...AND MUCH MORE:

AUTOMATED SMART TERMINAL PROGRAMS WRITTEN BY DICK BALCOM

All smart terminal programs feature manual or automatic upload/download, two user defined messages plus auto "who are you" interrogation, autoprompt message entry, transfers EDTASM and Orchestra 80 files formatting of line length, speed selection, upper lower case drivers, autorepeat key, "beep" function for keystroke entry, buffered print output with page and perforation skipover, compatible with SCRIPSIT and PENCIL, built-in text generator for those who do not have a word processor.

DISK

SMART80D
The LEADING smart terminal program for Model I and at a popular price. Requires one drive and 32-48K. Compatible with all major disk operating systems including double density—**$79.95**

SMART IIID
The FIRST smart terminal program for the Model III. Offers popular features of SMART80D plus several new features associated with the Model III—**$99.95**

SMART/BAS
A simple basic smart terminal disk program that can be user modified. Adaptable for cassette based Level II systems—**$19.95**

CASSETTE

SMART80C
For Model I, additional features include ASCII-token-ASCII conversion for transferring basic programs. For Level II, 16-48K systems. Specify if for MICROCONNECTION or RS232 version—**$79.95**

SMART80E
The first and only smart terminal program for the Exatron "Stringy Floppy", similar to SMART80C—**$79.95**

SMART IIIC
The first cassette smart terminal program for the Model III. All the power of SMART80C plus numerous other features. Specify if for MICROCONNECTION or RS232 version—**$99.95**

HOST PROGRAMS

MESSAGE 80 *By Richard Taylor*
Now you can operate your own bulletin board, either public or private, with MESSAGE 80 and the AUTOCONNECTION. Requires TRS-80, 2 disk drives, 48K and RS232CONNECTION (with the AUTOCONNECTION option)—**$149.95**

MIMI-MSG
Even if you only have a Level II, 16K keyboard, you can operate your own bulletin board! Requires TRS-80 keyboard and MICROCONNECTION with AUTOCONNECTION option)—**$49.95**

SUPERHOST
Allows other computer operators complete access to your system after password validation. Numerous features—**$29.95**

MasterCard VISA

To order your MICROCONNECTION or for more information, write or phone:

BT Enterprises
171 Hawkins Rd., Centereach, NY 11720
(516)981-8568 (800)645-6038
Computer 300/1200 (516)588-5836

Level IV Products Inc.
32238 Schoolcraft, Suite F4, Livonia, MI 48154
1(313)525-6200 (Mich only)
1(800)521-3305 (Elsewhere)

Microperipheral Corp.
2643 151st Pl. N.E., Redmond, WA 98052
(206)881-7544

the microperipheral corporation

Circle 3

difficulty with the problem, that you might contact the Radio Shack Computer dealer there in Tacoma. They, possibly, could help.

If Problem 1 is readily solved by changing lines, I would appreciate a printout of just which lines have to be changed and what the exact changes are. Then, I can edit "NFL PIX" and save the corrected version on my own cassette.

While you're researching Problem 1, could you please tell me in broad terms where all the .686K remaining memory goes when my machine is empty.

Is there a posssibility I could gain some of this back by entering a certain command?

Your cooperation in this matter will be greatly appreciated.

Larry W. Counce
New Orleans, LA

(These two letters are typical of those received here since we came out with NFL-PIX by James Talley. It will be up to Mr. Talley to (1) issue a new program or (2) provide the changes necessary for the 1981-82 season

The Model III, while being a new machine with many nice features, does have some drawbacks. As the literature accompanying the machine clearly states, there are 258 fewer bytes of memory available for user programs. Mr. Talley's program was stripped to fit the 16K tape format of the Model I and there is a good possibility that it will not be able to fit the 16K Model III machine. Since memory expansion in this unit is relatively inexpensive, I would think that this might be the simplest solution to Mr. Counce's problem.

As to recovering overhead space, (which is what Mr. Counce is talking about in his second problem) this is absolutely required by the operating system and cannot be recovered. If you think this may be a rip-off consider that most computer companies state memory size without any reference to overhead required by the system. If Radio Shack followed this practice, the (current) Model III 16K would be rated as a 32K computer! Ed)

Have you ever spent hours typing in a program from a magazine only to find more bugs than you can handle? Case in point: TRS-80 Graphics Editor, page 88 Jan/Feb issue. Every time I try to run it I get an FC error on line 180. Since I know next to nothing about machine language, I'm stumped. In fact, since getting your magazine I've only had one program that you printed that would run. The programs you print sound great if only one could use them. I enjoy getting 80-U.S. and I find many useful facts in it. One improvement you could make is to not assume prior knowledge in all your projects.

Richard Dysert
Columbus Grove, OH

(With very few exceptions, all program listings published in these pages are photographic reproductions of computer produced listings from the actual running program. It is very important that you recheck your typing against the listings in the magazine. The case Mr. Dysert brings up represents one of the most difficult bugs to trace in a Basic program. The error here is typically caused by problems in the DATA statements, rather than in the READ and POKE statements where the operation crashed. An FC error will be created by any *amount over 255 being poked into memory. Ed)*

I am too busy doing everything I learn in your #1 magazine to write long letters, but must thank everyone associated with your organization for the service you provide. Whether they realize it or not, Tandy owes you a lot for what a magazine like yours can do for hardware sales and to those of us who wish to learn as well as use a brand of computer. Regarding software, I will not buy unless I try or read a review of software. If truly authors will vanish (*quit?*) from copying, likewise buyers who are burnt will buy no more.

Malcom King
Ukiah, CA

Your magazine is a delight for a TRS-80 owner. One question: do your experts know of a piece of equipment which would transform 120 volt *50 cycle* current to 120 volt *60 cycle* current?

Joe Williams
American Embassy (Kinshasa)
APO New York, NY 09662

(The only method that we know of is to first convert to DC with a power supply and then back to 60 AC with an inverter.

Any of you who are aware of a commercial unit may want to drop Mr. Williams a note at his address Ed)

Help! Help! Does anyone out there know how to make a TRS-80 Model II work on ham teletype? I am an avid RTTY enthusiast who has recently upgraded from Model I to Model II and alas, I can not get back on radio teletype as I can not figure out how to make the RS-232 go down to 45 baud or how to write the program. I would be interested in offering your readers a handsome reward if I have written to all the current manufacturers of devices and programs for Model I RTTY and none of them are interested in helping us hapless Model II owners.

How about it guys?

Joel M. Greenfield
WA2KZD
156 Groton Place
West Hempstead, NY 11552

(Anybody interested in the challenge? Ed)

I have recently purchased a 16K Model III with an RS-232 board installed. I have had some minor previous experience with Basic, but know absolutely nothing about Assembly / Machine language programming, which puts me in a rather awkward position. I am interested in using my system for scientific applications, not in programming for the sake of programming; however, there is presently almost no software available for the Model III. I greatly appreciate your excellent efforts in the Jan/Feb 80-U.S. to specifically point out programs listed there which were applicable to the Model III, something which none of your competitors have as yet done. Some additional steps that would be extremely useful to myself and all future TRS-80 owners (as Mod I is discontinued) would be the following:

1) Ask contributors to design programs that can be used on both Mod I and Mod III whenever possible (both Basic and Z80 language programs), or explain what changes are required to use the program on the Mod III.

2) Present an article listing exactly what differences there are between allowable Basic for the two models, esp. PEEK/POKE addresses, etc. (never having used Mod I, I do not know how to modify published programs).

3) Present an article explaining differences in Z80 language and addresses between the two models.

4) Present articles explaining how to convert previously published programs (in both Basic and Z80) for the Mod I into suitable programs for Mod III.

Finally, I am interested in using my TRS80 mostly as a terminal to a UNIX system on a PDP 11/70 and a VAX, but no intelligent terminal programs presently exist for the Model III. I would be very interested in seeing a terminal program that can also transfer files (Basic and text files for editing) between a UNIX system and the Mod III (memory, tape, and disk).

Again, thank you for your service to new Model III owners!

Richard K. Wallace
Santa Cruz, CA

(For an intelligent terminal program, take a look in this issue's new products. In the meantime, potential authors: take note of this man's suggestions. Model III users should be aware of the 258 fewer bytes in their machine, but aside from this, Basic between the Model I and III is almost entirely the same. ROM routines present an entirely different matter as some (not all) have been changed. Suggestion number 4 is an area we usually do not get into, but watch for a "Best of..." book in the future. We'll check all published programs in that publication. Ed)

ACCEL/ACCEL2 SPEEDUPS

TRS-80 Model I BASIC Compilers

Table below shows the BASIC subset translated by **ACCEL** and **ACCEL2** to machine code. Figures represent the minimum expected ratio of execution times, compiler to interpreter. All other BASIC statements and functions run at interpreter speed after compilation.

	INTEGER	SINGLE	DOUBLE	STRING
Assignment (LET)	**115**	3.3	3.4	7.6
Array Reference (1-dim)	35	78	66	34.5
AND or OR	**41**	2.5	2.0	
Compare (<,etc)	**30**	1.6	1.4	4.2
Add, Subtract, Concat	**47**	2.0	1.5	4.9
Multiply (*)	3.3	2.0	1.5	
Divide (/)	2.0	2.0	1.02	
Reference to a constant	**69**	65	54	2.1
FOR with NEXT	**15**			
POKE	**82**	4.6	3.6	
SET or RESET	**6.7**	3.1	2.6	
IF THEN ELSE	**11.1**	3.0	2.3	7.6
ON expression GOTO	**15.8**	3.2	2.8	
Functions				
VARPTR	**33**	47	47	44
USR	**11.2**	3.7	2.8	
POINT	**6.9**	3.0	2.5	
PEEK	**52**	4.4	3.5	
LEN				43
MID$				4.1
LEFT$				3.0
RIGHT$				2.8
CHR$				4.7
ASC				30
CVI				28
Flow of Control				
GOSUB with RETURN	**137**			
GOTO	**204**			
All other BASIC statements and functions	**1.0**	**1.0**	**1.0**	**1.0**

ACCEL: For 16K TRS-80 Model I. Compiles boldface subset in INTEGER variable type. Compile-time size 2816 bytes, run-time size 256 bytes. Trade up later to ACCEL2 for the price difference.
$44.95 + .75 shipping

ACCEL2: For 32K TRS-80 Model I. Compiles subset in all variable types. Compile-time size 5120 bytes, run-time size 1024 bytes. Local and Global options help control growth of the compiled code. Output save to Disk, Stringy/Floppy. TRSDOS and most vendor DOSs supported. Use ACCEL2 during BASIC program development for easy huge speedups. $88.95 + 1.50 shipping

TSAVE: Writes ACCEL/ACCEL2 compiler output to independent SYSTEM tape. $9.95

Developed in Britain
by Southern Software

Circle 4

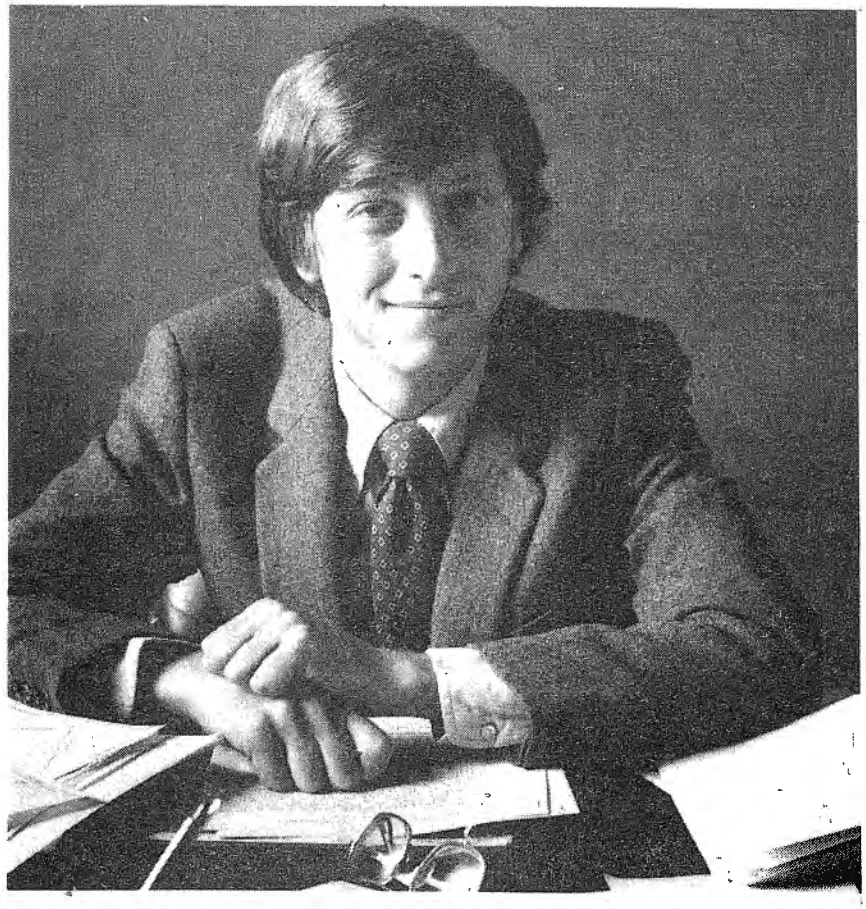
ALLEN GELDER SOFTWARE
Box 11721 Main Post Office
San Francisco, CA 94101

Mastercharge/Visa
Ca add 6%

TRS-80, TRSDOS tm Radio Shack
Stringy/Floppy tm exatron, inc.

80-U.S. Interviews Bill Gates of Microsoft

Bellevue, Washington



It's nearly impossible to work with the TRS-80 without coming into contact with software written by Microsoft of Bellevue, Washington. They have supplied the TRS-80's Basic interpreter for the Model I, II and III as well as a host of supporting software of consistently high quality. Though there are a few gripes about how Microsoft does business, most people who work with their products have praise for the high quality of the software and responsiveness of the organization.

Microsoft was founded in 1974 to supply software for the newly developed Microcomputer marketplace . Since that time they have grown into a major force in software development. Among their products are the Basic Interpreter, Basic compiler, FORTRAN, COBOL, PASCAL, Data Base Manager, Word Processor and a new line of consumer products.

Since the beginning, Bill Gates has led this organization of software superstars. He is responsible for many of their best products and continues a program of software research and development which refines and improves the products which have made Microsoft the leader in their field.

Bill attended Harvard University, where he was in an accelerated graduate program. He worked at the TRW Corporation where he feels that the open atmosphere helped him learn much of what he knows about software.

When the Altair computer was first introduced, Bill moved to Albequerque New Mexico. This period saw the introduction of Microsoft Basic and the development of the TRS-80 Model I, Level II ROM packages.

In 1978 it was decided to move Microsoft to a major metropolitan area. After looking around, they made their move to Bellevue, Washington in early 1979.

Like most of the other people we have met at Microsoft, Bill Gates turned out to be knowledgeable and easy to talk to. At Microsoft, Bill is the last resort for the final, authoritative answer to any technical question. Bill remains apparently unaffected by the fantastic success of the company. He loves to talk about computers and computer software.

Most recently he completed the work on the Radio Shack color computer. He is most excited about the Extended Color Basic package he wrote.

We met with Bill in his office, where we talked about the past, present and future of Microsoft.

80-U.S.: What do you think about the various people who have adapted your Basic, changed things about it, and are now marketing it?

Bill: One of the things we are trying to promote is a standard where people can interchange programs from one system to the next. So obviously, it takes away from that a great deal when there are specific things that are in one system implementation and not in others. A good example of that would be graphics. There is no standard for how you implement graphics in hardware and there are so many different characteristics, different resolution, and just different ways of doing the stuff that it's tough to be compatible today. Likewise, people's programs are often dependent on the size of the display. Radio Shack considered going to an 80 character display when they designed the Model III. But they felt that would create too much incompatibility, so they didn't. And obviously that same

incompatibility would exist when someone is taking the Basic ROM up to an 80 column system. The degree of incompatibility is less than people would expect. It's really isolated to the I/O and graphics areas. When we did the Basic, we decided that there would be several levels. As you moved from one level to the next, it would always be a superset of the previous level. We had 4K, 8K, Extended and Disk: four levels. When I had to do the Radio Shack ROM, I had 12K, more than enough for 8K but not enough for extended. It's a hybrid, in betweeen extended and 8K. For example, there are no user defined functions or short error messages; two features typically in extended. But there are PRINT USING and Double Precision variables, which aren't found in 8K Basic. The additional data types were one of the key things we wanted to get.So we took that as given and pared it down to cram it into the 12K. We try to get people not to make changes and we have been very successful as far as the language itself goes. We have even participated in doing specific graphics stuff. But it's going to be a long time before there are any real standards there.

80-U.S.: I can see that. How about some of these inside Microsoft Basic books? Have you looked at any of them?

Bill: One or two of them. It's illegal for anyone to take material out of the ROM and publish it based on our software. I think what one of them did was publish comments and say that if you owned a ROM, somehow then you could fill in. That's legitimate. Anybody who actually gives the disassembled listing, ... when I hear about that I might take action. In a way, it's a legitimate thing that people want to know how the stuff works.

One of the most popular cassettes of all time for the TRS-80 is the EditorAssembler. I never expected it would be. I thought that it would be compared with Basic, where you don't have to bang bits. Well, an impressively high percentage want to understand what is going on in the ROM and in assembly language after they have used it for a while. So a natural place to look is to say how does Basic work? Or, the thing that people often say is they want to call routines in the ROM. People have a little overblown view of how easy that is to do. The ROM wasn't designed as a bunch of subroutines, with the exception of the first 2K, which we did as the I/O stuff. It's extremely messy to use anything but those I/O things because on any error condition there are branches off to other places. In the case of the color computer we expect to come out with something that won't be a complete dump of the ROM. But we will have portions of it listed which is legitimate for us. We will talk about its overall structure and about advanced techniques using the thing. This should sort of head off all the decoding and insider-type books. We will see if this helps, and possibly distribute that through Radio Shack.

Actually, in a way I'm impressed that people have done as well as they have. The code is reasonably tricky and very compact. We didn't have a chance to totally rework the thing for the Z-80 because we only had about four weeks from start to finish on that entire project. The ROM was really difficult. Also, it was the first time anyone had ever done a 'hook' scheme where we allowed additional disk code to come in as a RAM load. And we were never able to test that. There were no disks working at the time. So we put the hooks in and used our judgement that those things would work out properly. As it turns out, we and a few other people have used those hooks for other types of extensions. And we have gone beyond that. We have much better hook schemes that we have used in other Z80 products in some of the other microprocessors. And they even add reserved words and don't just work within the fixed set that was there.

80-U.S.: Are you considering publishing any more inside information other than about the color computer?

Bill: Well, as far as the Model I goes, there are four or five books which go into the stuff. There is so much crazy terminology. Disk Basic was originally going to be called Level III Basic. That's why it says L3 ERROR. And then we went and called our product Level III Basic. And Radio Shack, on the Model III, call that Level III Basic. But I don't think that for Model I or Model III Basic we will do anything. I don't endorse any of the ones that are out there nor do I damn them unless they have used our material.

80-U.S.: How about the Model II Basic?

Bill: Well, the idea there was that the Model II people were probably more businessmen. The spare time to be curious and really learn about it didn't come into it as much. That's why we didn't put PEEK and POKE in; Radio Shack told us not to. The next thing we knew, some guy comes out with patches for PEEK and POKE. I doubt the audience there is large enough to justify that type of book on the Model II Basic in RAM.

As far as the Model I and III go, we did come out with the advanced Editor-Assembler. People can use that to probe into the ROM and play around because it has a very nice symbolic decoder as part of ZBUG and that's probably enough. As far as the color computer goes, we will probably come out with the same package, but we'll also have a text that talks about the inside.

With the color computer there's more to talk about. You can talk about high-res graphics and how you do 3D graphics and color mixing and PAINT and what the PAINT algorithm is and how you do sound modulation and how to use joy sticks to do really interesting things. There's a lot of complexity that goes beyond that. The things the people talk about on the Model I are why the cassette routines aren't reliable, or how to add keyboard debounce, such things aren't nearly as interesting.

80-U.S.: Are there any plans for the TRS-80 in the future that you can talk about?

Bill: Of course, we have our Consumer Products division that sells products directly for the TRS-80. And we have a lot of stuff that will be coming out there. We have a modeling program, some forms handling stuff, the disk version of the Editor-Assembler, the disk version of Level III Basic, and then we have a totally new release of the Basic compiler that is a significant improvement over the previous version.

80-U.S.: Is it one with a run-time package?

Bill: Yes, so you don't use nearly as much memory or disk space.

You know, it surprises us how few software vendors support the Model I. We are going to continue to do it. The challenge to us is that since we can't get our stuff into the stores that sell the computers, the only people we can sell to independently are the ones who stop by computer stores or who buy mail order. Most of it today is really mail order. Radio Shack's controlled distribution makes it tough for

(Continued on Page 12)

SUPERSCRIPT

FOR TRS-80*

By Richard Wilkes

Now you can easily print
BOLDFACE, underlined text,
slashed Ø's, special characters,
superscripts & subscripts (A^c X $\log_e[A_1-A_n]$) and more!

With this enhancement to Radio Shack's Model I Scripsit,* you will turn a good word processing system into a great one. SuperScript adds many of the features that Fort Worth left out, including:

Custom print drivers for:
Diablo (serial or parallel)
NEC Spinwriter (serial or parallel)
RS Line Printer IV*/Centronics 737
RS Daisy II*
Standard serial and parallel printers
User-defined custom drivers (serial or parallel)

You can call up the disk directory or kill files without leaving Scripsit—and without losing your text! Pause the printout to insert text from your keyboard or change type wheels, then resume printing where you left off.

Using any printer with backspace capability, you can underline text and produce computer-type slashed zeros. And on Diablo, Daisy II and NEC printers, you can superscript, subscript, underline, print boldface and select 10- or 12-pitch. With the Line Printer IV/737, you can underline, print elongated boldface, select 10 or 16.5 pitch, and use proportional spacing (unjustified).

The keyboard driver is now modified to correct for repeating key hangups. You may specify spacing requirements to eliminate awkward spacing of critical text. And, with SuperScript, you can now enter special characters (brackets, braces, etc.) that are not found on the TRS-80* keyboard.

All these capabilities, and more, are available when you add SuperScript to your Model I Scripsit program. Available for just $29.95 on disk, including easy instructions for patching to Scripsit and an enhanced lowercase driver.

SYSTEM SAVERS

By Tom Stibolt

If you ever type "SYSTEM" on your TRS-80,* this two-program package will make life easier for you.

One of the programs, FLEXL, lets you make backup copies of any system format tape. Using your own recorder usually means easier loading than with machine-duplicated original tapes, and you will be able to store your original safely away. Copies made using FLEXL display the filename of each program as it loads, making file searches easier.

Disk drive owners can use TDISK to save any system format tape onto disk. "Editor/Assembler", "Air Raid" and other programs cannot normally be loaded to disk, but with TDISK, they can. It will even load non-contiguous tapes. Why put up with slow tape loading? TDISK files will load from disk in seconds.

Get this two-program package now for only $14.95. Just one of Acorn's fine utility programs.

* trademark of Tandy Corp

Acorn
Software Products, Inc.

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

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

DEALER INQUIRIES INVITED

Circle 5

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

Epson.

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

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

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

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

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

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

But at this price, you'd better hurry.

EPSON
EPSON AMERICA, INC.

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

(from page 9)

someone like us. So we have a choice, we can either OEM everything to them or we can try to build an independent image and marketing channel. We have so many things we want to do, so many products, that we need to have the independent image as well as the OEM market. We OEM'd the FORTRAN and we are OEMing some nice new stuff for Model III, but not the Basic compiler and the really key consumer products stuff.

80-U.S.: Do you have any plans for Model III software?

Bill: Not really. Anything we do for the Model III we'll also come up with a Model I version. They are so compatible, you might as well address both markets. There are 200,000 Model I people out there. With the way we market and the recognition factor we have, it's hard for us not to get at least 1% market penetration and that's a couple thousand sales, so we continue to work them both even-handedly. We've moved a little slow to bring that stuff up on the Model III because frankly we haven't seen a market. We are just beginning to see a little demand.

80-U.S.: Why didn't Radio Shack go with Microsoft's Basic compiler for the Model II?

Bill: We have a compiler that is compatible with the interpreter and runs dramatically faster because it is a true compiler. It fits into all the other development tools including common relocatable format so it can call things from FORTRAN. We felt that we were the obvious solution and perhaps in some ways we overplayed that. Our relationship with Radio Shack has had its ups and downs, but they dealt with us very honestly and continue to work with us on many things. In that one case, we think they made a bad mistake.

The thing that's great is to have this synergy of being able to debug under the interpreter and then go to compiling. The problem with compilers is that it's not at all interactive. In our compiler and Radio Shack's compiler, the amount of time it takes to compile something and get it ready is minutes. For some small typing mistake that's outrageous. And so that's why we have the combined approach. In the case of the Model II, we'll be coming out with an enhanced interpreter that has all the features of the compiler. In the Model I and III the compiler has gotten a little ahead of the interpreter since that was burned into ROM a little over three years ago and so you have to be careful to stay within the interpretive set. There are a few things such as dynamic dimensioning which are not supported in the compiler which will be out in future versions of it.

80-U.S.: Have you upgraded or tested any of your software on Radio Shack's new DOS release for the Model II?

Bill: On the DOS 2.0? We had to go and convert all of our files as part of our OEM contract with Radio Shack and we had to run FORTRAN under it. It seems OK.

The DOS's have had an interesting history. The first was written by the guy who sells VTOS and that was a stormy relationship. The in-house group did the Model II TRSDOS, and of course the size of the Model II TRSDOS is different, but they put in some improvements and I guess Model III TRSDOS is a merger of that. I really haven't looked at it at all.

80-U.S.: From the standpoint of the average TRS-80 user, your manuals have improved considerably, particularly on the Model I.

Bill: We're trying to improve the manuals. The company started out with pretty much an OEM orientation. What we were addressing was a low end of the data processing market where people were using micros at low cost to do what they had done previously on mini's and large machines. And in some ways our software and our documentation was adapted to that mode.

Now what we are addressing is a whole new class of users who aren't perhaps as sophisticated. So we have had to really beef up our tech writing. We have four times as many tech writers as we did. Some people like Apple have set pretty reasonable precedents for good documentation and so that is really our focus. We have a couple of products where our investment in documentation is as great as in the software itself, which is a new thing.

80-U.S.: Occasionally, I've seen complaints about Microsoft's user support or their software. An example which comes to mind is the M80 assembler, where one gentleman complained about the lack of real Macro capability as he called it, because it could handle Macros but it couldn't handle a Macro libra[illegible] How do you feel about those kinds of comments?

Bill: Software, if it did all things for all people would be arbitrarily big and arbitrarily slow. Obviously you make tradeoffs. That's what we're known for doing real well. Various users will complain about how we have made those tradeoffs, which is to be expected. There is nothing wrong with healthy criticism. We've used that feedback many times and factored it into our new product plans. Like our loader or some of the new compiler features, which are very much based on feedback. You know, if someone comes down on us real hard, I can get upset because I think we have made an excellent tradeoff as to what should be in a macro assembler. It's extremely powerful. I doubt that there are many people out there who get near to its full capability. And it's the same thing with the loader. The loader is confusing to some people because it was designed to support a pretty sophisticated operation where you can locate things at arbitrary locations in memory. It's just a tradeoff we made. Maybe in that case we went too far towards sophistication. But if you put out as much software as we do you are not going to escape criticism. We have one or two things which I'm surprised there hasn't been more comment on.

80-U.S.: How much business do you do with Radio Shack? How much of a percentage of your operation is it?

Bill: Well, it's a classic tradeoff for an OEM software house like us. Are you willing to involve yourself in royalty arrangements with manufacturers or do you work on a fixed fee? On the initial Level II Basic we worked on a fixed fee. We have an on-going yearly fee relative to Model III Basic, it's strictly bounded. Some of the work we have done on the color computer is more royalty oriented and so we are super enthusiastic about it. We think the color computer is going to be incredibly successful so that, if I'm right about that, Radio Shack could be as much as 10 or 15 percent of our income.

One thing for most people to realize is that Microsoft does most of its work at a higher end of the computer spectrum. Although we are best known for our work with Apple and Radio Shack, Texas Instruments and people like that, most of our business is with NCR, ICL, XEROX and companies which are not down at the

low end. The thing about those companies is that they need more software for their products. You know when Zenith comes in, they want Basic, FORTRAN, COBOL, Basic Compiler, PASCAL and SORT - the whole bit. So that's a lot bigger piece of business and there are many of those manufacturers around. The low end tends to be dominated by a few manufacturers. Because of our Consumer Products Division, the low end is super important to us though. We have really done quite well selling to a user base of 100 to 200 thousand systems. When you think of what that is going to be like when there is a base of 500,000 systems, it's really neat. The key thing is to build a reputation in that market and as it grows we can capture more and more volume.

80-U.S.: It makes a lot of sense.

Bill: So, even from an internal point of view, Radio Shack is a very good customer. But if you also weigh in the consumer products things, you might say that they are our best customer.

80-U.S.: How about these TRS-80 imports like the PMC80 which advertises Microsoft Basic? Is that basically the same ROM that Radio Shack uses?

Bill: It is virtually the same ROM, and they have a license to Microsoft Basic.

I don't know if they are selling a lot of these. The hardware looks reasonable. The thing Radio Shack really proved about the computer business is that distribution is critical. So what if someone has an equivalent machine at an equivalent price. They are not going to hurt Radio Shack that badly. No one has 7000 stores, or whatever the number is, to place the product. As long as Radio Shack has a reasonable product, they are going to be a leader in this business.

80-U.S.: So as long as you can put out good software for the system, you're going to go right along.

Bill: Yes. We are certainly available to someone who has that kind of distribution. There are two ways we work, we like to make a good economic return, but we also like to be involved in bringing computers to more and more people. We get a lot of enjoyment out of the success of the Model I with Level II Basic. The initial prediction was that 10 to 20 percent of the people would upgrade (from Level I to Level II) and it makes me feel good that the number is almost the inverse of that. It's 80 to 90 percent. I hope to see the same thing on the color computer. Our participation is heavily tied to the success of the extension ROM.

80-U.S.: Did you do the actual Basic in the color computer?

Bill: We did the 8K ROM. But that is not actually subject to a royalty. It's the extension ROM. They are married. They fit together in a very strong way. It's another classic situation. There was limited memory, limited time, a lot of new features that needed to go in and so we had to work really hard to get that done. We are super pleased about it.

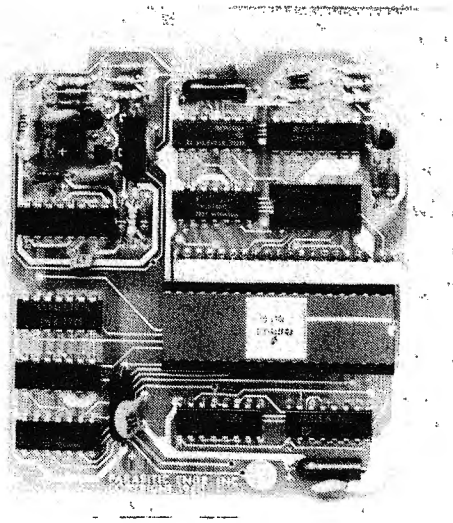
80-U.S.: From everything I've heard about it, it sounds like it is going to be a tremendous machine.

Bill: The only real threat to it is if Commodore can get their act together. It is possible that they could be a serious competitor. I'm interested to see if Commodore follows through. ●

NEW PRODUCTS

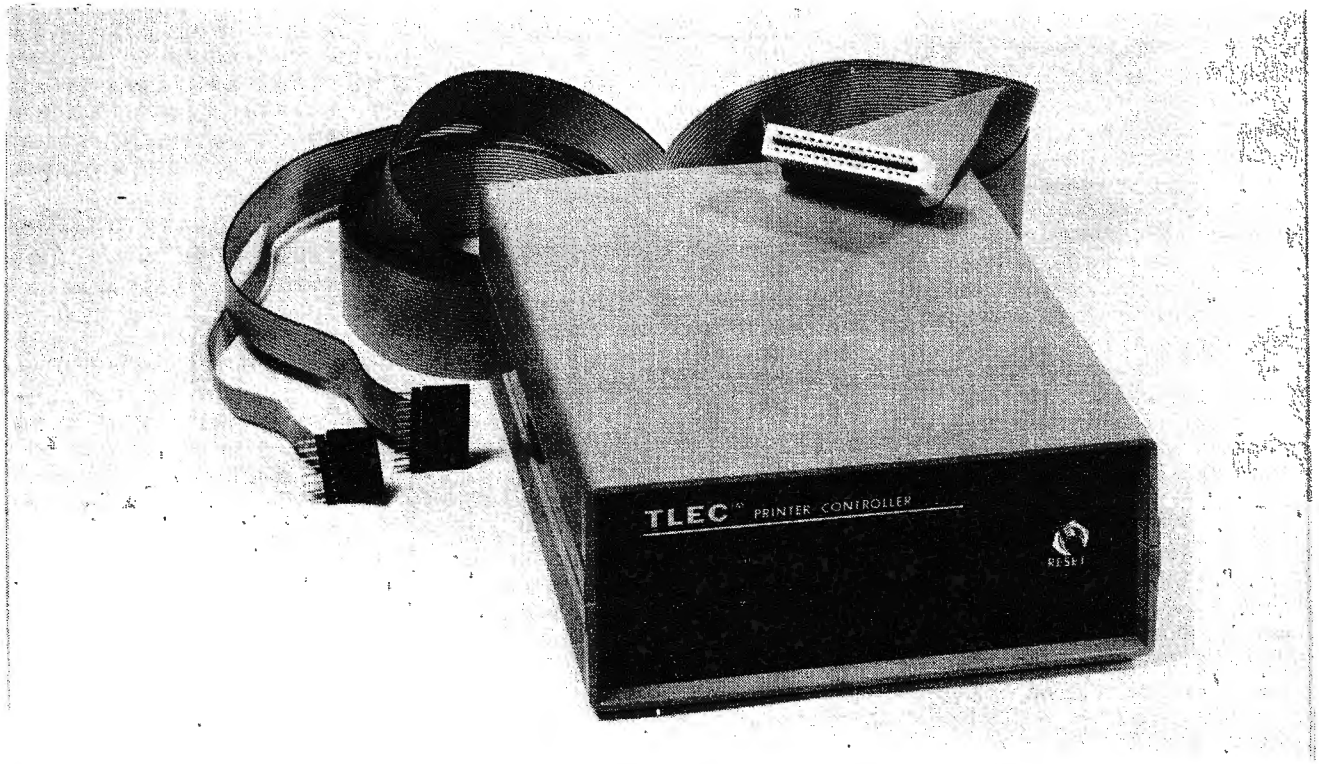
IBM Interface

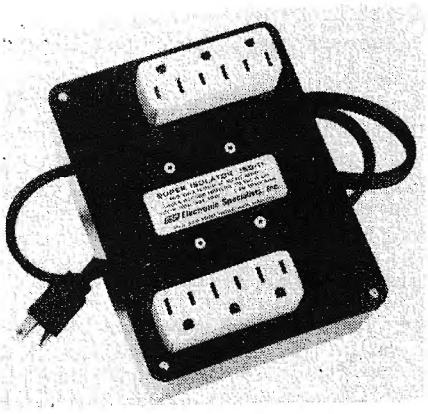
TLEC, is the name given to WEB International's new intelligent printer controller and formatter for the new IBM Electronic typewriter Models 50, 60 and 75. It is Z-80 controlled with up to 4K printer buffer memory. A preprogrammed ROM eliminates the need for any printer driver software. The Mark I at $399.95 provides full parallel handshaking along with many features. The Mark II at $499.95 adds proportional spacing capability with justification to the Model 60 and 75. The Mark III at $499.95 adds bi-directional printing and many other features including access to the Model 60 and 75 memories. Inquiries should be directed to WEB International, PO Box 96, Corona Del Mar, CA 92625 (714) 494-2869



Data Separator

Parasitic Engineering Inc has announced their data separator for the TRS-80 Model I. It eliminates disk errors, including CRC, TRACK LOCKED OUT and Disk I/O ERROR. Originally designed for their high performance 8" Maxi-Drive systems, the 5" Data Separator sets a new standard for reliable 5" disk operation on the Model I. The unit plugs inside your expansion interface. There are no additional wires and no cut traces or other permanent changes to your TRS-80. It is compatible with all TRS-80 software. The 5" Data Separator is upward compatible with 8" Maxi-Disk drives, and can be easily and economically upgraded at any time. It costs $250 and is available immediately. Contact Parasitic Engineering Inc, 1101 Ninth Ave., Oakland, CA 94606 (415) 839-2636





Super Isolator

Severe AC power line spikes, surges and hash are prevalent in many Micro-Processor installations. Program execution is plagued with unexplained crashes, memory loss or other glitches. Disks, printer and processor often interact, aggravating the problem. Electronic Specialists' recently announced Model ISO-11 is designed to curb these severe electrical problems. Complementing the popular Super-Isolator line, the Model ISO-11 features two individually dual-Pi filtered AC socket banks (6 sockets total). Heavy-duty spike/surge suppression is incorporated in the design. Equipment interactions are eliminated and disruptive/damaging line spikes and hash are controlled. The Model ISO-11 Super Isolator controls power line spikes and hash while providing interaction-free microprocessor operation. $94.95 Electronic Specialists, Inc 171 South Main St., Natick, MA 01760 (617) 655-1532

Racet Computes Catalog

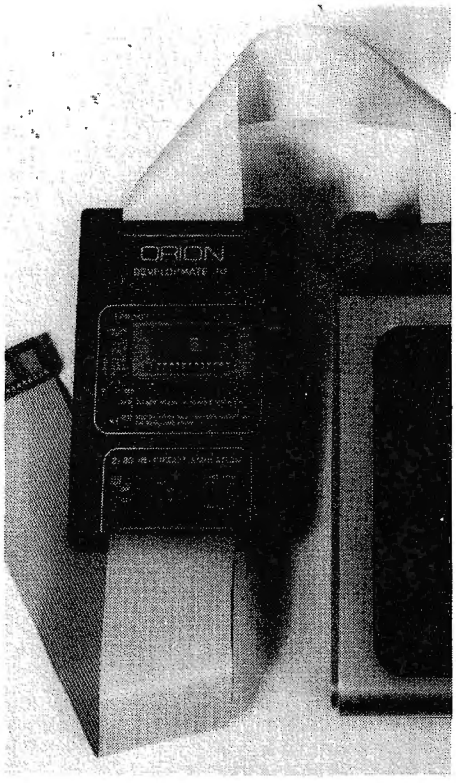
Racet Computes has issued a new spring catalog covering new prices and products for the TRS-80 Models I, II and III. They have also relocated to a new address in Orange, CA. Write for their catalog: Racet Computes 1330 N Glassell, Suite "M", Orange, CA 92667

New LDOS Operating System

LOBO Drives International announced a new generation in operating systems for the TRS-80. LDOS is a totally new and advanced operating system that will bridge the gap between TRS-80 microcomputers: compatibility with Model I is available now; compatibility with Model II and III will be ready soon. LDOS is fully device independent and supports 5" and 8", single or double sided disks. It supports hard disks. Among its many features you will find keyboard typeahead, auto-repeat, key-stroke multiply, wildcard and job log. For more information contact LOBO Drives International 354 South Fairview Ave., Goleta, CA 93117

Software Speedup Facilty

PROSOFT, the first computer company to offer full-function Word Processing for the Line Printer IV, has introduced a software speedup facility for most TRS-80 Model I Level II and disk Basic programs. This utility program is appropriately called "FASTER". It analyses executing Basic programs, then displays or prints information that enables the user to make a simple change (usually just one line). Resulting execution times are typically reduced by 10-15%. No hardware modifications are involved, and FASTER can be used with packages as well as with user-written programs. It is available from PROSOFT, Box 839, North Hollywood, CA 91603 (213) 764-3131.



Radio Shack's new Line Printer VII

The Line Printer VII is a low-cost, full-performance printer which works directly with the new Color Computer. It features a 4½" to 9½" adjustable tractor feed mechanism, high-density graphics, 80 or 40 upper and lower case characters per 8" line, and 30 characters per second print speed. It's an impact printer, using plain paper, and will produce the original plus two copies. Included is both parallel and serial interfaces. Available from Radio Shack stores and dealers, stock # 26-1167, $399.00, excluding cable.

Model III RSM Patch

REMarkable Software is proud to present a program to patch Small Systems Software's RSM for operation on the Radio Shack Model III microcomputer. The patch was written by James Haan and is marketed by REMarkable Software. This patch fills a need experienced by many new Model III users. Small Systems Software's RSM was chosen since it has withstood the test of time and has proven itself to be the "workhorse" of monitors. The RSM patch was designed to be compatible with the unpatched version and as such there are very few variations from the original. The patch operates tape at 500 and 1500 baud rates. Model III RSM Patch is available from REMarkable Software, PO Box 1192, Muskegon, MI 49443 and from many other software vendors.

EPROM Programmer/Emulator

An in-circuit emulation/EPROM programmer add-on can turn a personal computer into a development system. Microcomputer development systems are typically priced 2-5 times higher than equivalent "personal computer" systems because they are lower volume products in a much less competitive market. The Developmate 81 eliminates this gap by upgrading the popular TRS-80 personal computer into a full development system. Priced at only $329, the Developmate 81 adds both Z-80 in-circuit-emulation and EPROM/EEPROM programming capability making it possible to buy a complete diskette-based development system for about $2500, or an upgradable cassette-based system for under $1000! The Developmate 81 plugs into the expansion connector of the TRS-80 and includes both the PROM programmer and the in-circuit-emulator in a single compact box. The PROM programmer has a personality module which defines the voltages and connections to the PROM so that future devices with up to 28 pins can be accommodated. For further information call Tom Blakeslee (415) 851-1172 at Orion Instruments, 172 Otis Ave., Woodside, CA 94062

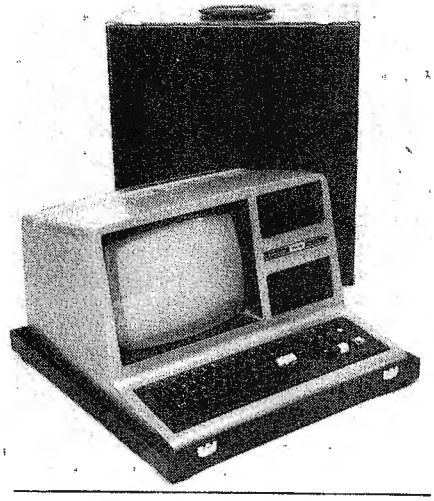
NEWBASIC for Model I

NEWBASIC, from Modular Software Associates adds new commands and utilities to the TRS-80 Model I Level II or Disk Basic. The disk Basic version is based on a unique linking loader concept. Many different versions of NEWBASIC may be created which can contain only a few or all of the over 30 commands available. Some of the disk Basic enhancements are a spooler, despooler, and a new directory command which may be issued while in Basic. Utilities included in both the disk and Level II version are: Quick Key Basic keyword entry, blinking block cursor, auto repeat and lower case display enable. Also

included are new graphics commands, the ability to input from or output to the RS232-C interface, a video to printer echo, GOTO and GOSUB to expressions or labels and more. Two powerful debugging aids are included in NEWBASIC. A new trace facility allows the display of program lines and variable or expression values as the Basic program executes. A locate command finds the occurances of strings or Basic keywords. The disk version is $29.95, while the Level II cassette based version is $19.95. The Level II tape may be returned and its price applied towards the disk version. Both are available for immediate delivery from Modular Software Associates, 3533 Prospect Ave., Glendale, CA 91214

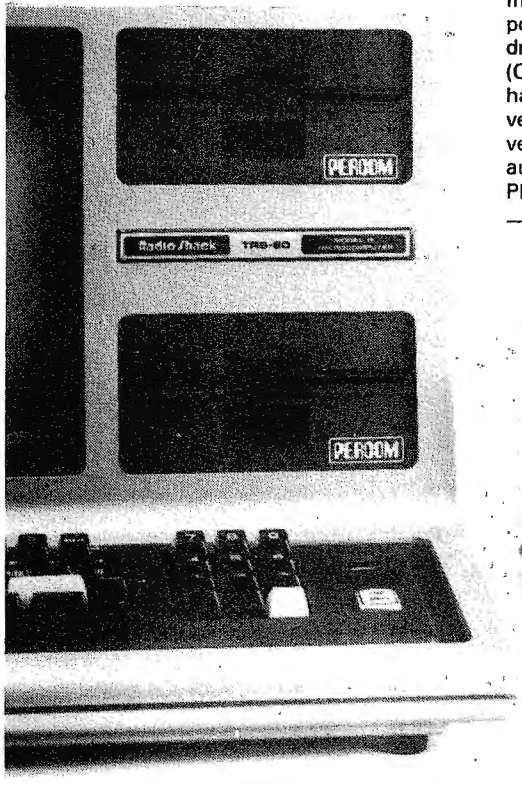
Smart Terminal for Mod III

The MicroPeripheral Corp has just announced SMARTIII, the first smart terminal program written for the TRS-80 Model III. The program is available on either a 1500 baud cassette or double density disk. SMARTIII permits transfer of Basic programs between the host computer and the cassette or disk storage device. The program permits off-line text preparation with Electric Pencil or Scripsit for on-line transmission. An additional program called FILE permits generation and storage of text, then transmission by SMARTIII for those who do not have word processors. SMARTIII also permits transfer of source code files. Downloading from FORUM 80 bulletin boards is accomplished automatically. The program was written for the Model III by Dick Balcom. It can be used with any RS-232 compatible modem such as the RS232CONNECTION, a direct connect telephone interface manufactured by The MicroPeripheral Corp. SMARTIII is supplied in a protective binder with extensive easy-to-use operating instructions and is priced at $99.50. For additional information contact The MicroPeripheral Corp, 2643 151st Place NE, Redmond, WA 98052 (206) 881-7544



Rate of Return Program

Soft Sell has announced its Internal Rate of Return program for real estate investors and sales people. Fashioned after the worksheets of the National Association of Realtors(R), the program calculates and produces an Annual Property Operating Data Worksheet as desired on a CRT or printer. In addition, capitalization rates, cash on cash rates, gross rent multipliers and equity return rates are supplied. The Internal Rate of Return is supplied on a floppy disk and is supported by Radio Shack Models I, II and III. The price of the program is $380. Soft Sell is comprised of real estate and computer professionals and is dedicated to the development of affordable real estate software. Contact Chris Ithomitis, Soft Sell, (509) 466-0503 or write in care of North 17512 Michael Road, Colbert, Washington 99005



Carrying Case for Color & Model III

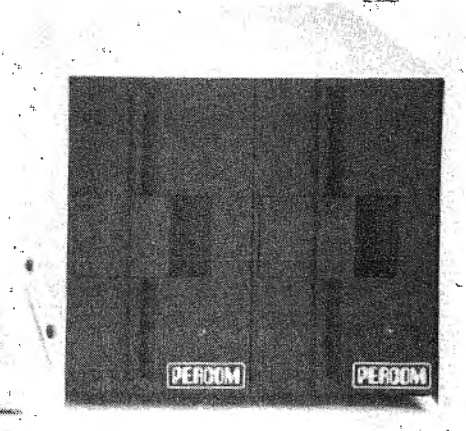
The Computer Case Co has announced the release of the RS204 for the TRS-80 Model III retailing for $129 and the RS205 for the Color Computer (or Videotex terminal) retailing for $99.00. The cases can be obtained through most computer stores or direct from the Computer Case Co, 5650 Indian Mound Court, Columbus, OH 43213 (614) 868-9464

Model III Disk Drives

VR Data, a leading manufacturer of innovative computer products has recently announced its Model III Disk Drive Assembly. Designed and manufactured by VR Data in its suburban Philadelphia plant, the assembly enables Model III users to add 40 or 80 track 5¼" disk drives either internally or externally. The internal drive assembly includes one mini-drive, power supply, controller and mounting hardware, and sells for $599, a 29% savings compared to Radio Shack. An additional internal drive is $265. Immediate delivery is available and all inquiries should call (800) 345-8102, (215) 461-5300 in PA or write VR Data Corp, 777 Henderson Blvd, Folcroft, PA 19032

PERCOM Model III Disk Drives

PERCOM is now producing mini-disk storage systems for the TRS-80 Model III. Systems may be ordered with either 40 or 80 track drives which are rated for both double and single density operation. The first two drives mount inside the computer. The complete internal drive system, which includes a four-drive controller, two drive power supply for the internally mounted drives, a double-density operating system (OS-80/III), and miscellaneous mounting hardware, sells for $749.95 in the 40 track version and $914.95 in the 80 track version. These drives are available from authorized PERCOM dealers, or direct from PERCOM (1-800-527-1592)



THE 80'S BELONG TO SNAPP!

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

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

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

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

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

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

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

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

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

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

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

DISK SPOOLING OPTION $50.00

ALL PRODUCTS NOW AVAILABLE FOR THE MODEL III

SNAPP, INC

8160 Corporate Park Dr.
Cincinnati, Ohio 45242

Call Toll Free
1 - 800 - 543-4628
Ohio residents
call collect (513) 891-4496

All products now available to run with TRSDOS 2.0.

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

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

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

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

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

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

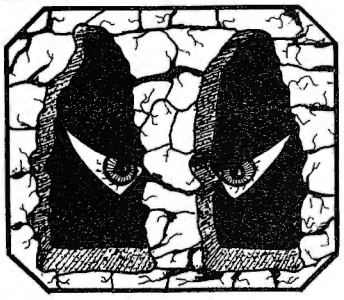
Electric Pencil is a trade mark of Michael Schrayer Software, Inc.

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

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

Circle 9

THE PROGRAM STORE



DEVIL'S PALACE

By Greg Hasset from Adventure World
Find the Devil's Palace somewhere in the deep, dark forest. You will have to use all your wits to enter the palace and conquer the evil which stalks the dismal corridors. This adventure (#10) is written in machine lan guage for super excitement and suspense.

Level II 16K...$14.95

ASYLUM

from Med Systems
You are sitting alone. It is 2 am. Your eyes are bloodshot. You peer into yor computer screen and suddenly scream, "I must be crazy!" If this has ever happened to you, or the men in white coats from "Deathmaze 5000" have hauled you away, it's time to try Asylum, the most ambitious 3-D graphics adventure yet offered by Med Systems.

Asylum features the 3-D perspective graphics that have made Deathmaze 5000 and Labyrinth bestsellers. You actually see where you are and where you are going!

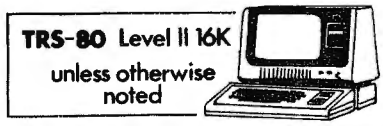
Asylum places you on a cot in in a small room. Periodically, a janitor lobs a hand-grenade through the window of your locked door. What you do next could mean survival or escape!

Level II 16K...$14.95

DRAGONQUEST

By Charles Forsythe from Programers Guild
It's a desperate race as you search for SMAEGOR, who has kidnapped the Princess of the Realm and holds her in a distant and unknown place. In a quest for honor and glory, you must search the land, seeking out the tools needed for the ultimate confrontation. Clues abound on The River Delta, in the abandoned Temple of the Goddess of the Blade -- everywhere! But WHERE is the Princess? Ordér this new machine lanuage adventure now. You may never find the Princess, but you'll have fun trying!

Level II 16K tape...$15.95 32K disk...$21.95



COSMIC FIGHTER

By Hogue & Konyu from Big Five
Terrific sound, graphics and unique challenges mark this new space game a winner! While fighting off the alien convoys, each more powerful than the last, you must keep track of your rocket fuel or risk explosion as you maneuver toward the mothership to refuel. Can you dock immediately, or is the station overrun by aliens? Find out by ordering Cosmic Fighter today.

Level II 16K tape...$14.95
32K disk version...$17.95

MISSILE ATTACK

By Philip Oliver from Adventure Int.
Closely patterned on the latest king of the arcade games, you must use your twin silos of ABM's to fend off barrage after barrage of enemy missiles that rain down toward your cities. As your skill increases so does the difficulty and speed of this ever machine lan guage arcade game! Watch the skies and may your aim be true! Missile Attack has sound and fast moving graphics.

Level II 16K tape...$14.95
32K disk...$20.95

TECHNICAL SOFTWARE

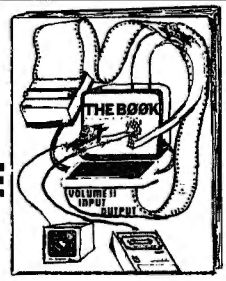
By Howard Berlin from Sams SOS
A series of seven different packages designed to increase your aid your technical knowledge and skills:

PLOTTING GRAPHS FOR LINE PRINTER
ACTIVE FILTER DESIGN
DESCRIPTIVE STATISTICS & REGRESSION ANALYSIS
ELECTRONICS I
ELECTRONICS II
ELECTRONICS III
PLOTTING GRAPHS FOR VIDEO DISPLAY

Both educational and useful, these programs include thorough and well-written documentation.

Level II 16K...$24.95 each

THE BØØK VOLUME II



From Insiders Software
Everything you want to know about video, keyboard, cassette, and print driver routines. Learn to write your own! Remarkably detailed listings illustrate well-commented source code. Complement Volume I, now.

$14.95

SUPERscript

By Rick Wilkes from Acorn
Using your SuperScript modified Scripsit Word Processor and a compatible printer, you can now underline, boldface, insert text during printout, slash zeros, subscript, set type pitch, and of course SuperScript! You can even read your disk directory and kill files without leaving Scripsit.

SuperScript comes with drivers for popular serial and parallel printers, and easy instructions for patching to your Scripsit program (does not include Scripsit).

Level II 32k Disk...$29.95

QUICK-FIX

By Kim Watt from Breeze Computing
Finally - a disk repair and modification utility that doesn't require a PhD to use! Quick Fix is a stand alone program that has its own I/O routines and does not use any ROM or DOS calls. As a result, it will operate on standard and "CP/M" machines and does not even require that the disk be in any drive after initalization.

Quick-Fix does just about everything Apparat's "Superzap" does, and so much more: Eliminate system files, kill files with common extensions, zero unused filenames and sectors, repair boot and directory automatically, change or eliminate all passwords, compute the master password, read non-standard disks, compare disk sectors, do string or sector searches.

With Quik-Fix, you can even reformat disks without losing existing files and data! Great for repairing damaged disks and refreshing old ones. Order this essential utility today.

Level II 32k Disk...$34.95

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

Circle 10

THE PROGRAM STORE
4200 Wisconsin Avenue NW, Dept.KR9Box 9609
Washington, D.C. 20016

MAIL ORDERS: Send check or M.O. for total purchase price, plus $1.00 postage & handling. D.C. residents, add 6% tax. Charge card customers: include all embossed information on card.

ITEMS AT RANDOM

Many of you have wondered if we ever got our Model III out of the shop. Yes we did, thank you. It is working 8 hours a day in our mail room with a new Floppy Disk Controller Board. It seems the original design didn't allow for the heat produced by the board's components.

I got a quick look at the new copier available only through Radio Shack's Computer Centers at the Computerama '81. It is **not** a plain paper copier. But for $795, **that would be** a miracle. It is a nice, compact book copier. I haven't seen it operate yet, but can report it is a wet process using treated paper, can produce multiple copies at about 6 a minute on 11" or 14" paper.

In reading about the new Color Computer, one magazine article talked about piggy-backing the 16K RAM chips with another set to give you a total of 32K. This sounds neat, but think this through—the dynamic RAMs used in memory produce heat, and two (one on top of the other) may produce more than can be safely tolerated by the unit. All sorts of RAMifications to consider.

We are starting to get programs for the pocket computer and would like to produce a listing directly from the machine. However, there is no way to attach a printer—or communicate with another computer—or read the pocket computer tapes with another model. We would like to see a program for the Model I or III that would read Pocket Computer program tapes. These should be independent of the ROM routines of either model.

What are some of the things you'd like to see? Do you have some pet program, utility, subject or whatever you would like to see in these pages? Send us your ideas. And, if you have something really neat, put a cover letter with it, describe it, and who knows? Maybe you'll end up being a contributor. If you want to know more on what we like and don't like, ask for our guidelines sheet.

We'd also like to see some machine language articles for the Model II and the Color Computer. Short business routines are always welcome.

One more thing we would like to see is an evaluation of some of these Stock Market programs now available. This would include the Radio Shack/Standard & Poor's STOCKPACK™ system, Max Ule's TICKERTEC™, and others.

Apple and several others received a waiver until the end of March to comply with the new regulations imposed by the FCC. All TRS-80 models *except* the Model I met these requirements which state that *thou shalt not disturb thy neighbor's television reception.* They don't care about yours, just your neighbor's. And in apartment buildings, this does present a problem.

Anyway, Tandy also got a waiver for Model I Expansion Interfaces. They can produce up to 30,000 units in 1981. That will be all unless the FCC gets enough input expressing otherwise.

While we're on the subject of the FCC, it seems that Japan does not have anything like these regulations. That is the reason you haven't seen a huge influx of Japanese personal computers. Even when they do meet our requirements, they will still face the problem of distribution which is why Radio Shack is number one in computers.

The post office has been up to its tricks again. We have seen a large increase this year in rates for overseas mailing. This has forced us to send all subscriptions out by second class mail. That translates into *surface* mail for foreign mailings.

We have a new line printer at 80-U.S. Lords was our source for a new Starwriter I. It's a great unit and produced the listings for this issue.

Programmers should be aware of a contest sponsored by Radio Shack and John Hopkins University for ideas and inventions through which personal computing may be used to aid the handicapped. The contest ends June 30th and those interested should write to:

Personal Computing to Aid the Hadicapped
John Hopkins University
P.O. Box 670
Laurel, Maryland 20810

This month we have a lot of Leo Christopherson. No apologies, just a lot of information for you on how he does it. And many of the other articles in this issue dwell in the same area.

Leo's **Line Packing** leads the way and should be a boon to those of you who want to imbed machine language programs in Basic lines but didn't know how. Not only that, but the resultant game is really fun to play. Dennis Taylor gives **An Alternative to String Packing** and Gary Sanderson provides a primer to **Animation and the TRS-80**. Paul Gerhardt comes up with a solution to those who don't like to program in graphics in his **Sketch and Pack.**

We also do a close-up on Leo that many will find interesting.

Terry Dettmann does a **CP/M** evaluation which also includes a quick look at **Filetran** and **The CP/M Handbook**.

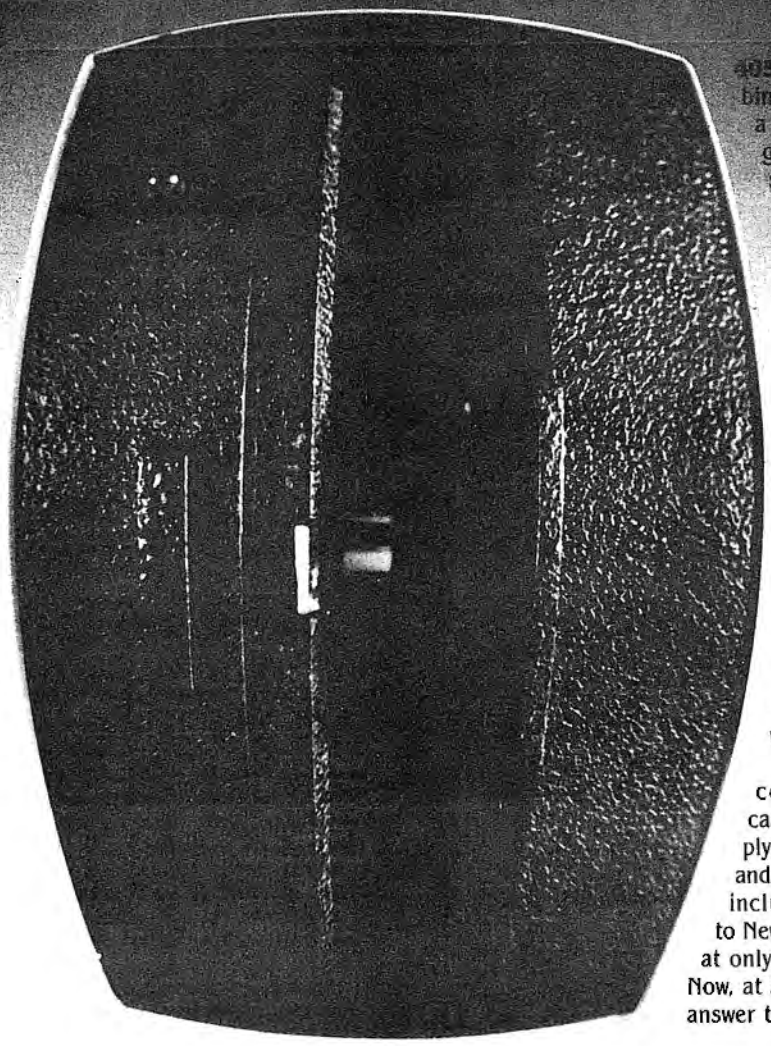
For those learning how to program in Basic, R. C. Bahn stalks the **Elusive Print Using** while Terry continues explaining random file techniques. For the advanced, Wallace Havenhill gives us a glimpse into the Basic interpreter and David Busch tells us how to configure a **Custom Operating System** using NEWDOS80. James Williams adds a **Memory Size Reset** program and Phil Pilgrim takes another look at machine language relocation with Omni Load in **System Command**. Business users will appreciate **Making Sense and Dollar$ Too** from George Haller.

We did a few reviews this month and Panattoni gets around to silencing our Line Printer I (the Centronics 779).

This month's reader service card has a spot for your comments. Let us know what you liked and didn't like. In the meantime, have some fun and enjoy life, after all, its yours to live...

Tom

APPARAT INTRODUCES
Now even more bytes per buck

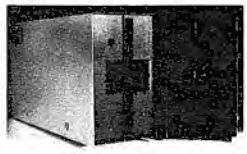


405 K bytes of storage. Apparat has combined its Newdos/80 operating system and a dual-sided 80 track mini-floppy drive to give you 405,000 bytes of storage in a single volume. Modification patches to Newdos/80 expands the capability of single density drives, so you'll have greater applications for your TRS-80 model 1.®

Drives plug directly into an expansion interface with no modification required so you can now have over 1 megabyte of storage on-line with standard mini-floppy diskettes. Each drive has 316 free grans, for a total of 948, on a maximum of three 80 track drives, which can be added to a TRS-80.

Upgrading to double density is possible by running under most double density controllers, (e.g. Percom's Doubler).

These drives can "read" standard 35 or 40 track diskettes with special software that skips every other track.



Drives come complete with case, power supply, interface cable and documentation including patches to Newdos/80. The drive mechanism is priced at only $699 (Cat. No. 1-705, specify I or III). Now, at 579 bytes per buck, it just might be the answer to your storage problems.

Apparat, Inc.

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

To order any product, call us toll-free (except Alaska, Colorado and Hawaii)

800-525-7674.

Circle 12

Now available for model III, 728K bytes of storage.

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 "flippy" 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 "flippy" 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, "flippy", 96TPI (80 track; single density unformated 250K bytes/side; double density unformated 500K bytes/side)

• MODEL DISC DRIVE 160-2 $599.95
Double-sided, 96TPI (160 track/80 per side; single density unformated 500K bytes; double density unformated 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 signle - 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.

master charge VISA

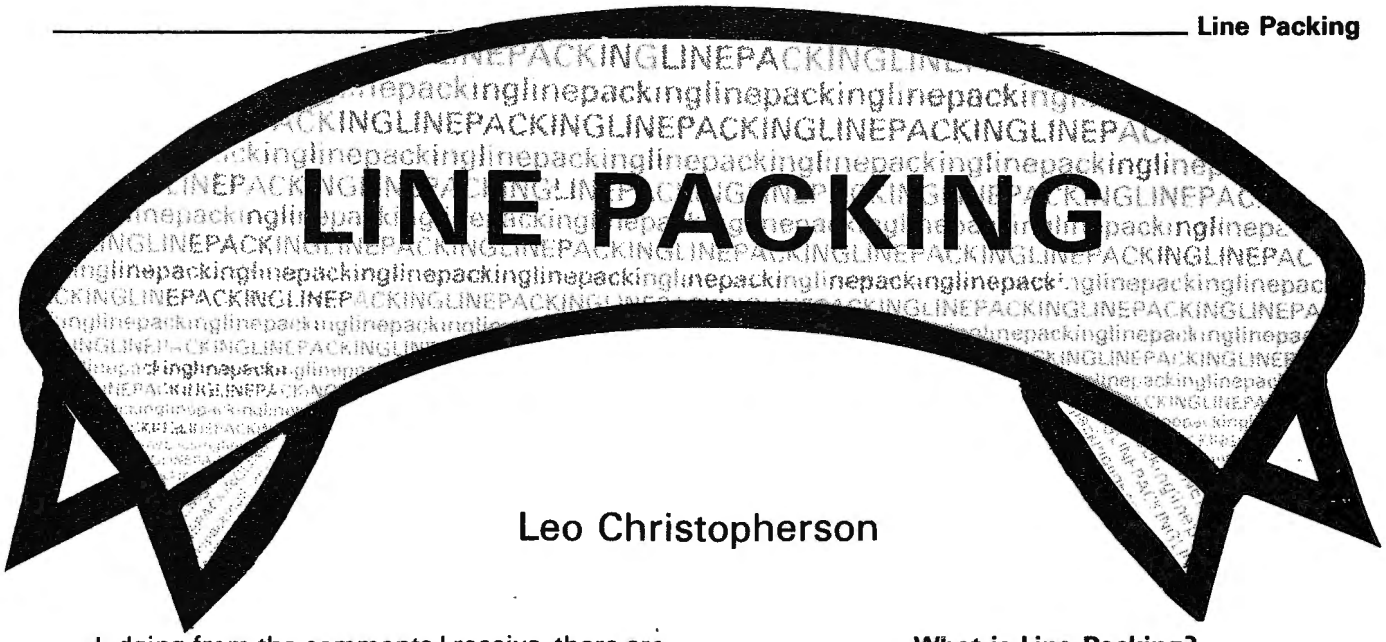
MICRO SYSTEMS SOFTWARE INC.
Specializing in the Tandy Line

(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

Circle 81



LINE PACKING

Leo Christopherson

Judging from the comments I receive, there are quite a few who study my programs in order to learn to use the techniques. These people will have noticed that all of these programs mix Basic and machine language lines. The earliest programs use mostly Basic and the latest mostly use machine language. These programs fall into two major categories: those which use "string packing" and those which use "line packing".

I first used string packing in *Android Nim*. In that program sound effect routines and graphics arrays are packed into strings. The same is true for *Snake Eggs*. *Life Two* is a program that requires string packed machine language subroutines to work out each generation of the Game of Life quickly. Sound routines and graphics are also packed into strings.

I think that *Beewary* is the program that takes string packing about as far as it should go. Most of that program is string packed. Basic is mainly used to find strings and to generate random numbers (I have found generating random numbers in machine language to be a tough problem). This is not to say that *Beewary* has very little Basic in it, however! In fact, it has so much peeking and pokeing using VARPTR that some strings extend into the expansion interface when using some disk systems, resulting in the program crashing.

And then came *Dancing Demon*. The first version of that program used string packing. But it just wasn't satisfactory...it wouldn't run fast enough for one thing. The second and final versions of *Demon* used line packing for almost all of its operation. There is a very small amount of string packing used. This change in techniques made a whale of a difference! *Duel-N-Droids* and *Fire Bird* both use line packing and very little actual Basic programming.

What is Line Packing?

Line Packing is a programming technique which allows machine language subroutines and related data to be stored and used from within a Basic program. The line numbers used in Basic serve to separate one routine from another and from the actual Basic part of the program. Subroutines are entered by typing in a dummy Basic line containing the number of bytes necessary to hold the subroutine. Then the line is packed or loaded with the actual codes for the subroutine. A special "Line Finder" subroutine is used to find the start of such a packed line. This allows the subroutine to be called from Basic or from another machine language subroutine. The Line Finder also finds the start of a line of data when needed by a subroutine. This programming technique has quite a number of advantages. It also has a few limitations.

Why Use Line Packing?

Line packing offers the programmer the benefits of both Basic and machine language programming. Where speed is not important, Basic can be used since it is easier to use for most programmers. Computation using large decimal numbers is easier in Basic. There is a whole 12K Basic ROM there which allows for some pretty fancy programming. On the other hand, sound effects, fast graphics, and certain calculations require the speed of machine language.

Line packing results in a program that can run on a TRS-80 Model I or III, with or without disk. So the programmer can write one program and expect that same program to run on any present version of the TRS-80 Model I or III.

Since the machine language routines are packed into Basic lines, there is no need to load them into a separate memory protected area. This is mainly an advantage to tape users since it seems that loading Basic programs from tape tends to be easier than loading SYSTEM tapes. Line packed programs load as a single Basic program.

Line packing does not require using an editor/assembler or a machine language monitor. All of the packing can be done from Basic, nor does the programmer need disk.

I would say that line packing is mainly for the programmer who wishes to create one single version of a program that will run equally well with tape or disk, Model I or Model III. It is for the programmer who wishes to make use of Basic as well as machine language programming, using each to do what it can do best.

Line Packing versus String Packing

Some of the reasons just mentioned for using line packing also apply to string packing. However, there are enough differences to make line packing superior to string packing. The only time string packing would be just as effective is in a program where there are only a few packed strings. The "how to do it" of string packing has been described in earlier *80-U.S. Journal* articles, so I will not go into that now.

Unlike packed strings, a packed line is just as easily used above the absolute address 32767. Addresses larger than 32767 are found in the expansion interface and the PEEKing and POKEing needed for string packing requires special attention out there. This is a problem caused by the Basic PEEK and POKE, and is not a problem for line packing.

Since lines do not need to be enclosed in quotation marks as strings do, the Z-80 code 34D, or 22H, can be used. The ASCII 34D, or quotation mark, is the very useful Z-80 instruction - LD(NN),HL - when packed into a subroutine line.

A packed line takes about six bytes of memory in addition to the actual substance of the line. These bytes have to do with line numbers and addresses. A line with a packed string takes something like five to seven more bytes to name the string and then another five or more bytes to identify and locate the string in a sort of look-up table. Thus, line packing does save some memory - the more packed lines, the greater the number of bytes saved.

In string packing it is necessary to run the program through all of the lines containing packed strings. This is done so Basic can set up its look-up tables for strings which VARPTR uses. In line packing the program must not run through the packed lines, and instead must jump around them while in Basic.

Line packing results in faster programs since it makes it easier to jump into a machine language routine from Basic and also makes it quite easy to jump to or call one machine language line from another without returning to Basic.

I feel that line packing is a better choice than string packing for a program which mixes Basic and machine language programming.

Limitations of Line Packing

Line packing, as I use it, does have some limitations. Some of these limitations are the result of compromises I have chosen to make, and some result from the inherent characteristics of line packing itself.

Because Basic identifies the end of a Basic line with a zero, zero cannot be used as a Z-80 instruction. Since zero is the Z-80 code for a NOP (no operation), this has not been a problem.

Though it can be done, packing lines with more than 248 bytes is difficult. A lot can be done in 248 bytes however, and I have not found this to be much of a problem. And anyway, one can easily cause the program to jump to another line and continue on there.

It is necessary that a pair of bytes in reserved memory always contain the address for the start of Basic. This is no problem so far, since the Models I and III both store this information at the same place. Basic's starting address is loaded into these locations at power up or boot up.

I have chosen to use LPRINT as the command that jumps into a machine language line. The LPRINT vector is at the same place in reserved memory for all present versions of Basic and DOS for both Models I and III. This means LPRINT will work the same from machine to machine. One can also use LPRINT to pass information to the machine language routine, and this is very useful.

Using LPRINT leads to some limitations though. The most obvious is that a printer can no longer be used. I have not found this to be a problem, but it certainly could be for some programmers. I would suggest using USR in that case. USR will cause the programmer to have to allow for the machine being used, and the kind of DOS (if any).

Another minor problem with LPRINT is that lines 10 and 13 should not be packed. LPRINT always concludes its cycle with a control code 13, or carriage return with line feed. Thus, line 13 would be called every time LPRINT was used. The control code 10 for line feed is also to be avoided. Since there are plenty of other line numbers available, this is not much of a problem, once you know about it.

I have also chosen to limit the range of lines to be packed to those numbered from 2 through 249 (with the exception of lines 10 and 13 as mentioned before). This leads to a somewhat simpler line finding routine. I have found there to be 50 or more line numbers still unused in large programs such as *Dancing Demon* and *Fire Bird*. I usually run out of memory before I am out of lines to pack.

As with string packing, using LIST in a line packed program can result in some rather frighteningly spectacular screen displays. One should remember to LIST from line 250 up, since only normal Basic lines exist there. Fortunately listing causes no damage to the program.

Finally, with all of this preliminary commentary out of the way, it's time to look at what line packing is really like.

The Heart of the Matter

You and I are going to put together a program. It is going to be a simple game program which uses some fairly fancy machine language subroutines and of course, line packing. Since this is the kind of program that can't easily be typed in from a magazine listing, you will need to follow through the article carefully, step by step. I think you will be quite satisfied with the result.

The entire success of line packing depends on the Line Finder subroutine, so we will have to write that first. The place to start is to find out where our Basic starts. So, power up your machine and get into Basic. Then type the following line on the keyboard and press ENTER (no line number... and, by the way, the numbers you will see in parenthesis at the left of the example lines serve to identify the line for discussion sake - they are not to be typed on the keyboard):

(1) Q=PEEK(16548)+PEEK(16549)*256:PRINT Q

This number, Q, is the decimal numeral for the address where Basic starts in your machine. Q for the Model I, Level II should be 17129. With DOS 2.3 and 0 files, Q is 26302. The Model III without disk gives us a Q of 17385. In other words, the actual value for Q changes from one model to the next and from DOS to DOS. But the memory locations 16548 and 16549 will find the start of Basic for any of these machines. Keep track of that number. Q is the address where Basic starts in your machine.

Now we have to examine a Basic line of memory. Enter the following line (use no spaces, they just take up space!):

(2) 1GOTO250

To see the structure of this line, type in the following (without a line number), and where Q is used, type in the value you found for Q in (1) above:

(3) FOR N=QTOQ+10:PRINT PEEK(N);:NEXT N

Your machine will print out something like the following:

```
M I Level II  : 242  66 1 0 141 50 53 48 0 0 0
M I DOS 2.3   : 199 102 1 0 141 50 53 48 0 0 0
M I NEWDOS80  : 200 102 1 0 141 50 53 48 0 0 0
M III NO DISK : 242  67 1 0 141 50 53 48 0 0 0
M III DOS 1.1 : 216 104 1 0 141 50 53 48 0 0 0
```

The first two numbers are the least and most significant bytes of the address for the start of the *next* line of Basic. The next two numbers are the least and most significant bytes of the present line's Basic line number. Thus, this line is (1)+(0*256) or line #1. The 141 is the token that the Basic ROM will understand as the Basic command GOTO. The 50, 53 and 48 are the ASCII decimal values for the digits of the number 250. The Basic line ends with a zero. We have no further lines in our program as yet, so there are two more zeros after this to identify the end of the Basic program.

Again, what we have learned is that the first two bytes of a line contain the address of the start of the next Basic line, and that the third and fourth bytes together give the number of the Basic line we are at.

What we are going to do is write a machine language subroutine that makes use of all of this information to find a given line for us. To make this subroutine even more useful, it will find the start of the desired line, placing its address in the HL register pair. Then, if the line starts with an "S" for subroutine, the Line Finder will jump the program to that line. If the line starts with a "D" for data, the Line Finder will store the line's address in reserved memory for later use.

The "Line Finder"

After we have put line 2 in memory, we are going to poke the address of line 2's start into the LPRINT vector so that LPRINTing any line's number will send the program flow through line 2 to find the desired line.

When the command LPRINT"!" is encountered in a program, the program flow is directed to the LPRINT vector in the Line Printer Control Block. At that point, the C register of the Z-80 contains the ASCII value of the "!" which is 33D, or 21H. In our case the "!" represents a line #33, for which we are trying to find the start. LPRINT"A" would look for line 65, since the ASCII for "A" is 65D. The same results will be obtained by the commands LPRINTCHR$(33) or LPRINTCHR$(65).

Let's see what happens in line 2. Refer to the Z-80 code listing in Figure 1.

Steps 0 - 3 of line 2 check the number stored in the C register. The program returns to the calling routine if C has 13 in it. This causes the program to ignore that carriage return with which all LPRINTs end. It also rules out using line 13, as mentioned before.

Steps 4 - 6 load the HL register pair with the address of the start of Basic. This is the Z-80 equivalent of what we did in (1) above.

Steps 7 and 8 load the address from the HL to the DE registers, giving us a place to store the address.

Steps 9 and 10 set the address in the HL register two memory positions higher. If you will remember back to what we learned about how a Basic line is stored in memory, you will realize that DE is set to the first byte of line 1. The first two bytes of line 1 are the address of the next Basic line. You should also realize that HL points to the third byte of the line. This byte contains the least significant byte of the line's number in Basic. We will be able to ignore the line number's most significant byte because we are restricting the numbers for our packed lines to those less than 250. This means that the most significant byte will always be zero.

Steps 11 - 13 check to see if the line we are at is the one we are looking for. The A register has the line number from Step 0. So, step 11 compares the line number we are looking for, which is in the A register, with the line number HL is pointing to.

If the program hasn't found the line we are looking

for yet, then steps 14 - 20 load the HL register with the address for the start of the next line of Basic, which was kept in the DE register. The address of the next line after that is placed into DE and the program flow jumps back to step 7 where the process of checking for the desired line number is repeated, and so on...

When the program comes to step 21, the HL registers are set to the address of the third byte of the line we are looking for. Steps 21 - 23 set HL over to the fifth byte of the line where we have put either an "S" for subroutine or a "D" for data. Then, after A is loaded with this fifty byte, HL is incremented once more. It is now actually pointing to the start of the line's packed subroutine of data.

Steps 25 - 28 compare that fifty byte of the Basic line, which is now copied into the A register, with the number 68D, or 44H. This is the ASCII for a "D". In our case it means this is a data line. If it is a "D" line, Steps 29 - 32 load the address of the start of this data from HL to memory locations 16424D and 16425D (that's 4028H and 4029H). These locations are two more bytes in the LPRINT control block. Doing this gives us a way to recover the data line's address when we are back in Basic. If this was a "D" line, the programs return at this point.

In steps 33 and 34, the program checks to see if the line is an "S" or subroutine line. It compares the fifty byte to 83D, or 53H, which is an ASCII "S". If we are at neither a "D" or an "S" line, the program returns to the calling routine with the HL registers still pointing to the start of the packed part of the line.

If the fifth byte is an "S", then step 36 causes the program flow to jump to the address in HL where the packed subroutine awaits in the Basic line. The program flow does not come back through line 2, but is returned from the packed subroutine.

Packing Line #2 into our Basic Program

In the following instructions for building up our program, asking you to save your program on tape or disk is a safety precaution. One wrong byte in a machine language subroutine and all will be lost when the program is run.

Type in the following Basic lines and then save your program. Then run the program once - *only once so far!* Line 2 will be packed by this time, but not quite ready to use.

```
(4)    1GOTO250            (USE NO SPACES)
(5)    2.................(37 PERIODS).......
(6)    250 Q0=PEEK(16548)+PEEK(16549)*256+13

(7)    1000 DATA121,254,13,200,42,164,64,84,93,35
       1001 DATA35,190,40,7,235,94,35,86,235,24
       1002 DATA242,35,35,126,35,254,68,32,4,34
       1003 DATA40,64,201,254,83,192,233

(8)    1011 RESTORE:FORN=0TO36:READD:POKEQ0+N,D:NEXTN
(9)    1020 STOP
```

Now we need to test line 2 to see that it is packed properly. Enter the following lines and run the program again.

```
(10)   33 D0123456789
(11)   251 Q2=INT(Q0/256):Q1=Q0-Q2*256
(12)   252 POKE16421,195:POKE16422,Q1:POKE16423
       ,Q2
(13)   255 LPRINT"!"
(14)   260 L=PEEK(16424)+PEEK(16425)*256
(15)   265 FORN=0TO9:PRINTPEEK(L+N);:NEXTN
(16)   270 STOP
```

If everything is correct so far, you should see the following numbers printed out on the screen:

(17) 48 49 50 51 52 53 54 55 56 57

If this does not happen, something is wrong! Start back at the beginning of this section and try again. If all seems to be as it should be, save the program again.

Let's examine what we have just done. In (5) we set up a dummy line of bytes to receive our Line Finder subroutine. The calculation at (6) finds where the line starts by first finding the start of line 1 as we did in (1). Then the number 13 is added to this address which counts this address past line 1 and gets us to the actual start of the subroutine to be in line 2. Step (7) is the data representing the decimal values of our Z-80 coded program to go into line 2. The line at (8) packs line 2 by poking the data into it.

To test the Line Finder, we set up the LPRINT vector by finding the least significant (Q1) and most significant (Q2) bytes of the starting address of the Line Finder routine. This is done in (11). That POKE16421,195 sets up a machine language jump (Z-80 code 195D) so that a machine language call to that address (16421) will result in a jump to the Line Finder, whose address is poked into the LPRINT vector (16422 and 16423) by (12). This means we will be able to get to line 2 in two ways: by LPRINTing in Basic, or by calling 16421 (4025H) from a machine language program.

What we do in (10) is set up line 33 for test purposes. It starts with a "D" which identifies it as a data line. That means that its address will be stored at memory locations 16424 and 16425 by the Line Finder. In (13) when we LPRINT"!", where "!" is an ASCII 33D, the program jumps to the Line Finder, looking for line 33. When the line is found, its address is stored at 16424 and 16425.

The Basic line in (14) calculates the decimal address of line 33 and the line in (15) prints out the ten bytes of data in line 33 (not including the "D"). So what we get in (17) is the decimal ASCII values for 0123456789.

Now that we have the Line Finder working, we are ready to start packing a program. What we have done so far is to set up the line packing framework around which a program can be built.

Packing Lines 5, 33 and 32

Look at the listing in Figure 2 for line 5: Screen Painter. Step 0 is the "S" which identifies the line as

(Text continues on page 95)
(Listings start on page 94)

Be a computer Picasso . . . Disney . . . DaVinci . . . On your TRS-80*

Introducing Programma's Hi-Resolution 80-GRAFIX™

For sheer fun and flexibility nothing compares with Programma's exclusive 80-Grafix for your Model I or Model III TRS-80. With its greater high-resolution capability, it outgraphics the Commodore CBM/PET*, and even Apple II*. In short, anything they do, you can now do better.

As for effective resolution, imagine this . . . 80-Grafix gives your TRS-80 an effective screen of 384 x 192 (vs normal 128 x 48) . . . while the CBM/PET is 80 x 50, and Apple II, 280 x 192. And that's only the beginning.

Black & White. Or Vice-Versa. Use the standard white on black, or for variety, call up Inverse Video as the need demands.

Lower Case. Free. No need to spend extra. 80-Grafix includes two sets of lower case characters. Plus Character Generator software so you can create your own set of up to 64 original characters.

Great Graphics. Here's unlimited fun. Pick a subject. Draw it. Sketch it. Design it. Change it. You're limited only by your own imagination.

Games People Play. 80-Grafix lets you develop your own software to create fantastic new real-time games to amuse, amaze or challenge. Or, order our ready-to-go software. It adds a new dimension to 80-Grafix without any extra programming on your part.

Easy Does It. It's simple to hook up 80-Grafix inside your TRS-80 case . . . takes just a few minutes. While this voids Radio Shack's warranty, it's a minor inconvenience compared to the usefullness and excitement you gain.

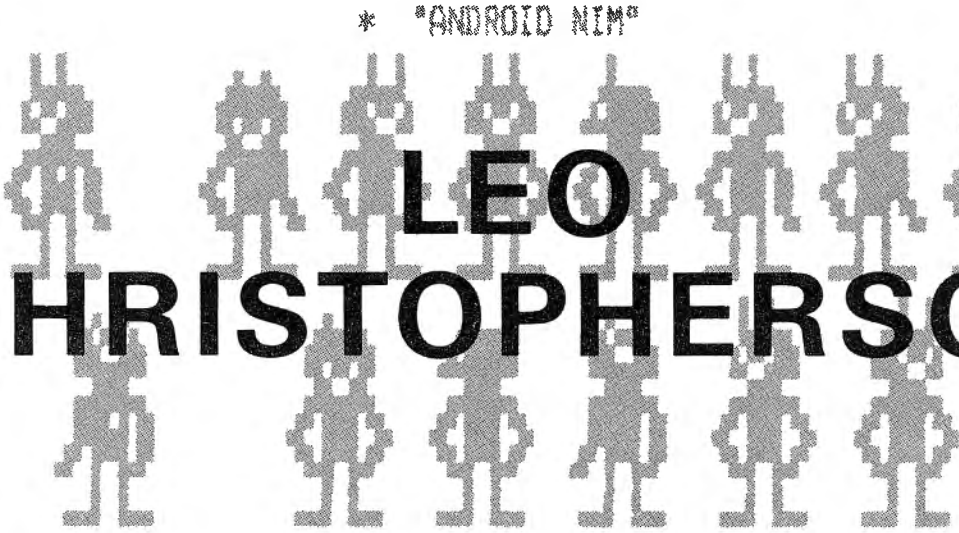
The Rush Is On. Frankly, this is one of the hottest items we've offered. Don't miss out. At your dealer now, or order direct for immediate delivery. (Specify version for Model I or Model III.) **Introductory Price:** $149.95 plus $4.50 shipping. Includes free lower case and demo programs. (Visa & Mastercharge accepted. Calif. residents add 6% sales tax.) Dealer inquiries invited.

PROGRAMMA

Programma International Inc.
2908 No. Naomi Street, Burbank, CA 91504
(213) 954-0240

Circle 14

* 80-Grafix is a trademark of Programma International, Inc • TRS-80 is a trademark of Tandy Corporation
Apple II is a trademark of Apple Computer, Inc • Commodore CBM/PET is a trademark of Commodore International Ltd.



LEO CHRISTOPHERSON

At the time when most TRS-80 users were still trying to put intelligent looking graphics characters on the screen, a program called "Android Nim" appeared. It not only had characters, they *moved*! And they were not slow either, they blinked their eyes, nodded heads and shot each other off the screen. The Androids were first featured in the Nov/Dec 1978 issue of 80-U.S. At that time, fast animated graphics really caught some attention, and Leo Christopherson was their author.

He went on to write others - Snake Eggs, Beewary, Dancing Demon, Duel-N-Droids and more, some for the Apple computer.

Leo's fast graphics were a result of his "string packing technique", which allows packing machine language code into strings for fast action. For a while, it was called "super graphics" by others, but it never caught on and rightfully so, since string packing can be used for other than graphics.

Over the past couple of years we have had many requests for more information about Leo. Some asked if we chained him to his computer to get his programs. Others just wanted to know more about him. One person called him the "Walt Disney of TRS-80 Graphics".

Leo (see cover photo) is a very quiet, soft-spoken school teacher. He teaches science and mathematics to seventh and eighth grades in a school district near Tacoma, Washington. His students enjoy his classes because he makes his subjects interesting and challenging. His school district has several TRS-80 and Apple computers, which Leo uses in his courses. He uses his computer for grading, and has recently completed a Gifted Student program to run on the school computers.

At home, Leo is a very private person, and surrounds himself with the things he has an interest in. He owns a three-manual Rogers pipe organ which he plays with considerable expertise. He is also a composer of sorts. One day while there, he played a version of his own composition, based on a Beatles song, called "The Up and Down Toccata and the Everlov'n Fugue". It almost made Bach sound like a fake!

Aside from organ music, Leo has a nine voice music board in his Apple computer, and creates versions of his own as well as music from Scott Joplin and others. As with all things Leo does, these come off with meticulous preparation and care.

Aside from music, he is an avid science fiction buff, likes philosophy, and art. He does pen and ink sketches and works with oils on occasion. He is the artist on several 80-U.S. covers and several inside drawings. (Note the drawing which accompanied "The Great 80-U.S. Chess Tournament", in the Nov/Dec 79 issue). Leo signs his works with .037, which when read upsidedown says Leo.

Leo currently uses a Model I TRS-80 with disk, an Apple II with disk, a Model III and the pocket computer. His school computer projects keep him busy, but he is working on a new graphics game for the Model I/III. His latest technique, line packing, (see previous article) allows him to write machine language in Basic, thereby gaining the best of both methods. Perhaps when he adds a color computer to his collection we will see even more of his genius.

Leo works alone. His creations show a definite subtle humor which is uniquely his. If you ever have the good fortune to spend an afternoon with him, you see the same humor show through in his speech and manner. Above all though, he likes being a teacher, and views the computer as an interesting diversion. ●

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 accomodate 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

Circle 7

An Alternative to *String Packing*

Dennis Taylor
Coquitlam, British Columbia

Do you have problems with string packing machine language routines?

String packing does offer the advantage of needing no "memory size". But there is the problem of avoiding code 34 and zeros in the code, not to mention the constant struggle with string variable pointers.

In the May/Jun 1980 issue of 80-U.S. Leo Christopherson gave us the machine code for his sound routines. The system presented here adapts that same demo program to an integer array which avoids the problems involved in string routines and the memory size question.

The TRS-80 stores integer array variables in continuous blocks in memory, with each element pointing to two bytes in the block that defines it's value, i.e., the value for A(1) may be stored in locations 30000 and 30001, the value for A(2) in locations 30002 and 30003, and so on. Integer elements are stored with the least significant byte (LSB) below the most significant byte (MSB).

It is important to note that any integer with a MSB of more than 127 will be stored as a negative number, and trying to stuff a number greater than 32767 into an integer variable will give you an FC? error. Therefore, line 50 in the accompanying program converts this condition to the proper integer.

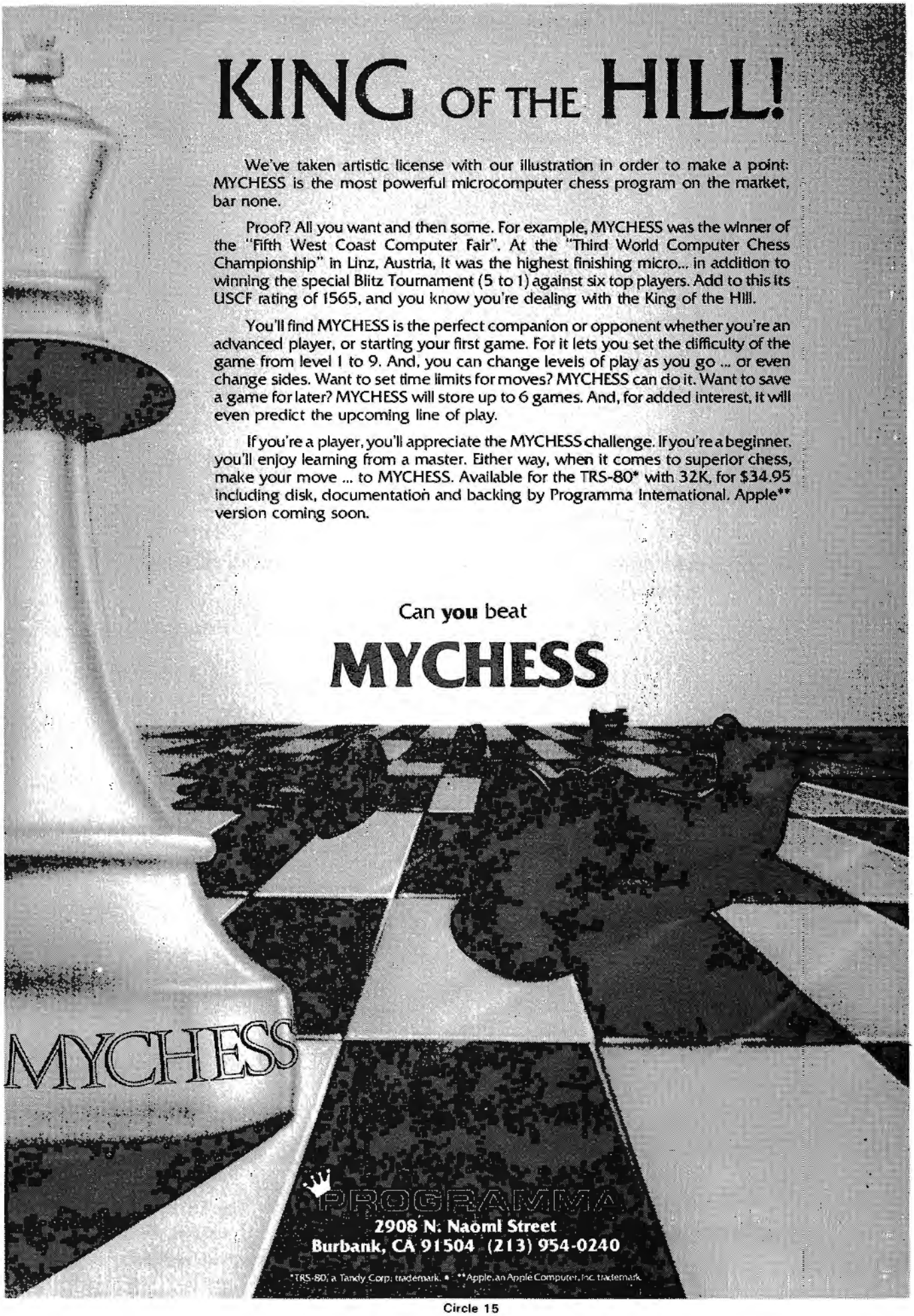
Integer packing is very convenient because you *can* use 34 and zero in your routine. Not only this, but it is safe, becuse the only pokes involved are used to set up the USR call. (Many a program has been lost through indiscriminate poking!)

Here is a rundown on how the program operates:

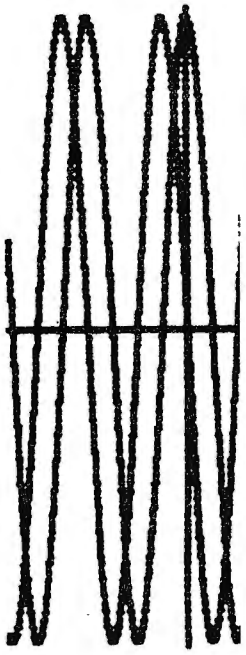
10 Creates the integer array
30 Gets data from data lines 210-270
40 Converts two data elements into an integer
50 Test and store properly
70 Set up call address
80 Test for disk
90 Define USR for Level II Basic
100 Define USR for disk Basic
110 Choose your sound
140 Tell routine your choice (as determined in lines 120 and 130)
170 Random pitch for choice #1
180 Random duration for choice #2
190 Random duration for choice #3
200 Execute machine language program

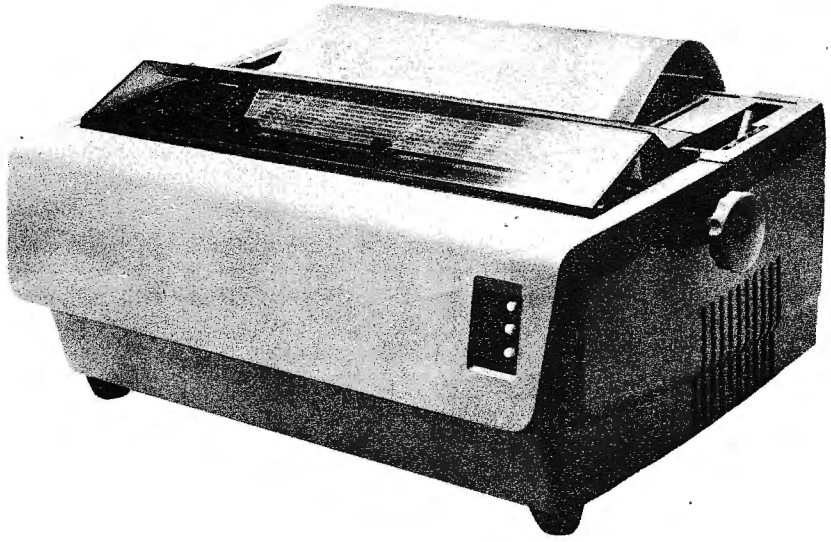
Have fun with this alternate method to machine language routines in Basic!

```
LISTING OF ALTER/BAS        PAGE 1
10 CLS:DEFINT A:DIM A(37)
20 FOR X=1 TO 37
30 READ A1,A2
40 B=A2*256+A1
50 IF B>32767 THEN A(X)=B-65536 ELSE A(X)=B
60 NEXT X
70 U=VARPTR(A(1)):U2=INT(U/256):U1=U-U2*256
80 IF PEEK(16396)=201 THEN 90 ELSE 100
90 POKE 16526,U1:POKE 16527,U2:GOTO110
100 DEFUSR=U:CMD"T"
110 PRINT@400,"PRESS 1, 2, OR 3"
120 Z$=INKEY$:IF Z$="" THEN 120
130 IF Z$<>"1" AND Z$<>"2" AND Z$<>"3" THEN120
140 A(4)=(254*256+VAL(Z$))-65536
150 ON VAL(Z$) GOTO170,180,190
160 GOTO110
170 A(9)=2560+RND(255):GOTO200
180 A(18)=123*256+RND(5):GOTO200
190 A(28)=100+RND(5)*256:GOTO200
200 K=USR(0):GOTO110
210 DATA33,1,2,14,255,62,1,254,2,40,21,254
220 DATA3,40,38,17,200,100,237,97,67,16,254,237
230 DATA105,67,16,254,21,32,243,201,17,1,5,123
240 DATA237,97,71,16,254,237,105,71,16,254,60,32
250 DATA243,21,32,239,201,17,100,5,123,237,97,71
260 DATA16,254,237,105,71,16,254,61,32,243,21,32
270 DATA239,201
280 END
80 END
70 END
```


KING OF THE HILL!
We've taken artistic license with our illustration in order to make a point: MYCHESS is the most powerful microcomputer chess program on the market, bar none.
Proof? All you want and then some. For example, MYCHESS was the winner of the "Fifth West Coast Computer Fair". At the "Third World Computer Chess Championship" in Linz, Austria, it was the highest finishing micro... in addition to winning the special Blitz Tournament (5 to 1) against six top players. Add to this its USCF rating of 1565, and you know you're dealing with the King of the Hill.
You'll find MYCHESS is the perfect companion or opponent whether you're an advanced player, or starting your first game. For it lets you set the difficulty of the game from level 1 to 9. And, you can change levels of play as you go ... or even change sides. Want to set time limits for moves? MYCHESS can do it. Want to save a game for later? MYCHESS will store up to 6 games. And, for added interest, it will even predict the upcoming line of play.
If you're a player, you'll appreciate the MYCHESS challenge. If you're a beginner, you'll enjoy learning from a master. Either way, when it comes to superior chess, make your move ... to MYCHESS. Available for the TRS-80* with 32K, for $34.95 including disk, documentation and backing by Programma International. Apple** version coming soon.
Can you beat
MYCHESS
MYCHESS
PROGRAMMA
2908 N. Naomi Street
Burbank, CA 91504 (213) 954-0240
*TRS-80, a Tandy Corp. trademark. • **Apple, an Apple Computer, Inc. trademark.

Circle 15





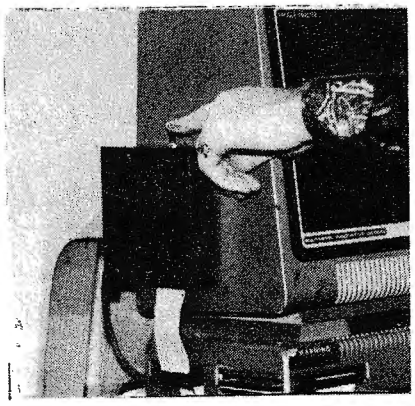
Starwriter I - 25 cps - $1795
Starwriter II - 45 cps - $2195

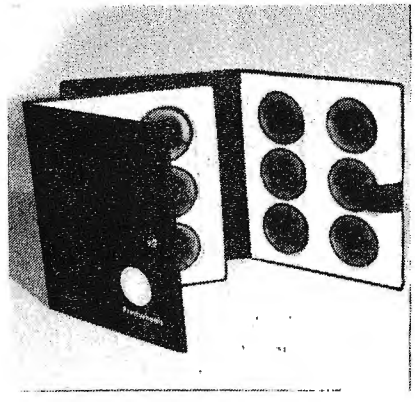
You knew it could print...
Did you know it can PLOT??

Cables: $38 - specify TRS-80 Model I, II or III

TWO Printers on ONE Computer

Our WYE Interface is bidirectional.
CMOS switched lines.
For TRS-80 Model I and III - $119.00
For TRS-80 Model II - $149.95
Do it with a flick of a switch!





PRINT WHEELS

For your Daisy Wheel Printers
Set of 6, complete with case.

$69.00

LORDS
Small Systems Design
P.O. Box 99
Port Angeles, WA 98362
Telephone (206) 457-3064

LABYRINTH!

Labyrinth is not easy. Waking a sleeping watch-dragon is the least of your troubles. Somewhere in this nightmare of tunnels lurks an evil minotaur. To escape, you must kill it. And survive.

Only the legendary light sabre of Dnev can kill the minotaur. It is secreted away in stygian darkness. To get it, you must deal with cave bears, gnomes, pits, wraiths, and much more. You must avoid the minotaur at all costs.

Alas! Once armed with your light sabre, the minotaur flees! What treasures must be found and used as bait? Why is the maiden so beautiful? What lies beyond the fog . . . .

3-D PERSPECTIVE GRAPHICS

Labyrinth features the full screen 3-D perspective graphic displays that have made Asylum and Deathmaze 5000 best sellers. You can actually see what you are doing and where you are going! The mazes and buildings are bit-coded. This allows us to store **gigantic** mazes in small amounts of memory. These programs are **not** just a series of stored pictures. Our mazes typically contain **over 600 locations.** Further, machine-language programming gives instantaneous graphics generation and game response!

LABYRINTH places you in a huge maze of tunnels inhabited by gnomes, ghosts, witches, and an evil minotaur. You must find the weapons and treasures needed to destroy the minotaur before he destroys you! There is food enough in the maze to hold out for months!
TRS-80 Level II 16K or Model III 16K $12.95

ASYLUM places you on a cot in a small room. Periodically, a janitor lobs a hand-grenade through the window of your locked door. What you do next could mean survival and escape! It could also mean permanent residence in the home for Deathmaze survivors! To leave, you will have to deal with guards, fellow survivors, doctors, the infamous Crazed Carpenter, and much, much more. Don't expect to get out any time soon! There are over 1200 locations!
TRS-80 Level II 16K or Model III 16K $14.95

DEATHMAZE 5000 places you in a gigantic five-story building. There is only one goal. **ESCAPE ALIVE!** Monsters, dogs, vampires, and other vile horrors will plague your every step as you struggle to survive one of the most challenging adventures ever written. As of December 20, only two people outside the Med Systems' staff were known to have escaped!
TRS-80 Level II 16K or Model III 16K $12.95

SATISFACTION GUARANTEED!
Asylum, Deathmaze 5000, and Labyrinth are guaranteed to be the most incredible 16K 3-D graphic adventures you can buy. If for any reason you are not satisfied with these products, return your order within 14 days for a prompt and cheerful refund.

ORDERING INFORMATION
Orders are processed within five working days. We pay all postage and handling within the U.S., Canada, and U.S. territories. European orders please include $3.00 for air post.

☐ Asylum ($14.95) $ _____

☐ Deathmaze 5000 ($12.95) $ _____

☐ Labyrinth ($12.95) $ _____

☐ Programs on Disk (add $4.00) $ _____

☐ Programs on Cassette (N/C)

TOTAL $ _____

Name _____

Street _____

City _____ State _____ Zip _____

Computer:
☐ TRS-80 LII 16K ☐ Model III 16K

☐ Mastercard ☐ VISA ☐ check

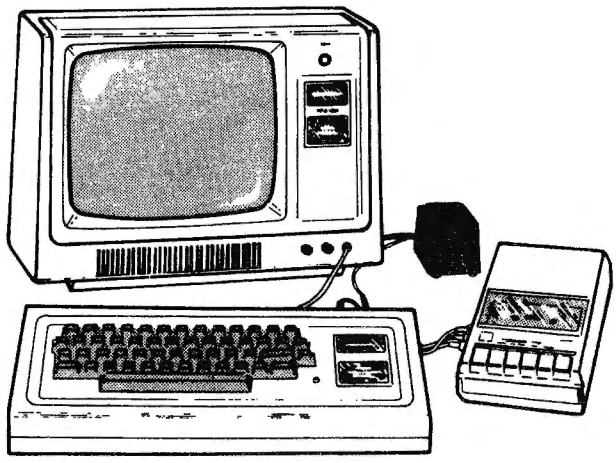
MC or VISA # _____

Expiration Date _____

Circle 21

Med Systems Software
P.O. Box 2674-Y Chapel Hill, NC 27514
(919) 933-1990

FINE TOOLS FOR YOUR TRS-80



UTILITY I

Here's software utility that can make programming easier for you. These powerful tools will take much of the frustration out of essential operations. You'll wonder how you managed without them. They'll stamp your work with the mark of professionalism.

RENUM—Now you can easily renumber any Level II program to make room for modification or to clean up the listing. (Only for use with 16K of memory.)

DUPLIK—This program will let you duplicate any BASIC, assembler, or machine-language program, verify the data, and even copy Level I programs on a Level II machine.

See how efficient and well-ordered your programming becomes. (T1)

Order No. 0081R $9.95.

UTILITY II

One of the most popular and useful of the utility packages, Utility II is the second in a series of programs designed to take the drudgery out of editing.

This combination of programs adds flexiblity to your BASIC programs by allowing you to combine them with other BASIC programs, or with machine-language programs or routines.

CFETCH—You'll be able to merge BASIC programs, with consecutive line numbers, into one program. CFETCH can also search through any Level II program tape and display the file name for all the programs.

CWRITE—Combine subroutines that work in different memory locations into one program. CWRITE works with BASIC and/or one or more machine-language programs. It will even give you a general checksum to verify that your program hasn't dropped any bits.

Use these two programs as your dependable "assistants" to speed merging processes and as a totally reliable retrieval means in search and edit operations.

A fine tool for your TRS-80! (T1)

Order No. 0076R $9.95.

COMPRESSION UTILITY PACK

Do you want to add sound routines to a Space Trek program that already uses 16K? Or maybe you need an extra column in that financial report program, but when you run it, you get OM errors?

With a wave of your hand, and a little help from either of the COMPRESSION programs, in this package, your problems are over.

Compress-80—fits in 265 bytes, deletes spaces, and offers the choice of leaving REMark line numbers in the program, or deleting them altogether.

Supercompress—uses 767 bytes and can do everything COMPRESS-80 can do PLUS, it packs the program into the smallest possible number of multiple statement lines.

With the Compression Utility Pack and your own programming skill, you can add all those little extras to your BASIC programs. (T1)

Order No. 0246R $19.95.

ULTRA-MON

ULTRA-MON is a unique and powerful machine-language monitor. It is ROM independent and will function in Level II or DOS BASIC. With ULTRA-MON, you will be able to write, modify, study and debug machine-language programs. Plus, you'll be able to avoid the frustrations and "bomb-outs" usually associated with machine-language programming. ULTRA-MON displays, disassembles, traces (hardcopy trace disassembly, too!), modifies, relocates memory, prints and even relocates itself with simple commands. Using interpretive execution, ULTRA-MON allows you to put breakpoints in ROM. This powerful monitor can even fetch, decode, disassemble and analyze each instruction individually so that your program cannot bomb out.

ULTRA-MON is designed for the beginning machine-language programmer as well as the professional. The documentation contains a Simple Demonstration section geared to the novice programmer. Consequently, the program is a learning device as well as an extremely useful programming tool.

If you are serious about programming, you need to add this powerful utility to your library today. (T1)

Order No. 5003R $24.95.

TLDIS & DLDIS

You've bought a super machine-code program, but now wonder how it works. Maybe you even used a quick PEEK routine to glance through it when it was in memory. If so, you definitely noticed the complete lack of comments in the code, making it almost impossible for you to decipher and understand it.

Well, Instant Software's Labeling Disassemblers are the answer to your problem.

TLDIS (Tape-based Labeling DISassembler) and DLDIS (Disk-based Labeling DISassembler) are three-pass, label-assigning disassemblers which assign labels (where appropriate) to the routines in a machine-language program. Their output is almost identical to that of a hand-assembled source code.

You can send the disassembly to a line-printer (Radio Shack parallel port) for either TLDIS or DLDIS. (The difference between these utilities is the storage mode of the disassembly.)

TLDIS can send the disassembly to cassette tape, DLDIS can send it to disk; both send it to the video monitor. The stored disassembly from TLDIS may be reassembled with Radio Shack's EDTASM™—the disassembly from DLDIS, with Apparat's extension of EDTASM™. Because of the use of labels, it is a simple matter to change any object code program by disassembling it and then making changes to the resultant source code, without losing track of jump/load addresses. Labels start with "AA00" and increment up, in even numbered steps (AA02, AA04, etc.). The odd numbers (AA01, AA03, etc.) are left for you to use for the source code during reassembly.

The printing of the disassembly may be temporarily halted by using [SHIFT] @ (just as in BASIC) or it may be ended by pressing the [BREAK] key. It also has a comments column to display ASCII characters used in a LD or CP opcode.

TLDIS and DLDIS may be relocated in memory to avoid conflict with the program you disassembler.

The next time you need to "climb inside" a machine-code program, take DLDIS or TLDIS with you. We promise that it will be an easier journey.

Order No. 0230R (TLDIS) $14.95 (T1)
Order No. 0231RD (DLDIS) $19.95 (T2)

TO ORDER:

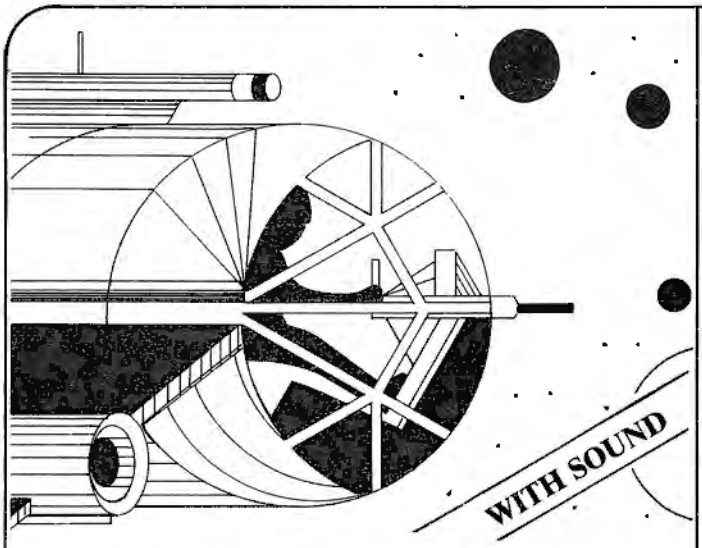
See Your Local
Instant Software Dealer
or
Order Directly:
CALL TOLL FREE
1-800-258-5473

Instant Software™

PETERBOROUGH, N.H. 03458
603-924-7296

GO BOLDLY . . .

Where No TRS-80* Program Has Gone Before!



ASTEROID

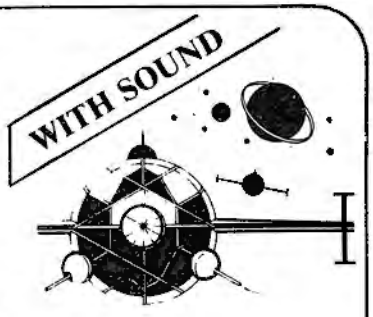
DATE: 28.02.2047
LOCATION: 270 million miles from Terra
MISSION: Maintaining Terra's Space Lanes

Briefing will follow:
1.1 Your mission is to destroy any asteroids in your sector and to prevent alien spacecraft from infiltrating the Terran Defense Network.
1.2 Your ship is armed with an anti-matter cannon. You can shoot large asteroids, but this turns them into many smaller asteroids, each capable of destroying your ship.
1.3 In addition, alien ships can make instantaneous hyperspace jumps into your area and start firing on your ship.
1.4 You'll need lightning reflexes and nerves of steel to survive Asteroid. We have no use for non-survivors!

Asteroid, a real-time, machine-language game, features variable levels of difficulty, superb high-speed graphics, sound effects and automatic score keeping. (T1) or (T2)

Order No. 0237R
$14.95 Tape.
Order No. 0247RD
$19.95 Disk.

BALL TURRET GUNNER

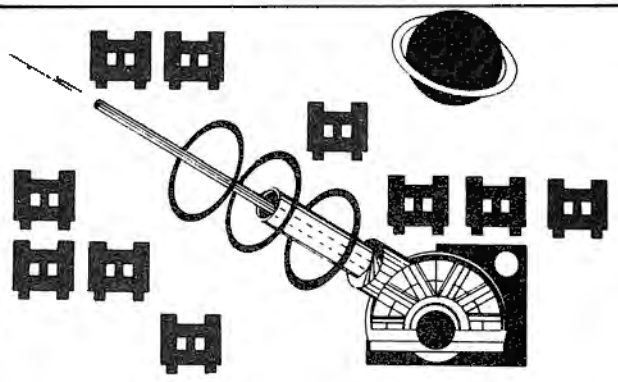


For years the Petro Resource Conglomerate has attacked our photon collection stations and strangled our deep-space trade routes. The PRC Exxonerator Class light fighters (code name: Gnat) have been their main weapon. Now you can strike back, by joining the Ball Turret Gunner Service.

Imagine yourself at the control console of an LW-1417 Stratoblazer (Type B Strategic Laser Weapon). Your Hindsight Director informs you that a Gnat fighter is coming in for an attack. You pivot your gigawatt laser turret until you can see the target on your monitor. The Range Indicator shows him coming in fast. The Targeting Computer studies his course and speed as your finger tenses over the firing key. You know you'll have only a fraction of a second in which to react. The Gnat fighter's evasive maneuvers cause him to dance in your sights. Suddenly, you see the FIRE Command and you react instinctively. Your laser beam lashes out and reduces the Gnat to an expanding ball of ionized gas. Mission accomplished!

Ball Turret Gunner, with your choice of multiple levels of difficulty, optional sound effects and superb graphics, is more than just a game. It's an adventure. Experience it! (T1)

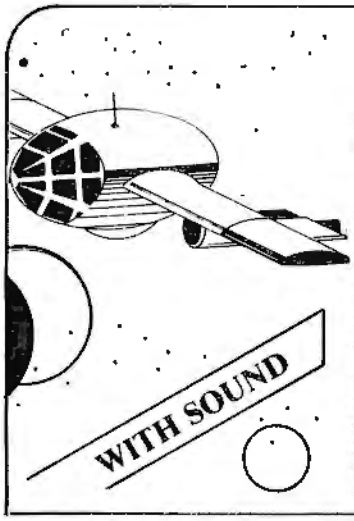
Order No. 0051R $9.95.



INVADERS

The INVADERS are coming! Earth's defenses are dead except for your Laser base. Your assignment is to destroy the approaching INVADERS before they destroy Earth. Before Earth's sensors failed, they detected 550 armed invaders in space, speeding toward us in 10 attack formations of 55 in each group. The sensors detected four different types of attack craft: Large, Medium, Small, and a short profile craft which is the most difficult to destroy. If you cannot stop these space attackers they will stop Earth . . . . . for good. (T1)

Order No. 0240R $9.95.



COSMIC PATROL

WARNING: PLAYERS OF THIS GAME SHOULD BE PREPARED FOR A STATE OF REALISM HITHERTO UNAVAILABLE ON THE TRS-80

Skilled players soon master many difficult computer games, but COSMIC PATROL is in a world all its own. The challenge intensifies! Supporting graphics and sound (optional) make each encounter an exciting new experience. It all adds up to a Super 3-S package . . . skill, sight and sound.

Scenario: The Cosmic Patrol program puts you in the command chair of a small interstellar patrol craft. Your mission is to defend Terran space and prey on the Quelon supply ships which carry essential parts and lubricants for that implacably hostile robotic force. The drone freighters are fairly easy pickings for the accomplished starship pilot, but beware of the I-Fighter escorts. They're armed, fast and piloted by intelligent robots linked to battle computers. They *never* miss.

The Cosmic Patrol program is not just another search and destroy game. With its fast, real-time action, impressive sound option and superb graphics, this machine-language program is the best of its genre.

Don't keep putting quarter after quarter into arcade games or spending big bucks for video game cartridges. Get Cosmic Patrol from Instant Software—and get the best for less! (T1)

Order No. 0223R
$14.95 Tape.
Order No. 0224RD
$14.95 Disk.

TO ORDER

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

Instant Software™

PETERBOROUGH, N.H. 03458 603-924-7296

(T1) = TRS-80 Model 1, Level II, 16K RAM.

(T2) = TRS-80 Model 1, Level II 16K, Expansion Interface 16K + 1 disk drive.

*A trademark of Tandy Corporation

PANATTONI'S PANACEA
Silencing the 779 Printer

Larry Panattoni

The Sep/Oct 1980 issue carried my article explaining how the original Radio Shack line printer (Centronics 779) could be modified, without cutting wires, to print lower case as well as standard upper case letters. One of the most requested follow-up projects for the 779 printer was one which would automatically silence the noisy motor associated with that printer.

What is needed is a project which will normally keep the motor in an off condition. Then, upon sensing that data is to be printed, it should automatically turn on the motor. When printing is completed, it should shut off the motor without action from the operator or additional commands inserted within a program.

This issue's project will do just that. It is one I have had in operation on my 779 for a little better than a year now, and it has not failed me once. This project can be implemented without cutting any wires, and the total cost is less than $15.00.

How it Works

In the Nov/Dec 1980 issue, I explained how the TRS-80 reads four bits from the printer to check its status prior to sending data. Only two of these four status leads are used by the printer, one for printer internal busy and one for out of paper.

If any one of these four status lines indicate busy, the TRS-80 will read them again and again until they indicate they are not busy. At that time it will send data to the printer. Either or both of the two unused status lines can be wired to indicate some other type of busy, and that is what the circuit shown in Figure 1 does.

Figure 1 shows the output of Q3 connected to one of the two status lines not normally used (pin 32 of the printer bus). A physical view is shown in Figure 3. This lead will be normally held low by Q3, indicating the printer is not ready because the motor is not on. This will cause the TRS-80 to continue to read the printer status lines. However, upon the first reading of the status lines, the read pulse itself is monitored by pin 1 of IC1 and passed on to timer "A" (pin 6 of IC2).

This negative read pulse causes the output of timer "A" (pin 5) to go from low to high and remain high for a set period of time (determined by C1 and R2), adjustable from 1.5 to 10 seconds. This high output is passed through D1 and on to RL1, a 5 volt DIP relay. RL1 operates and keys the Triac (Q4) into conduction, which provides power to the printer's motor. This DIP relay, Triac and RL1 circuit is identical to that used in my "External Expansion Unit", featured in a previous issue.

The motor now turns on, but the TRS-80 still sees a busy condition from Q3. Before continuing, let me explain that timer "B" is an 800 millisecond timer, which is being kept from timing out by Q2. The high output from timer "A" goes not only to D1 for turning on the motor, but also to Q2, turning Q2 off. Now timer "B" can begin its 800 millisecond timeout.

After this 800 millisecond timeout, the output of Q3 is forced high, removing the busy signal from the printer's newly used status lead (pin 32). This indicates to the computer that the printer is ready to receive data. This 800 millisecond delay is necessary to allow the motor to build up to its proper speed.

Timer "A", as mentioned above, has an on-time of 1.5 to 10 seconds, determined by R2. This does not mean the printer will be on only for that amount of time. The TRS-80 sends out a read pulse prior to each and every character it sends to the printer. Q1 receives this read pulse from IC1 and resets timer "A" back to the beginning of its time duration. Therefore, as long as read pulses arrive at Q1, timer "A" will never timeout.

If the set time should lapse without a read pulse arriving, it indicates that printing is finished. Then the output of timer "A" drops low, and the motor shuts off.

The operation sounds complete - so what is IC1 for? IC1 sets its second input (pin 2) from ME 24, pin 8 (from within the printer) which is an inverted status signal. It has a high signal when busy and a low when idle (ready to accept data).

As long as this signal is low (idle), the read pulses will pass through IC1 as if it was not there. But, say you run out of paper and need to add more, or you want to hold up printing for some other reason; all that is necessary is

for you to depress the PRINT switch on the face of the printer to the off position. When this happens, pin 2 of IC1 goes high (indicating busy) and IC1 no longer will pass the read pulses to Q1 or timer "A". Hence, timer "A" times out and the motor shuts off. You are now free to change paper, do your calculations or whatever, in peace and quiet. Without IC1, the motor would have continued to run during this time.

Physical Connections - Adding Jumpers

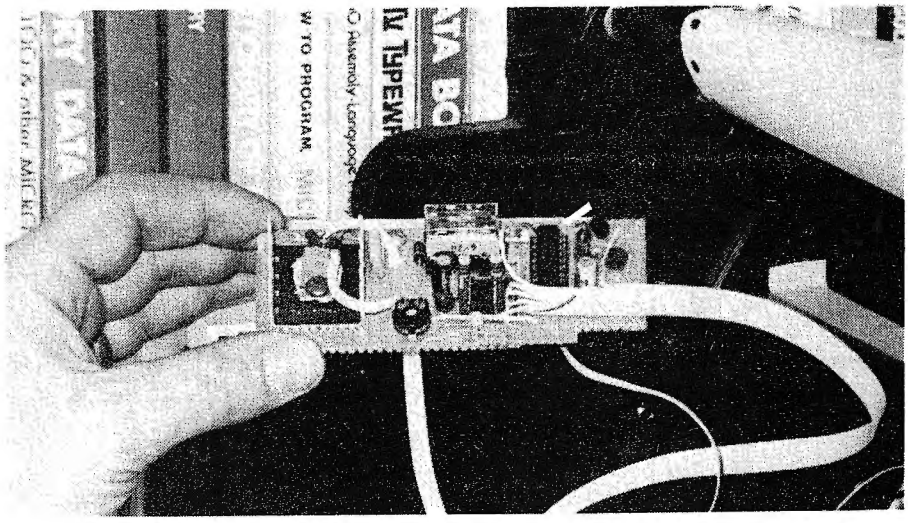
This all sounds good. But one small problem does exist. If the circuit in Figure 3 is to be located in the printer itself (and it should be) we will need to pick up the read pulse from within the printer. But this read pulse is not wired to leave the expansion interface unit, so it does not appear in the printer. This can be overcome by connecting it ourselves - to one of the unused bus leads between the expansion interface and the printer.

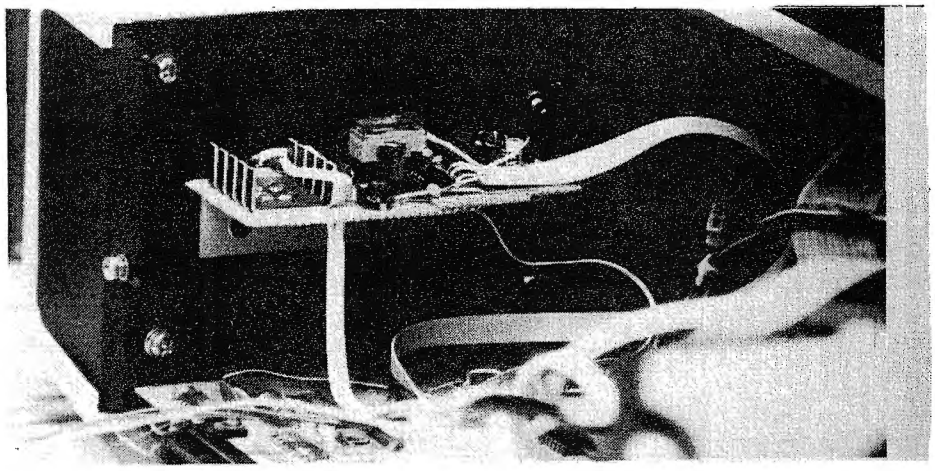
Figure 2 shows the physical location of this connection (strap). The read pulse is obtained from pin 1 of Z46 and connected to the vacant bus lead #32 within the expansion interface. This sends the read pulse through the ribbon cable and allows it to be picked up by the circuit of Figure 1 from within the printer on bus lead #34. The reason the bus lead numbers do not appear the same at both ends is because the two different producers of this equipment counted from different ends.

Figure 3 shows the location of this read pulse on the printer bus. Also shown is the connection of Q3 to bus lead 32, which is one of the two status leads that were previously unused. Figure 3 also shows the wire connected from pin 8 of ME 24 going to the input of IC1 pin 2 (in Figure 1). This is what shuts off the motor during an out-of-paper or manual busy condition.

Operation

After the project has been completed and installed, the printer can be plugged into an AC outlet and the power switch kept on. I leave mine connected to the same power outlet as the computer, so whenever the computer is turned on the printer receives power too. The initial powerup causes a spike in timer "A" which does activate the motor, but it shuts off again within two seconds (the setting I normally keep time "A" at).





Photos of the circuit board and the author's Line Printer I

Parts List

IDENTITY	TYPE	DESCRIPTION
IC1	74LS32	Quad "OR" gate
IC2	556	Dual 555 timer
Q1, Q2	2N3906	PNP General purpose transistor
Q3	2N2222	NPN General purpose transistor
Q4	Triac	400 Volt (RS 276-1000)
R1,R3,R5	10K ohms	1/4 watt resistors
R2	500K	1/8 watt mini PC Potentiometer
R4	100K ohms	1/4 watt resistor
R6,R7	150 ohms	1/4 watt resistor
C1	20 uf	12 volt electrolytic capacitor
C2,C5,C6	.01 uf	Tantalum capacitors
C3	4.7 uf	12 volt electrolytic capacitor
C4	.1 uf	200 volt metalized-film capacitors
D1,D2	1N914	50 volt 100 ma diode
RL1	5 volt DIP relay	Fits in an IC socket
Heat sink		for Q4, the Triac
Twin lead	connectors	RS part 274-222

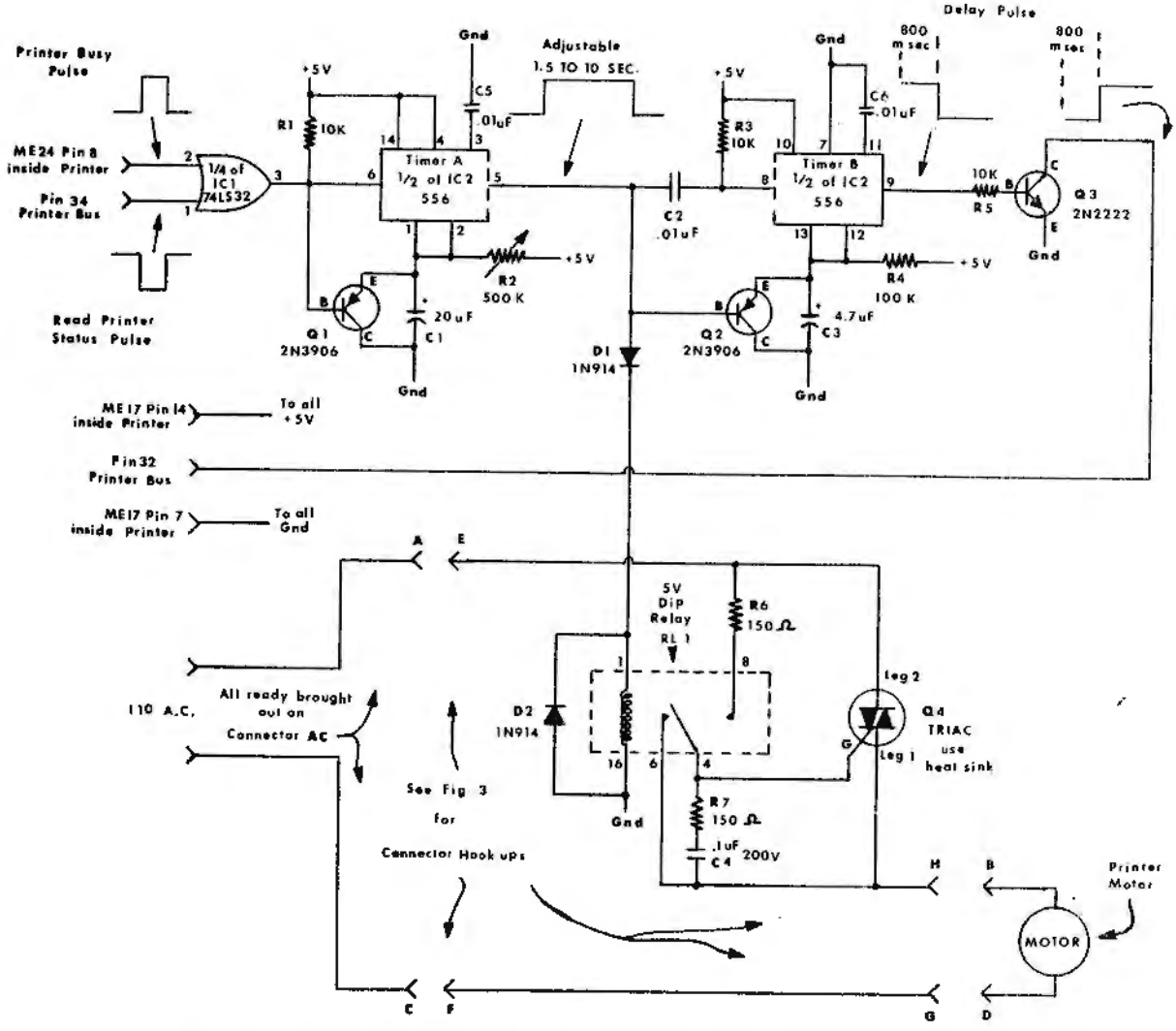


FIGURE #1

This circuitry automatically turns the 779 Printer motor on whenever the computer (S-80) wants to send data to the printer. It effectively tells the computer to wait 800 milliseconds for the motor to build up RPM. Then if the computer stops sending data to the printer for more than 1.5 seconds (adjustable to 10 sec.) the NOISY printer motor will shut off automatically.

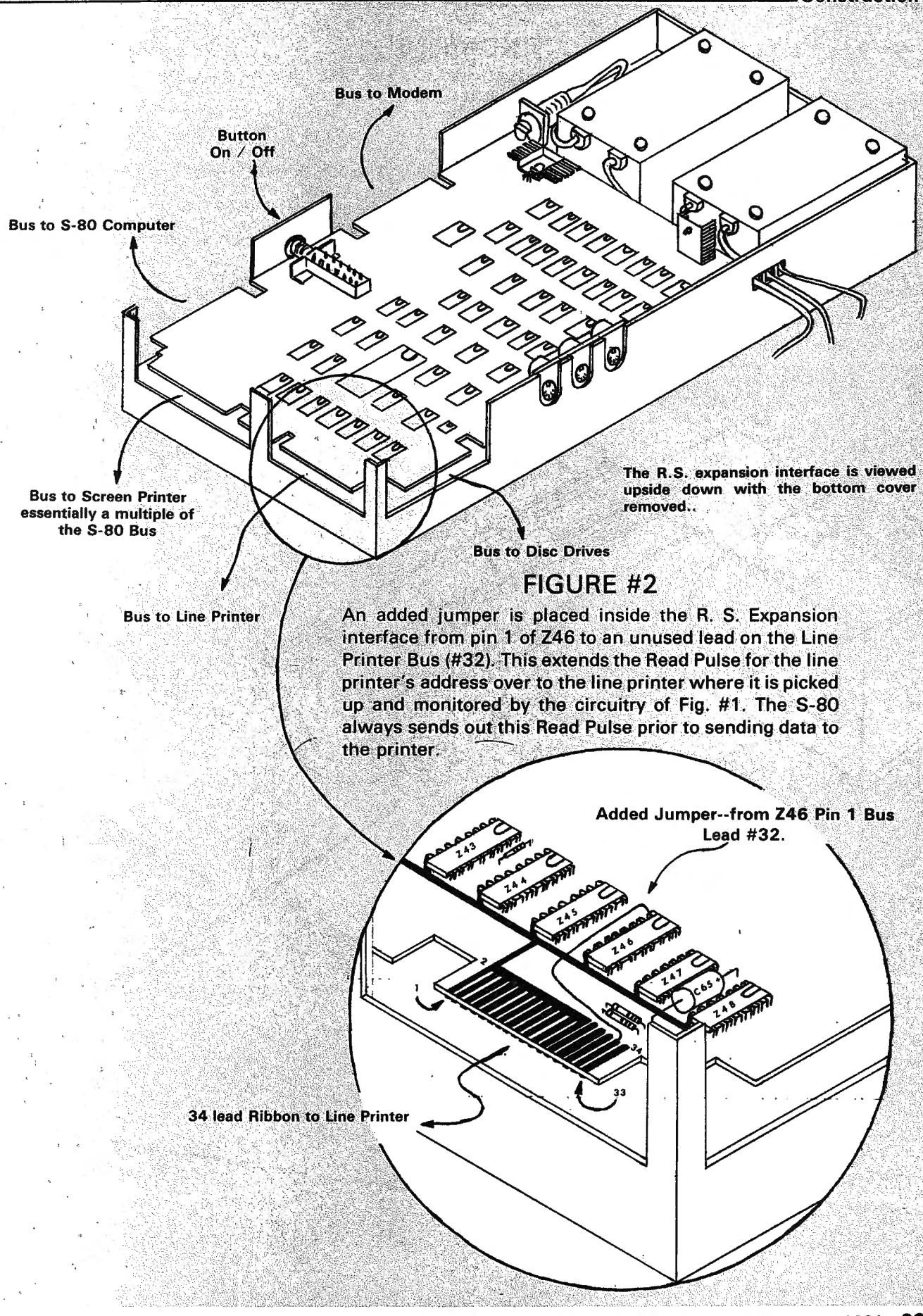


FIGURE #2

An added jumper is placed inside the R. S. Expansion interface from pin 1 of Z46 to an unused lead on the Line Printer Bus (#32). This extends the Read Pulse for the line printer's address over to the line printer where it is picked up and monitored by the circuitry of Fig. #1. The S-80 always sends out this Read Pulse prior to sending data to the printer.

FIGURE #3

The circuitry of Fig #1 can be placed on a small circuit board and with right angle brackets, can be mounted inside the printer on the vacant side panel (see photos on page 37).

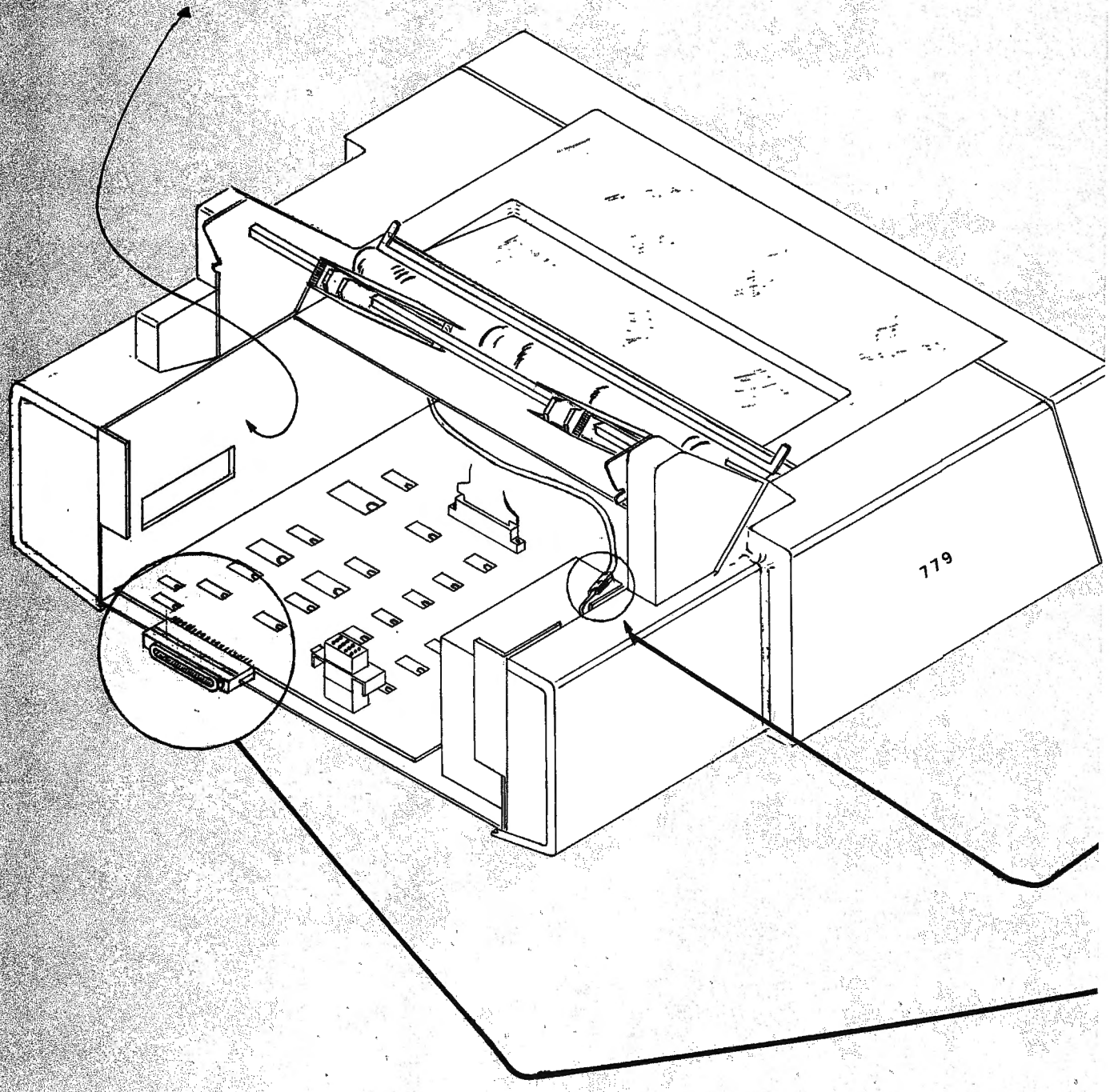
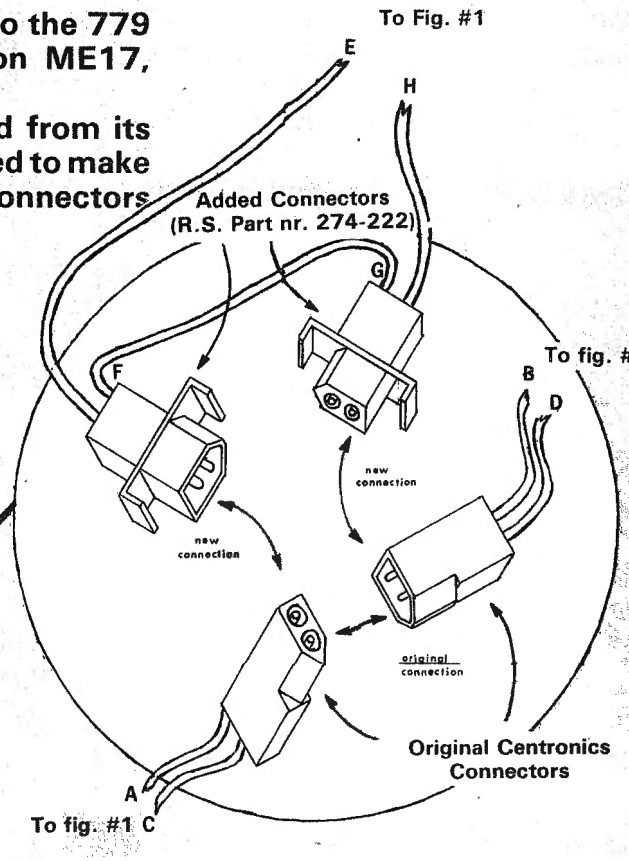
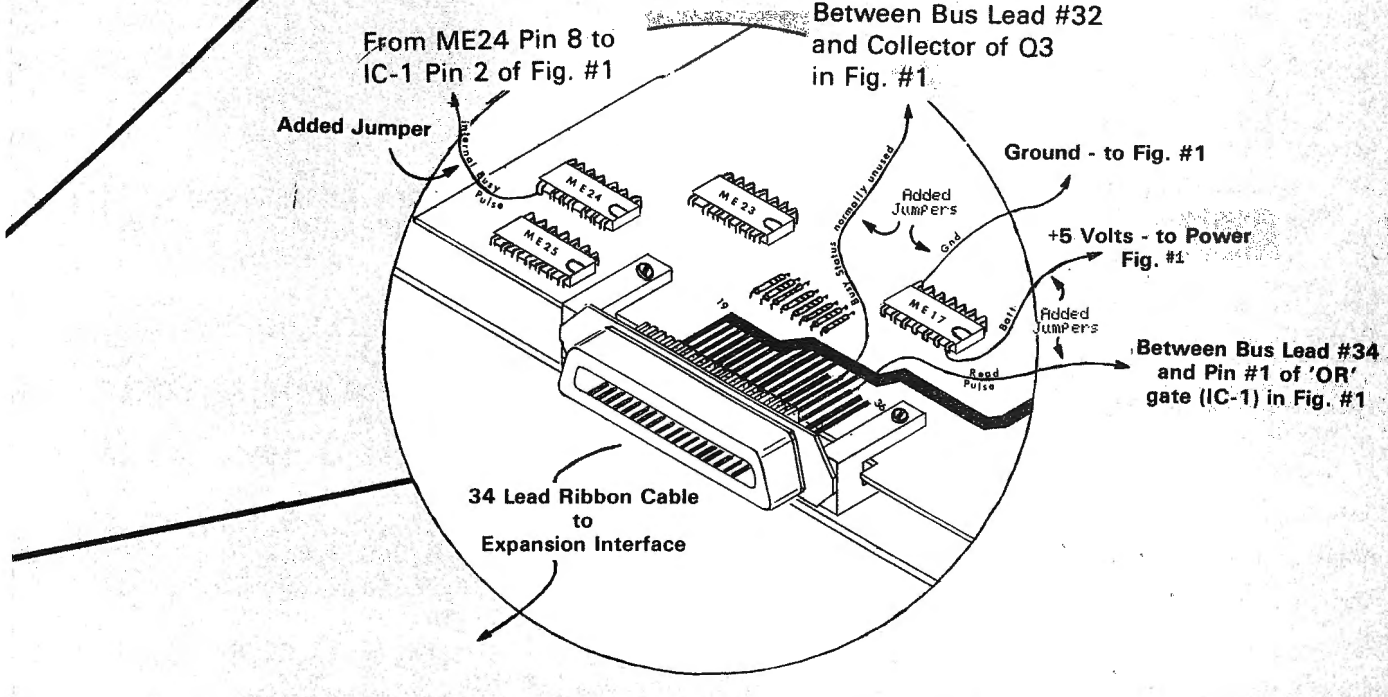
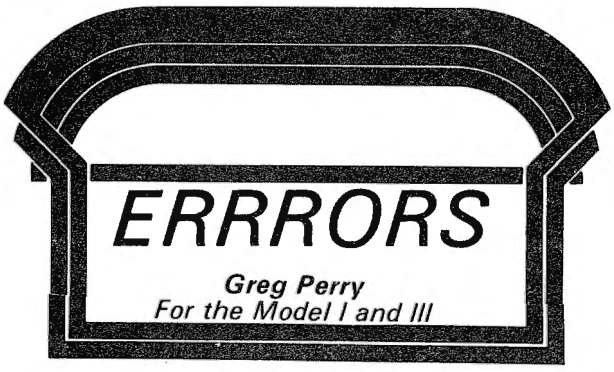


FIGURE #3

The circuitry of Fig. #1 is connected to the 779 printer via the five added jumpers on ME17, ME24, and the Printer Bus.
Then the motor connector is removed from its original connection and each part is used to make a new connection with the added connectors (R.S. part nr 274-222).
This allows the circuitry of Fig. #1 to gain control of the NOISY printer motor without cutting any wires.





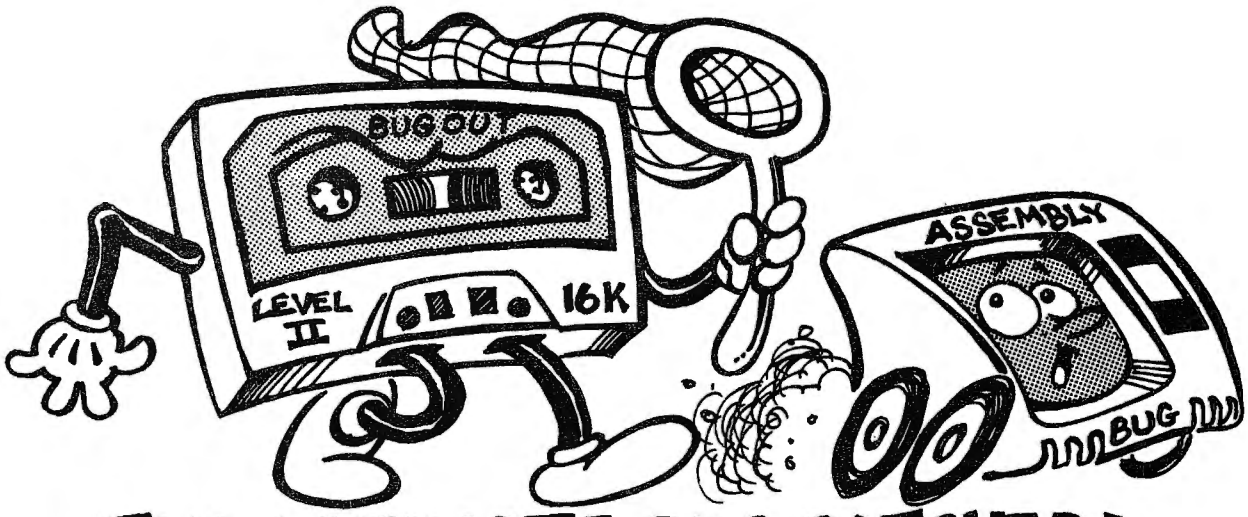


Dear Level II users-are you tired of the old drudgery of having to look up in your Level II manuals every time you get an error because the error messages supplied by the Level II ROM are not concise?

Wouldn't **you** like to have the specific, found-on-no-other-brand, detailed error messages that *only* appear on other versions of the TRS-80 Model I? Well, now you can! Enter the following program, (**Be sure to set memory size) and then generate a few errors by typing in garbage, or dividing by zero, or anything you can think of.

Now if you have tried this, feel free to clap your hands! You now have the detailed messages that only other models have!!

```
100 REM ** SET MEMORY SIZE TO 32650
110 C = 0
120 CLS
130 FOR I = 32664 TO 32767
140 READ N
150 POKE I,N
160 C=C+N
170 NEXT I
180 IF C<>7788 THEN PRINT"WARNING..DAT
  A IS NOT CORRECT":STOP
190 POKE 16806,195 : POKE 16807, 152 :
   POKE 16808, 127
200 DATA 123,33,209,127,6,6,190,202,18
  7,127,35,5,194,158
210 DATA 127,6,12,190,202,196,127,35,5
  ,194,169,127,33,245
220 DATA 127,205,167,40,195,25,26,33,2
  32,127,205,167,40
230 DATA 195,25,26,33,239,127,205,167,
  40,195,25,26,0,0,0,0,2
240 DATA 24,34,36,40,42,0,4,6,8,14,18,
  20,22,28,32,38
250 DATA 44,10,12,16,26,30,87,72,65,84
  ,63,11,0,72
260 DATA 79,87,63,11,0,83,79,82,82,89,
  11,0,0,0,0,0
270 END
```



THE ULTIMATE BUG CATCHER!

BUGOUT — A PROGRAM, TEXTBOOK, AND REVOLUTION!

THE BUGOUT PROGRAM is the most powerful Z-80 MONITOR/DISASSEMBLER/DEBUGGER and GENERAL PURPOSE UTILITY PACKAGE available! BUGOUT is so versatile, beginners can learn Assembly Language programming in record time. BUGOUT is so powerful, experts will refuse to program without it! Here are only a few of BUGOUT'S 90 individual functions.

- **Interpretive Traces**
- **Instant Breakpoints**
- **Address Breakpoints**
- **Register Breakpoints**
- **High Speed Traces**
- **ROM Analysis Routine**
- **1 Column Disassembly**
- **2 Column Disassembly**
- **Single Step in RAM**
- **Single Step in ROM**
- **Fast Patch Routine**
- **System FMT Tape I/O**
- **Base Program in 7K**
- **Bugout Relocator + 1K**
- **Video Save/Recall + 1K**
- **Extensive Memory Exam**

THE BUGOUT TEXTBOOK will answer questions other Assembly Language courses overlook, or cannot easily answer. There is no other course like it anywhere! The BUGOUT manual is written with the complete beginner in mind, yet provides interesting and valuable ideas for all programmers! Experts can easily skip what they already know and get down to important things....like squashing bugs in record time!

THE BUGOUT REVOLUTION, program and textbook combined, makes learning Assembly Language easy. New concepts are immediately brought to life on your video display! You can actually "see" how every machine code functions right before your eyes! You can enter BASIC language statements and single step trace them...all the way through the ROMS. You can link machine code routines to your BASIC programs more EASILY and EFFICIENTLY than you ever dreamed possible, and BUGOUT will remain a POWERFUL WORKING TOOL long after you've become an Assembly Language Expert!

HONEST TO BUGOUT — We couldn't begin to tell you everything about BUGOUT on one page, so we prepared a special information package for your inspection. Better yet, order BUGOUT today while it is still introductory priced at $24.95. We think this package is worth considerably more, and anticipate a price adjustment in the future. Order today and save...money, of course, but more important, your valuable time. BUGOUT will pay for itself immediately by saving you countless hours learning, writing, and debugging your own Assembly Language programs!.

RESERVE YOUR COPY NOW! BUGOUT BELONGS IN EVERY PROGRAM LIBRARY!

BUGOUT SWEEPSTAKES

Win $1,000 Cash

ENTRY DEADLINE — October 15, 1981

DRAWING — October 31, 1981

No purchase necessary to win.
Complete details provided to every entrant.

THE SOFTWARE PLANTATION, INC.,
P.O. Box 44623, Tacoma, WA 98444
(206) 531-1506

SWEEPSTAKES ENTRY FORM

Complete sweepstakes rules and procedures enclosed with BUGOUT or BUGOUT INFORMATION PACKAGE.
No purchase necessary. Void where prohibited by law.

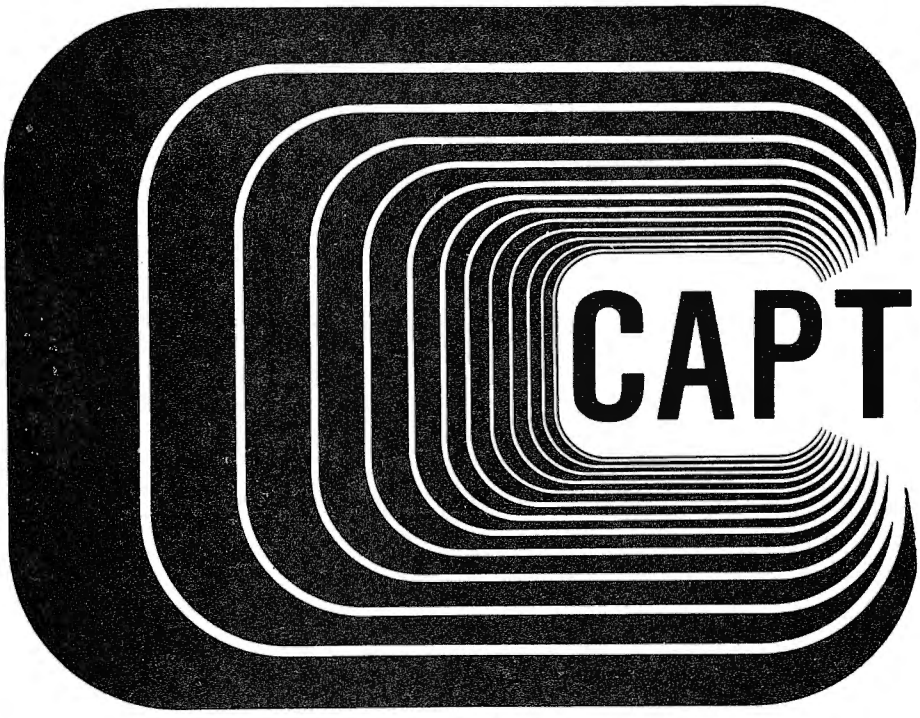
TO PURCHASE BUGOUT, complete and mail to:
THE SOFTWARE PLANTATION, INC.,
P.O. Box 44623, Tacoma, WA 98444

_____Check _____Money Order _____Master Card _____Visa
Card No. _____ Expiration date _____
Washington residents add 5.3% sales tax

_____ Please rush BUGOUT @ $24.95
_____ Please rush BUGOUT INFORMATION
_____ Please enter my name for the drawing

Name _____
Address _____
City _____ State _____ Zip _____

Circle 19



CAPTAIN 80™

Bob Liddil

Here's the Software Secret Agent traveling around the country at the speed of light. No, you did not read it wrong, - the speed of light. At the Boston Computer Show I broke down and bought a modem for the lab at Software Secret Headquarters. I'd seen a demonstration of something called CBBS, (Community Bulletin Board Service), accessable only by modem, which I was told was free.

After the modem arrived and the RS232 was installed I was informed that a terminal program was required. So I called up Instant Software, which is just down the street, to see if they had one for sale. They did. *Terminal 80*, as it is called, is Instant Software's entry into the semi-crowded smart terminal market. Checking in at an inexpensive $29.95, it filled the bill for a beginner on the wires.

The first attempt at telecommunications was an unqualified disaster. Absolutely nothing worked! We called CBBS Boston, which is one of the pioneer BBS's in the country, absolutely to no avail. In desperation, I called a local Tele-expert and explained my problems.

When his fits of uncontrolled laughter subsided I was smugly told that pressing ENTER two or three times after the phone was in the modem would likely solve the problem. Smarting from the sting of an obvious solution to what seemed a hopeless situation, I returned to the modem.

This time, after pressing ENTER the appropriate number of times, up popped the BBS. I signed on as Captain Eighty, which must have had the SYSOP, (the guy who operates the board), scratching his head. Immediately I started searching through the wonderland of messages, personal and public, which make up the bulk of the BBS.

So far, Terminal 80 had done a swell job for me. It is, for the price, a good investment for the beginner. Then I tried dialing ABBS, Cincinnati: ABBS (I found out later) is short for APPLE Bulletin Board Service, which may account for the fact that as soon as I put the phone into the modem, the screen disintegrated into garbage and gave up the ghost. I was lost.

CBBS Boston has a special feature which lists the phone numbers of BBS's around the country. Forum 80, I discovered, is a TRS-80 based BBS. Plus, there are very many of them around. So for a while, I was content with an occasional foray onto a new CBBS or Forum 80.

Enter *OMNITERM*. This new and relatively unadvertised smart terminal program was like a space ship is to a hay wagon when compared to Terminal 80, but at three times the retail price, it had better be. When confronted with the awesome complexity of Omniterm, I decided to break it down into manageable chunks. First, how did it work as a simple smart terminal? I tried it out on the Forum 80 in Nashua, New Hampshire, and it worked just fine. But when I tried ABBS Cincinnati, it fell apart, just as had Terminal 80. I was doing something wrong.

A quick call to Dave Lindbergh, Omniterm's author, revealed the necessity of resetting some of the configurations of the RS232. This can be accomplished from the keyboard, in Omniterm, a feature which occupies only a small percentage of the on-screen menu. Back to the ABBS. BINGO! It worked like a charm. I was linked by computer/telephone to one of the best and friendliest computer retail centers I have ever encountered by phone or otherwise.

The Cincinnati Computer Store, in the Princeton Mall Shopping Center, outside the Queen City, received a Software Secret Agent visit over Christmas, where I found their staff knowledgeable, courteous, and as enthusiastic about the TRS-80 as the Apple. They were incredibly well stocked and their store display area was the best design I've seen, pleasant and comfortable. I did not know they maintained a BBS, so I was surprised to find them the operators of a long sought and elusive contact.

Since it is possible to access the printer from the on-board menu in Omniterm, I decided to switch to hard copy for this BBS contact. I listed out the messages and recorded the numbers for BBS's, and even left a missing person message concerning a certain disappeared *Professor*

Megabyte, a subject which may be discussed in later columns. It was very satisfying to watch my computer humming along, doing exactly what it was designed to do.

BBSing can be a very expensive hobby if the contact is long distance, as is everything from where I live. But the entertainment it generates and the friends you can make potentially outweigh the expense. For example, by dialing (305) 862-6917 evenings or weekends, you can receive from, or leave a message for Scott Adams, the Godfather of Adventure. A call to (313) 465-9531, Mount Clemens, Michigan, you will get a Medical Forum 80 with messages of local interest, such as a printer for sale, or messages of a decidedly more national flavor such as Dave Goldman's warning about a Model II Scripsit bug.

The number (303) 399-8858 is the home of Log/On Magazine, (now a newsletter, but we know how fast that can change). After 5 PM you will encounter, of course, the standard Forum 80. They have a lot of neat things planned for the near future. The magazine itself is about to be born, and will review BBS's and their features. Tele-computerists or interested parties can get more information about this specialty sheet by writing Log/On, 1405 Krameria Office 3C, Denver, CO 80220, or call the above number in the daytime.

The number (313) 294-8248 is the home of Download 80, a BBS specializing in the tele-transfer of programs from the host computer to the user. This system must have a waiting list because in 32 tries over a three day period, I couldn't get past a busy signal.

For those with unlimited budgets, there is a Forum 80 in Hull, England. You can dial it direct, 011-44-482-859169 if your pocketbook can stand the overseas call. It would be worth it in my opinion, just to log on and solicit a couple of Pen Pals.

Another interesting BBS feature now coming into popularity, is the Computer Shopper, or spending money over the keyboard. Forum 80 headquarters has a good selection of modems for sale, as well as a wide selection of tele-communications accessories and Smart Terminal programs, (no Terminal 80 or Omniterm yet). Just give the computer your charge card number and poof! UPS delivers your purchase.

This same BBS, which promoted its own wares, gave equal time to the promotion of a nearby store. The Software Shack, Belton, Missouri also had an on-line catalog, with such goodies as Duel-N-Droids (by Leo Christopherson and produced by Acorn Software), Hayden's Sargon II and Adventure International's Lunar Lander. I have no doubt that it is a well stocked store.

This is a surface scratch of the incredible world of lightspeed travel. You don't have to be a Software Secret Agent to do it either. Simply get a good modem, a good smart terminal program, an RS232 in your expansion interface and....chances are, there is at least one BBS somewhere very close to you, or very far away and as close as your phone.

Here's the Software Secret Agent, at the pawn shop, pawning his watch that squirts tear gas, belt that turns into a chain link fence, and other Secret Agent goodies. I owe the phone company, and that's *one* foreign country nobody messes with! ●

Sketch & Pack

Paul Gerhardt
Bethel, Connecticut

Programs which rely heavily on animated graphics can be fascinating to run but terribly time consuming to write. Storing the graphics information can also use up large amounts of memory. To save memory Leo Christopherson's string packing technique, as described in the May/Jun 1979 issue of *80-U.S. Journal* is used. This is a slow and tedious process. First the frame must be drawn on a video work sheet. A frame is the drawing you wish to place on the screen enclosed in an imaginary rectangle. Then the ASCII codes for each element of the frame must be determined. This information must be entered by hand into a DATA line to use Leo's string packing routine. This routine pokes the graphics data into a string variable that has been reserved for this purpose. Only after this is done for each line (the whole screen has 16 text lines) of every frame can you get down to actually planning the animation.

Many programs are thoroughly planned before they are written. But others evolve as they are written. They start simply but as more features are added they grow in complexity. This sketch program falls into this category.

The first program that was written to help in the development of graphics was a simple sketch program. Although it lets you see what the frame would look like before you spend time coding, this program didn't save any time. It did however serve as the foundation of Sketch and Pack.

We want the computer to give us the ASCII codes for each frame so that we won't have to look them up on a chart. So it was decided to add this feature to the sketch program. At first this seemed easy enough. We could simply peek at the video display memory (addresses 15360-16383), and send that information to the printer. But we didn't want the whole screen peeked, just the addresses of the frame. So a method to define the perimeter of the frame was developed. This was done by the placing of corner markers. This worked out well, but if the TRS-80 could determine the ASCII codes, why not have it do the string packing while it was at it? This would free us to just be creative without worrying about how long it was going to take to code all those fantastic graphics. But could it be done?

The first question was: Could a subscripted string variable be packed? Yes. Now the computer could control the packing of a series of string variables. If we plan ahead so we know the number and size of the variables that would be needed, we could add them to the program before it was run. The program would then pack them for us. Terrific, right? No, not quite..

We don't want to have to plan ahead. We want to just draw. Let the computer look at the drawing and tell us how many variables we need and how long they should be. The routine was worked out to do this, but there was another problem. After the computer gave us this information, we would have to break out of the program and add these variables to the program. But when this was done, our beautiful drawing would go scrolling off the screen. There was nothing left to pack! Some way of protecting the masterpiece was needed. The solution turned out to be simple. Just before leaving the program to add the needed variables the contents of the video memory were poked into protected high memory. Then, after all the changes were made, the program was run again, and the video memory is restored. Ah, success!

Using the Program

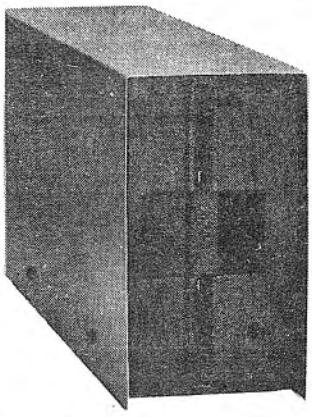
To use this program you must first set memory size to 31740. You can then load Sketch & Pack. Run the program and you will be asked if you want to restore the screen from memory. Since you do not yet have anything stored in high memory, answer no. You will now see a small blinking cursor in the upper left hand corner. You control the cursor with the four arrow keys. You can draw a diagonal line by pressing two arrow keys together. While drawing, rest your right thumb on the "?" key. To erase or move the cursor without drawing just hold down the "?" key. Don't worry about drawing off the screen. You now have full wrap-around.

Once your masterpiece is complete press ENTER and you will see a large blinking cursor in the upper left hand corner. This cursor is used to place the corner markers. It is also controlled by the arrow keys. *WARNING: Do not move this large cursor over your drawing. It will erase it!* When the cursor is in position to mark the first corner press the space bar, and an arrow will mark the spot. **The corners must be marked in this order: 1. Upper left, 2. Upper right, 3. Lower left and 4. Lower right**. Limited checking is done to see if the markers are in the proper position. If you make a mistake just scatter the rest of the markers around the screen. You will get an error message and a new cursor in the upper left hand corner. Set the markers again.

Once the corner markers are in place you will be given a choice. Press

(listing starts on page 49)
(text continues on page 50)

Mini-Disk Storage Systems for TRS-80* Computers



Access... Offers More. Charges less.

from $295

Check this line-by-line comparison of Access 5-inch mini-disk systems for Model I and Model III computers. *Nobody* offers more benefits, better service or lower prices.

TOLL-FREE ORDER NO.
1-800-527-4196†
(orders and literature only)

Company/Drive Model	Price 1/	Flippy	*Formatted* Diskette Storage Capacity Dbl-Density	Sgl-Density	100% Tested	48 hr. Burn-In	Warranty	Trial Period
40-TRACK DRIVES								
Access Unlimited								
AFD-100[†]	$295 00	no	180 Kbytes	102 Kbytes	yes	yes	yes	yes
AFD-100F[†]	329 00	yes	360 Kbytes	204 Kbytes	yes	yes	yes	yes
MTI								
TF-5	359 00	no	?	?	?	?	?	no
Midwest Comp & Per								
MPI B-51	321 00	no	?	102 Kbytes	?	?	yes	no
Aerocomp								
Mdl 40-1	349 95	yes	?	?	yes	?	yes	yes
CPU Shop								
CCI-100	314 00	no	?	102 Kbytes	?	?	yes	no
AMI								
40-track	325 00	no	?	?	?	?	?	no
80-TRACK DRIVES								
Access Unlimited								
AFD-200[†]	429 95	no	368 Kbytes	205 Kbytes	yes	yes	yes	yes
AFD-200F[†]	449 95	yes	736 Kbytes	410 Kbytes	yes	yes	yes	yes
MTI								
TF-8	639 00	no	?	200 Kbytes	?	?	?	no
Aerocomp								
80-tk mdl	459 95	yes	?	?	yes	?	yes	yes
CPU Shop								
CCI-280	429 00	no	?	204 Kbytes	?	?	yes	no
AMI								
80-track	560 00	no	?	?	?	?	?	no

1 As advertised in 80 Microcomputing Jan 1981

Free trial offer

Use your brand new AFD drive system for up to 15 days. If you're not *completely* satisfied, pack it in the original shipping container and send it back to Access. We'll refund the price of the system (less shipping charges), no questions asked. (No refund for misuse or improper handling.)

Operating burn-in test — too

Every drive that leaves Access is not only 100% electrically tested and double-checked for mechanical alignment, but it is also given a full 48-hour operating burn-in test. You'll find a test list — checked and signed by one of our competent technicians — in the drive carton. If a drive has latent defects, the burn-in life test will weed them out. The drives we ship just keep on running. And running. And running.

About our warranty

Venus de Milo has about the right number of fingers to count our warranty returns. Nevertheless, your new drive system is covered by our comprehensive 90-day limited warranty. The details are spelled out in the illustrated users manual included with each AFD drive system.

Software galore!

Games. Business programs. System software. We sell many of the best TRS-80* Model I and Model III programs, and at competitive prices. Make Access Unlimited your one-stop shopping center for all of your TRS-80* software, hardware and accessories. Save big! Call our toll-free order number, 1-800-527-4196† for free descriptive literature.

USE YOUR CREDIT CARD AND SAVE! VISA AND MASTER CARD CHARGES ARE NOT DEPOSITED UNTIL THE DAY YOUR ORDER IS SHIPPED. CALL NOW TOLL-FREE, 1-800-527-4196†.

the DOUBLER™

Percom's new plug-in adapter for your Expansion Interface stores almost twice the data on a diskette track as a single-density system. You can store up to four times more data — depending on the type of drive — on one side of a diskette than you can store using a standard Model I mini-disk drive. Other features: Reads, writes and formats *either* single or double density minidiskettes • Runs TRSDOS*, NEWDOS+‡ Percom OS-80™ or other single-density software *without* changing either software or hardware. Switch to double-density when convenient. • Includes DBLDOS™, a TRSDOS* compatible double-density operating system • Includes on card, high-performance data separator circuit. • Installs *without* rewiring or trace cutting. • Introductory price, including DBLDOS and format conversion utility, only $219.95.

Permits Model III software to be read on Model I computers.

DATA SEPARATOR™

This PC board plug-in adapter for the TRS-80* virtually eliminates data read errors (CRC error — Track locked out!) which occur on high-density inner disk tracks, a problem that has plagued TRS-80* systems. The Percom Data Separator™ is installed in the Expansion Interface without modifying the host system. Caution: Opening the TRS-80* Expansion Interface may void the limited 90-day warranty: $29.95.

Disk System Interconnecting Cables

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

Two-Drive Cable $24.95
Four-Drive Cable 34.95

Power Line Filter

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

Minidiskettes (Double-Density rated)

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

Disk Drive ID Tabs

1" x 1¼" self-adhering plastic drive identification tabs. Compatible silver with engraved black drive number. Two tabs (Nos. 0, 1): $2.50; three tabs (Nos. 0, 1, 2): $3.25; four tabs (0, 1, 2, 3): $4.50.

How to Order Order by calling Access Unlimited toll-free on 1-800-527-4196†. Mail orders also accepted. Orders may be charged to a VISA or MasterCard account or paid by a cashier's check, certified check or money order. We accept COD orders with 25% deposit. Sorry, we cannot accept personal checks. We pay shipping and insurance charges on orders over $1,000.00. Add approximate insurance and shipping charges for under $1,000.00. If in doubt about these charges, ask when you call in your order. Texas residents include 5% sales tax. Minimum order $20.00. Allow 2 to 4 weeks for delivery. †Texas residents call (214) 494-0206.

ACCESS UNLIMITED
315 N. Shiloh · Ste. D1 · Garland, TX 75042
(214) 494-0206

T trademark of Access Unlimited Company. *RADIO SHACK and TRS-80 are trademarks of Tandy Corporation. ™trademark of Percom Data Company, Inc. ‡trademark of Apparat Corporation.
ALL PRICES AND SPECIFICATIONS SUBJECT TO CHANGE AND ALL OFFERS SUBJECT TO WITHDRAWAL WITHOUT NOTICE.

HOMES for the TRS-80

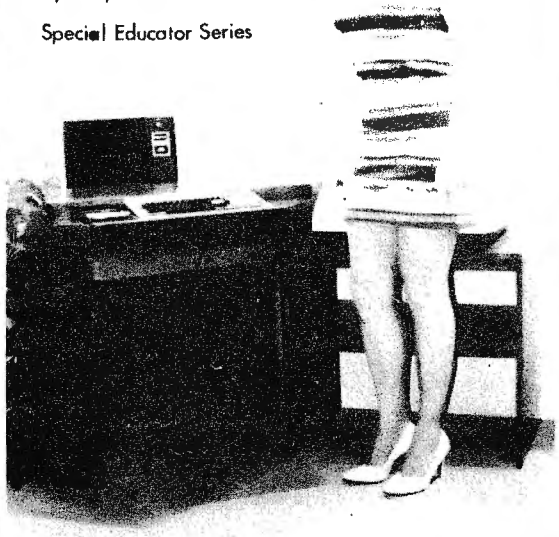
Featuring

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 homogenious 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, carrols, 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 availabe 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.

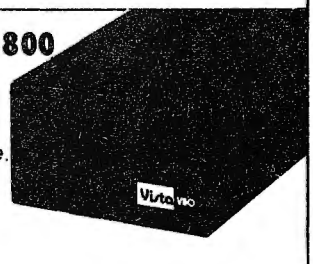
Miscellaneous Accessories: For our customer convenience, we provide a series of AC outlet strips, line filters, and cooling fans.

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.*

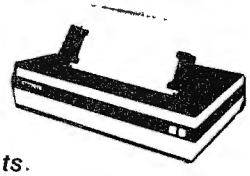
List $395.00

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 Charactor 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 Legibilty.*
- *Decreases Glare.*
- *System Looks Prof*

List $7.95
(Add $1.00 for shipping & handling, and California residents add 6% sales tax)

Miscellaneous System ACCESSORIES

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

Circle 24

(Listing for Sketch and Pack - cont'd from text on page 46)

```
10 CLEAR 500:DEFINT A,B,C,D,E,M,N,S,X,Y: DEFSTR F,G: CLS
12 DIM E(63,12)
15 INPUT"DO YOU WISH TO RESTORE THE SCREEN FROM MEMORY";V$
17 IF LEFT$(V$,1)<>"Y" AND LEFT$(V$,1)<>"N" THEN 15
18 IF LEFT$(V$,1)="Y"THEN GOSUB 3000: GOTO 20
19 CLS
20       ADD VARIABLES ON LINES 30-500
505 ON ERROR GOTO 680
510 X=0: Y=0
520 A=PEEK(14400): B=PEEK(14368)
530 PRINT@768,STRING$(63,"-");
540 IF A=8  THEN Y=Y-1 ELSE IF A=40 THEN Y=Y-1: X=X-1
550 IF A=16 THEN Y=Y+1 ELSE IF A=72 THEN Y=Y-1: X=X+1
560 IF A=32 THEN X=X-1 ELSE IF A=48 THEN Y=Y+1: X=X-1
570 IF A=64 THEN X=X+1 ELSE IF A=80 THEN Y=Y+1: X=X+1
580 IF A=2 THEN CLS
590 IF A=1 THEN 740
600 IF Y>35 THEN Y=0
610 RESET(X,Y)
620 FOR N=1 TO 6: NEXT N
630 SET(X,Y)
640 FOR N=1 TO 6: NEXT N
650 IF B<>128 GOTO 520
660 RESET(X,Y)
670 GOTO 520
680 IF X<0 THEN X=127: ' Error routine which provides screen
690 IF X>127 THEN X=0:  ' wrap arround.
700 IF Y<0 THEN Y=35
710 IF Y>47 THEN Y=0
720 O=0
730 RESUME NEXT
740 O=0: C=0
750 A=PEEK(14400): B=PEEK(14368)
760 IF A=8  THEN O=O-64
770 IF A=16 THEN O=O+64
780 IF A=32 THEN O=O-1
790 IF A=64 THEN O=O+1
800 IF A=2 THEN 10
810 IF O>704 THEN O=0
820 PRINT@O,CHR$(191);
830 IF A=128 THEN PRINT@O,CHR$(91);: GOTO 870
840 FOR N=1 TO 10: NEXT N
850 PRINT@O,CHR$(128);
860 GOTO 750
870 A(C)=O+15360
880 C=C+1: O=O+1
890 IF C=4 GOTO 910
900 GOTO 850
910 IF A(2)-A(0)=A(3)-A(1) AND A(1)-A(0)=A(3)-A(2) GOTO 930
920 PRINT@960,"CORNER MARKERS INCORRECT                ";:
    GOTO 740
930 IF A(0)+A(3)<>A(1)+A(2) THEN 920
940 L=A(1)-A(0): L1=(A(2)-A(0))/64: R=0
950 PRINT@832,"PRESS P FOR PACKING      PRESS S TO SET VARIA
  BLES";
960 IF INKEY$="S" THEN 1130
970 IF INKEY$="P" THEN 990
980 GOTO 960
990 FOR Z=A(0) TO A(2) STEP 64
1000 R=R+1: D=0
```

"P" for packing or "S" to set the variables. Well, you can't pack variables you don't have, so press "S". The computer will then tell you how many variables you need and how long they must be. Press BREAK and add these variables to your program. Don't worry, your masterpiece will be safe. It is stored in high memory. Variables "F" and "G" have already been defined as string variables so you might as well use them. More can be added if needed. Now look at line 1080. You will see the subscripted variable G(R) buried in there twice. If a different variable is used you must also replace G(R) with that new variable in this line. But since "G" is already there let's use it. Suppose the computer told you that you needed three variables, each of which should be 5 spaces long. On any line between 20 and 500 you would add the following:

```
30 G(1)="     "
40 G(2)="     "
50 G(3)="     "
```

The above three variables have been set equal to five spaces. When the proper variables have been added and line 1080 has been corrected to reflect the variable name used, type RUN. You will again be asked if you want to restore the screen. This time you do, so answer yes.

You will find yourself back in the drawing mode. You can make some last minute changes or simply press ENTER and mark the corners again. You must mark the corners again because the computer lost these values when you added lines to the program. When finished you will again be given a choice. This time press "P" to pack the variables. You will see the ASCII codes as they are poked into the string variables. After the packing is complete, you will be back in the drawing mode. If you wish, you may now change your drawing and create another frame. Remember that the previous frame is always held in high memory until the next frame is stored. It can always be recalled by simply running the program and answering yes to the restore question. When you are finished creating all the graphics you will need for your program, break out of it. You can now DELETE all of the program except for your newly packed string variables. Now build your new program around them or merge them with an existing program. There you have it, an easy route to string packing and character string graphics. ●

```
1010      FOR S=Z TO Z+L
1020      D=D+1
1030      H=PEEK(S): IF H=32 OR H=91 THEN H=128
1040      E(D,R)=H
1050      NEXT S
1060      GOSUB 1080
1070 NEXT Z
1075 GOTO 20
1080 T=PEEK(VARPTR(G(R))+2)*256+PEEK(VARPTR(G(R))+1)
1085 PRINT@832,STRING$(64,128);
1090 FOR D=1 TO L+1
1095 PRINT@832,"PACKING VARIABLE #";R,E(D,R);
1100 POKE T+D-1,E(D,R)
1110 NEXT D
1120 RETURN
1130 ME=31740:
     PRINT@896,"OK, WAIT JUST A MINUTE I'M WORKING";
1132 FOR X=15360 TO 16191
1134 POKE ME,PEEK(X)
1136 ME=ME+1
1138 NEXT X
1140 CLS
1150 PRINT"FOR THIS FRAME YOU WILL NEED";L1+1;" VARIABLES"
1160 PRINT"EACH VARIABLE WILL REQUIRE";L+1;" SPACES"
1170 PRINT"AFTER ADDING THE THESE VARIABLES AND INSERTING TH
  E VARIABLE"
1175 PRINT"NAME IN LINE 1080 TYPE RUN TO RESTART"
1180 'END
3000 ME=31740
3002 FOR X=15360 TO 16191
3004 POKE X,PEEK(ME)
3006 ME=ME+1
3008 NEXT X
3010 RETURN
```

10-12	Initialization
15-18	Lets you restore the video screen
500	Sets up error trapping routine for the screen wrap around
510	Returns cursor to home
520	Looks at keyboard memory; "A" for arrow keys, CLEAR, and space bar. "B" for "?" key.
540-670	Sketch routine
680-730	Error routine for screen wrap around
740-900	Routine for setting corner markers
910-930	Checks for proper placement of corner markers
940	Determines size of the frame
950-980	INKEY$ routine which gives the choice: Packing or setting of variables
990-1070	FOR NEXT loop that reads the frame's graphics data. Line 1030 does the peeking and changes spaces (32) and the corner markers (91) into blanks (128). Line 1040 temporarily stores the graphics data in array E(D,R)
1075	Return to drawing mode
1080-1120	String packing routine
1130-1138	Stores the contents of the video display memory in protected high memory
1140-1180	Prints the information needed to add the proper variable to the program
3000-3010	Subroutine that restores the video display memory

Fastload

FOR TRS-80* MODEL I USERS ONLY

16 Times Normal Speed

*TRS-80 is a trademark of Tandy Corp.

- **High speed load TRS-80* Level II cassettes**
- **Input 15K byte Level II program in 15 seconds**
- **Search BASIC or SYSTEM programs by name**

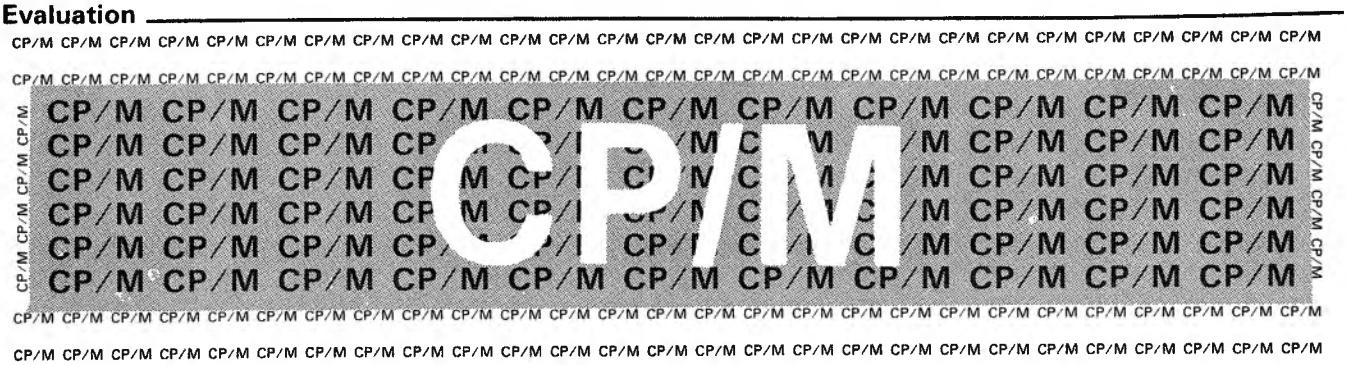
Unlike other high speed tape input devices, FASTLOAD uses standard format cassettes. Therefore, there is no need to re-record on other media. At 8000 baud, FASTLOAD is faster than disk for short programs. FASTLOAD reads tapes at the fast-forward speed of the CTR-41 cassette recorder. The recorder can also be used for CSAVE at the normal speed.

FASTLOAD connects to the 40 pin I/O or to the Expansion box. The control program does not use computer memory because it is in a built-in PROM. Other valuable features are keyboard debounce program, automatic key repeat routine and key-beep via cassette speaker. Price is $188.00 for FASTLOAD and $95.00 for the modified CTR-41 recorder.

Personal Micro Computers Inc.

475 Ellis Street, Mountain View, CA 94043 **(415) 962-0220**

Circle 26



Last year we took a look at CP/M for the Model I computer with the idea in mind that it was an alternate system for the serious user. Since it was first developed by Gary Kildall in the early 1970's, CP/M has grown from a relatively unnoticed operating system to one of today's most widely used systems for microcomputers.

On the Model I however, we have a real problem with CP/M implementation due to the ever present ROM BASIC Interpreter. CP/M is designed to operate without the ROM and CP/M programs expect to find important references there. This leads to basic incompatability in the Model I CP/M and the "standard" implementation.

The TRS-80 Model II system is constructed differently. With the Model II, we can load a standard CP/M system and execute nearly *any* CP/M program that comes on 8" disks. In the course of this evaluation, several commercially available programs were tried. They were not specifically for the Model II, yet they all worked satisfactorily.

What is CP/M?

Anyone who gets a Model II immediately gets a Tandy Radio Shack Disk Operating System (TRSDOS) with it. The only way to make the system do *anything* is to put the TRSDOS disk that came with the computer into the drive, turn the Model II on, and get to TRSDOS READY. If this is all you know about Disk Operating Systems, you are not alone.

A DOS will handle all of the dirty details for us like keeping track of where our information and programs are stored, communicating with the various devices we are running with the computer, and making it possible for languages such as BASIC to do their work. In short, they are like a translator for us. We speak BASIC, but the computer only speaks in Machine Language. The DOS allows the two to talk to each other successfully.

(OK ... some of you may think that is oversimplified, but we don't need to make *everyone* into a Systems Programmer!)

Like TRSDOS, CP/M provides an *Environment* that makes it easy to communicate with the computer. In this environment, we can use different high-level languages such as BASIC, PASCAL, FORTRAN, COBOL, C, FORTH and many others. We also get the advantage of a standardized system that lots of things will work with.

Also, CP/M (like TRSDOS) provides us with a convenient set of *Utilities* to help us with our processing. These programs handle jobs like file copying, disk backup, system reconfiguration and much more.

Because of its popularity, an amazing number of programs have been written to work with CP/M. Many of these will run on the TRS-80 Model II without change. This makes CP/M a serious contender for your use. But, when we start looking, we find that there is more than one CP/M for the Model II. Each has the same basic principles of operation, but each also has some unique features that the others do not have.

You could probably go to Digital Research (the people who developed CP/M) to get a system. But to use it you would find that you have to do a lot of tailoring and a large amount of assembly language programming. If this isn't what you want, then you want one of the CP/M's we are covering here.

Here, we will look at what CP/M includes as standard features and what the CP/M's advertised for the Model II have available. We will draw some comparisions and try to give you the information you need to help select a system for your own use. This information is based on CP/M systems obtained from each supplier during 1980. Revisions of the systems after September 1980 have not been taken into account. Further, pricing information quoted here may not be current at the time of publication.

We contacted the four suppliers of CP/M for the TRS-80 (FMG Corp, Cybernetics, Lifeboat Associates and Pickles & Trout) and asked for the opportunity to evaluate their CP/M against the others. Three suppliers (Cybernetics, Lifeboat and Pickles & Trout) were not only cooperative, but actually seemed anxious to show us how good their systems were. One supplier (FMG Corp) promised a system, but it never materialized. For this reason, we will be covering only three systems for the remainder of this report, since we cannot evaluate an unseen system.

Standard CP/M

Standard CP/M has two general types of commands. The built-in commands and the *Transient Commands*. Built-in commands are part of the *Console Command Processor* (CCP). The standard commands are:

ERA - Erases files from a disk.

DIR - Lists the files in the disk directory.

REN - Renames files.

SAVE - Saves memory to a file.

TYPE - Types the contents of a file.

The Transient Commands typically provided on CP/M are:

STAT - Provides disk statistics on files, disk storage, and allows the user to alter input or output device assignments.

ASM - The CP/M assembler to load the assembler and assemble a specified file.

LOAD - Loads a file in HEX format and produces a machine executable form.

DDT - The CP/M dynamic debugger.

PIP - The peripheral interchange program which handles all types of file transfers.

ED - The system text editor, normally used for writing programs.

SYSGEN - Used to create new CP/M diskettes.

SUBMIT - Submits a file of commands for BATCH processing (like the DO command in TRSDOS).

DUMP - Dumps the contents of a file in HEX.

MOVCPM - Regenerates CP/M for a particular memory size.

TRS-80 Model II CP/M's

We were impressed by the quality of the CP/M systems. Each system provided the standard commands, and a bit more.

CP/M has a standard set of manuals (a must for CP/M users) which apply to all systems. These manuals will never win prizes for training material. They are clear if you have some experience, but the average beginner is liable to get lost. Beyond the regular manuals, the documentation provided by each supplier varied. Cybernetics had a few pages, Lifeboat had 28 pages and Pickles & Trout had a 142 page manual.

Cybernetics produces a "standard" CP/M package with some enhancements. It is fast and impressive, but adds only a few utilities:

FORMAT - For disk formatting.

SET - Set Input/Output characteristics.

RX - File receive utility.

TX - File transfer utility.

BACKUP - For disk backups.

This caused no difficulty in working with the system, but some of the utilities provided by the other systems were very helpful. Cybernetics big selling point is that they also have a large number of programs for CP/M which they can provide, including some special purpose programs such as their Automated Patient History for Doctors (which we haven't seen advertised elsewhere).

Lifeboat's CP/M added some new files and utilities to their distribution diskette. There were:

TOF - Top of form utility for printing.

FORMAT - Formats a diskette for single or double density.

COPY - Copies from one diskette to another. This is a full diskette copy function like a BACKUP.

FILECOPY - Transfers files from one disk to another on a single drive system.

SAVEUSER - Saves a modified system to disk without having to go through SYSGEN. Saves the complete BIOS portion of the system, allowing drivers to be modified with DDT.

CONFIG - A menu driven program that allows the user to display his system configuration and modify things like serial port parameters, disk seek rates, parallel printer formats, etc.

MEMR - A memory test routine for your system.

ONEDRIVE - Used to simulate a multiple drive system on a one drive system so the distribution disk can be backed up.

The Pickles & Trout (P&T) CP/M also includes a number of new utilities:

RESIZER - Creates a P&T CP/M system for a particular memory size.

FORMAT - Formats single and double density diskettes.

DDCHECK - Non-destructive double density disk check.

DDTEST - Tests double density diskettes.

SDTEST - Tests single density diskettes.

SETUP - Allows the user to set up Input/Output assignments and serial ports.

FASTCOPY - A fast copy routine for BACKUP.

IOFREEZE - Displays Input/Output parameters and allows them to be permanently set.

While we were testing CP/M, the FILETRAN utility for CP/M from Business Microproducts (1838 Catalina Court, Livermore, CA 94550) also arrived for review. This is another TRSDOS to CP/M transfer program which allows the user to move over to CP/M if it appears that he needs to. FILETRAN worked fast and reliably during testing.

FILETRAN has some special abilities which make it a perfect addition to many of the CP/M systems. The Model I version will transfer files both ways (CP/M to TRSDOS and TRSDOS to CP/M). Business Microproducts has indicated they might extend the Model II FILETRAN to include this capability.

In addition to the ability to transfer files, FILETRAN also allows you to display memory and disk sectors (CP/M or TRSDOS format). This can be very useful for checking on problems in transfer, or even for just looking at a storage format on the disk when you are not transferring files.

FILETRAN also has a utility built in that allows you to scan files for ASCII strings or for differences between the file and the proper syntax for Microsoft BASIC 5.0 as it has been released for CP/M.

TIME - Displays the system time.

SETTIME - Sets the system time.

SETMISC - Sets various system parameters.

TRS2CPM - Transfers a file from TRSDOS format diskettes to CP/M diskettes.

DENSITY - Tests the density of a diskette.

Evaluation

Tests of the three versions of CP/M preceeded tests of the software listed in Table 2. All of the programs performed without difficulty on each CP/M system.

Attempts to transfer programs from one system disk to another met with limited success. It turned out to be necessary to copy files to be transfered onto a disk with no CP/M system to prevent disk errors. Digital Research was contacted by telephone, but could not explain why that should happen. However, it was later found by talking to others that it comes from a disagreement as to how certain areas of the system disk should be handled. It turned out to be a minor problem, yet still somewhat disturbing.

The Pickles & Trout TRSDOS to CP/M conversion utility was especially pleasing. With this utility, one is able to transfer both programs and data to CP/M files and then execute the programs under Microsoft BASIC in CP/M. More important to many users, the possibility of converting files to CP/M without losing your data means that it is not necessary to stay with TRSDOS if the system you want is available only on CP/M.

One major problem which occured with CP/M turned out to be due to the way the Line Printer III works. It seems that this printer is set with a line feed that most of the CP/M's do not expect. This results in much double spaced copy, unless the printer port is set to handle it or an internal switch setting in the Line Printer III is changed. This is all well and good, except that when the printer port is set up to reject extra line feeds, it also rejects line spacing information from text editors when you try to double or triple space.

Overall, I like CP/M, and would recommend any of the three we tested. The big deciding factor for most people will not be what is in CP/M: They are all the same. Rather, it will be a combination of price and the additional utilities that will be the deciding factor. Table 3 gives the addresses and the price information as of February 1981.

How does it compare to TRSDOS?

For many of you the big holdup in using CP/M will probably be: 1) you already have a DOS, 2) it will cost extra money to get CP/M, or 3) you are not sure you can really handle the new DOS.

On the first two points, you will have to wrestle with yourself. The last is easily covered by comparing CP/M with TRSDOS.

Some of the really impressive features of CP/M are tied with its speed. When you first boot up a

TABLE 1
Comparison of TRSDOS and CP/M

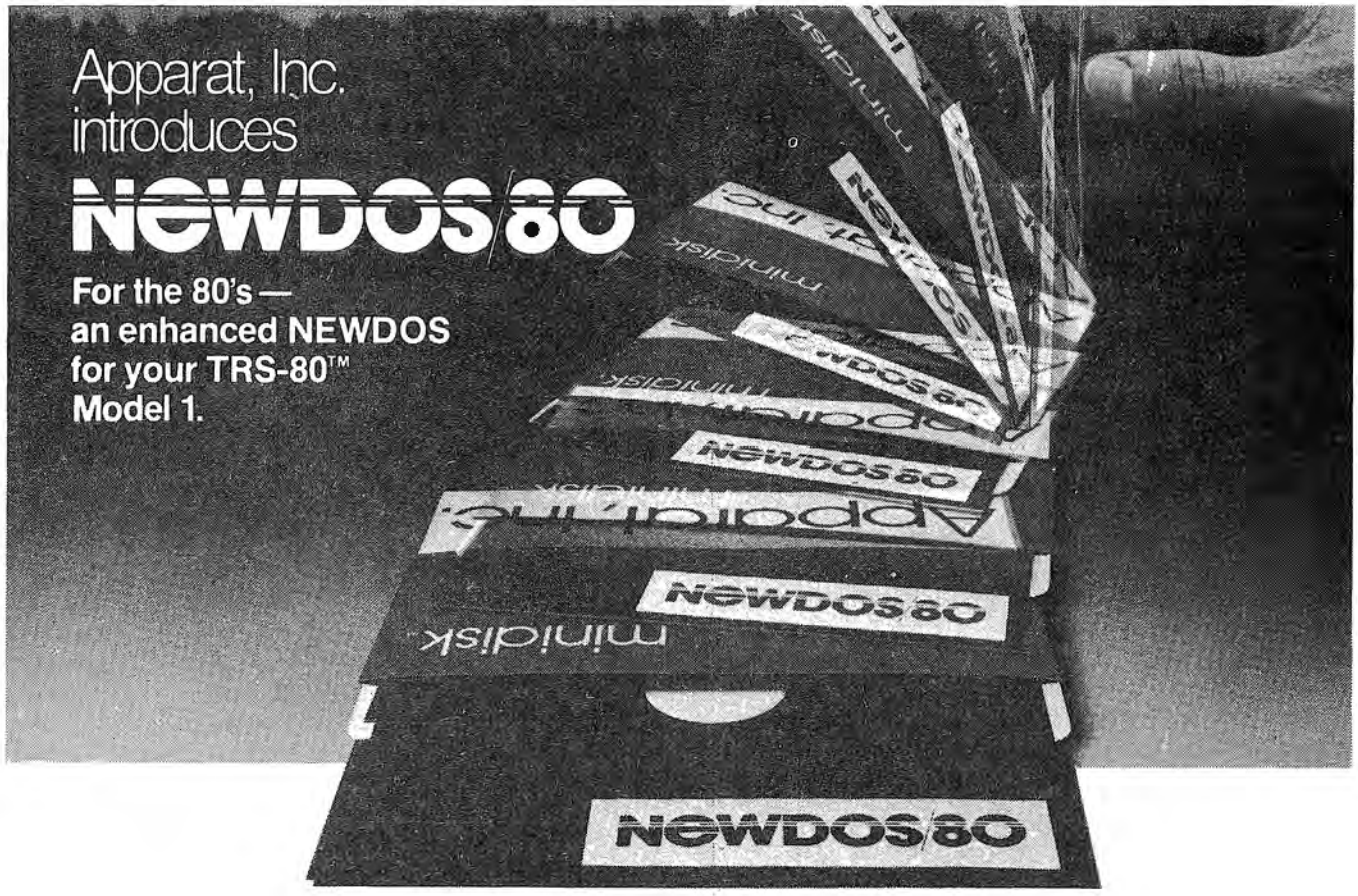
TRSDOS	Cybernetics	Lifeboat	P & T
AGAIN	***	***	***
ANALYZE	STAT	STAT	STAT
APPEND	PIP	PIP	PIP
ATTRIB	STAT	STAT	STAT
AUTO	***	***	***
BUILD	ED	ED	ED
CLEAR	***	***	***
CLOCK	***	***	***
CLS	***	***	***
COPY	PIP	PIP/COPY	PIP
CREATE	***	***	***
DATE	***	***	***
DEBUG	DDT	DDT	DDT
DIR	DIR	DIR	DIR
DO	SUBMIT	SUBMIT	SUBMIT
DUAL	(CTRL-P)	(CTRL-P)	(CTRL-P)
DUMP	SAVE	SAVE	SAVE
ECHO	(CTRL-P)	(CTRL-P)	(CTRL-P)
ERROR	***	***	***
FORMS	SET	CONFIG	SETMISC
FREE	STAT	STAT	STAT
HELP	***	***	***
HOST	***	***	***
I	(CTRL-C)	(CTRL-C)	(CTRL-C)
KILL	ERA	ERA	ERA
LIB	***	***	***
LIST	TYPE/DUMP	TYPE/DUMP	TYPE/DUMP
LOAD	***	***	***
MOVE	PIP	PIP	PIP
PAUSE	***	***	***
PRINT	TYPE/PIP	TYPE/PIP	TYPE/PIP
PROT	STAT	STAT	STAT
PURGE	ERA	ERA	ERA
RECEIVE	PIP	PIP	PIP
RENAME	REN	REN	REN
RESET	(CTRL-C)	(CTRL-C)	(CTRL-C)
SCREEN	***	***	***
SETCOM	SET	CONFIG	SETUP
SPOOL	***	***	***
STATUS	STAT	STAT	STAT
TIME	***	***	TIME
VERIFY	***	***	***

NOTE:

1. CP/M commands are often multi-purpose and very flexible so they often appear in more than one place.
2. This table was produced by simply listing the closest commands, transient or built-in, that could perform the task done by the TRSDOS command.
3. This table is based on the capabilities of the TRSDOS 2.0.

Apparat, Inc. introduces NEWDOS/80

For the 80's — an enhanced NEWDOS for your TRS-80™ Model 1.



Apparat, Inc announces the most powerful Disk Operaion System for the TRS-80. It has been designed for the sophisticated user and professional programmer who demands the ultimate in disk operation systems.

This new member to the Apparat NEWDOS' family is upward compatible with present NEWDOS 2.1 and is supplied on Diskette, complete with enhanced NEWDOS + utility programs and documentation. Some of the NEWDOS/80 features are:

- New BASIC commands that support files with variable record lengths up to 4095 Bytes long.
- Mix or match disk drives. Supports any track count from 18 to 80. Use 35, 40, or 77 track 5" mini disk drives or 8" disk drives any combination.
- Upward compatible with NEWDOS 2.1 and TRSDOS 2.3.
- A security boot-up for BASIC or machine code application programs. User never sees "DOS READY" or ">READY" and is unable to "BREAK", clear screen or issue any direct BASIC statement including "LIST".
- New editing commands that allow program lines to be deleted from one location and moved to another or to allow the duplication of a program line with the deletion of the original.
- Enhanced and improved RENUMBERED that allows relocation of subroutines.
- Powerful chaining commands.
- Print Spooler.
- DGF function; simultaneous striking of the D, F, and G keys will allow the user to enter a mini-DOS to perform some DOS commands without disturbing the resident program. (e.g. dir while in scripsit)
- Includes machine language Superzap/80 and all Apparat 2.1 utilties.
- Enter debug any time by pressing 123 keys. Also allow disk I/O.
- Diskette "Purge" commands.
- Specifiable system options (limited sysgen type commands).
- Increased directory capacity.
- Most software operating under NEWDOS will run on Model III.

NEWDOS/80 with all of the NEWDOS+ utility programs, many of which have been enhanced, is priced at just $149.00 and is available at most TRS-80 dealers.

NEWDOS/80 relies on the TRSDOS and Disk Basic Reference Manual published by Radio Shack.

NEWDOS/80 documentation supports its enhancements and upgrades only.

NEWDOS/80 FOR THE TRS-80 MODEL III AVAILABLE NOW!

FEATURES DOUBLE DENSITY 80 TRACK DRIVE SUPPORT

TO PURCHASE NEWDOS/80, COMPLETE AND MAIL TO:

Apparat, Inc.
4401 S. Tamarac Parkway
Denver, CO 80237

CALL TODAY
Order Desk 1-800-525-7674 (Except CO, AK & HI)
Customer Service 1-303-741-1778

☐ Check ☐ Money Order ☐ Master Charge ☐ Visa

Card No _____ Expiration Date _____

Colo residents add 6.5% sales tax
Outside Continental U.S. residents add $10 for Postage & Handling

☐ Model I ☐ Model III

Please rush _____ NEWDOS/80 @ $149 each to:

Name _____

Address _____

City _____ State _____ Zip _____

Phone _____

Circle 28

CP/M system on the Model II there is no messing around with memory or special checks and protection setups. The system comes right up to CP/M in a very few seconds. It happens so fast you are not sure at first that it is done.

Once you are in CP/M, it is even more delightful since the first thing that does *not* happen is the error because you forgot to push the CAPs key!. CP/M is prepared to accept either upper or lower case letters for its commands.

Now comes CP/M's speed advantage in everything else. Format a disk with one of the utilities, and it steps through the tracks so fast you are not expecting it to be done. Want a system disk? You can dump a tailored system to a disk in seconds. Then you use PIP, the general file transfer program to add whatever else you want to the disk.

Do you need to do a large number of file transfers? CP/M's PIP utility outdoes even TRSDOS 2.0's ability to use wildcards for transfer. A wildcard allows you to specify a group of files with some common feature. For example, PIP A:=B:*.ASM, will copy all files on drive B with the extension ASM onto drive A. (Unlike TRSDOS and others, CP/M names the drives A,B,C, etc). We can also get single letter matches such as, PIP A:=B:FILE?.ASM, to transfer all files with the first four letters of the name "FILE" and a fifth letter of any character and an extension of ASM to disk A from disk B.

TRSDOS 2.0 has a limited wildcard facility. However, the trend from the early TRSDOS's seems to indicate they are including more CP/M-like abilities.

CP/M does have some differences that can be frustrating if you are used to working with TRSDOS. First, CP/M will not automatically search for a file if it is not found on the system disk. You have to specify which disk the file is on or it will not find it.

The only case where you don't tell the system what disk to look on is when you want to find a file on the currently logged-in disk. All commands are about this disk *unless* you tell the system to go to another disk for the file. This disk doesn't even have to be a system disk and it does not have to be in drive 0. If you are used to working with TRSDOS, this could be somewhat confusing.

If you are converting from TRSDOS, the terminology is confusing and the manuals are not clear. The best way around this this the new *CP/M Handbook*, by Rodney Zaks (see inset). Without some such help, you may find it hard to pick up the threads of CP/M.

CP/M tends to be more of a programmer's system. All of the capabilities are there and pretty well documented if you like fooling around with operating systems. The beginner or non-programmer can be lost very quickly. However, if you have a good programmer putting a package together for you, he can do a lot with CP/M very easily and tailor a very special system to your needs so that you never see anything but your particular application.

Should you get CP/M?

It depends - where the software you want to use isn't available for TRSDOS, do not be afraid to get it on CP/M. It is every bit as good an operating system as TRSDOS. However, don't believe the "true believers", who preach that either 1) TRSDOS is so full of holes that it is worthless, or 2) CP/M is an OLD system that is outdated. Neither is the case.

TRSDOS is improving steadily. Holes that we saw earlier are getting fixed, slowly but perceptibly. On the other hand, CP/M has been around for a while

The CP/M Handbook
by Rodnay Zaks
SYBEX

Rodnay Zaks is widely known for his how-to books in the microprocessor field. CP/M Handbook is another book in his successful series.

The CP/M operating system is a popular, versatile operating system. However, documentation for the system has always tended to be difficult for the beginning programmer to read. Digital Research provides a good set of manuals, but they leave many questions unanswered. Pickles & Trout provides the best supplementary documentation which goes far beyond Digital Research manuals, but even this leaves unanswered questions.

Zaks' CP/M Handbook is the most complete and readable introduction to the CP/M language available. It covers CP/M (Control Program for Microprocessors) through version 2.2 and MP/M (Multiprogramming Control Program for Microprocessors) through version 1.0. The eight chapters are:

1. Introduction to CP/M and MP/M
2. CP/M and MP/M Facilities
3. Handling files with PIP
4. Using the Editor
5. Inside CP/M (and MP/M)
6. Reference guide to CP/M and MP/M Commands and Programs
7. Practical Hints
8. The future

Also included are 15 appendices covering a variety of material such as keywords, devices, error messages and others.

Upon getting the book from a local computer store, I rapidly read the book from cover to cover. Then I read it again and again. It is now one of the most browbeaten books in my library, heavily annotated, highlighted, and gets a lot of use when programming with the CP/M operating system.

It contains many useful tidbits. I admit that each thing I thought was new was also in the Digital Research manuals. However, Zaks' book is *much* easier to read.

While I cannot vouch for the accuracy of the MP/M material (I don't work with MP/M), I can say that I found nothing in error with the CP/M material other than a couple of typos in the text.

It is important to make clear that this book deals with several versions of CP/M. So everything it says may not apply to any particular system. In addition, for the TRS80, most of the available CP/M systems are "enhanced" or modified in some way by the people who adapted them to the TRS-80. So even here, you may find some differences from the text.

Everything considered though, the book is invaluable and is recommended to anyone who seriously is interested in working with CP/M. Even old hands with CP/M will probably find some interesting ideas in it.

The CP/M Handbook is available from SYBEX, 2344 Sixth Street, Berkeley, CA 94710 for $13.95

MODEL II

26-4002
64K 1 Drive
$3440.00

26-4160 1 Drive Exp.....$1034.00
26-4161 2 Drive Exp.....1574.00
26-4162 3 Drive Exp.....2114.00
26-4530 Scripsit II.....265.00
26-4512 Profile II.....162.00
26-4511 Visicalc II.....265.00
26-4501 Gen Ledger.....180.00
26-4506 Mail List.....72.00

$ DISCOUNT $

TRS-80®

DEALER A301

COMPUTER SPECIALISTS

CALL US. . .
SAVE MONEY

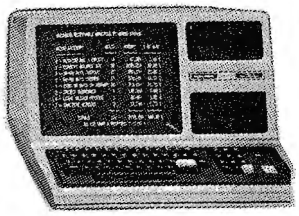
We carry the full line of TRS-80 Computers. All other R.S. software, furniture, and accessories at discount from catalog price. We stock most items to assure you fast delivery and save you money.

26-1140 Expansion Interface.....$249.00
26-1141 16K Exp. Interface.....359.00
26-1142 32K Exp. INterface.....469.00
26-1145 RS232C Board.....84.00
26-1160/1 Mini Disk Drive.....419.00
26-1563 Scripsit-Disk.....79.00
26-1566 Visicalc.....83.00

* * * * * **PRINTERS** * * * * *

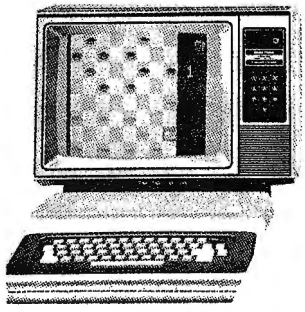
26-1155 Quick Printer.....187.00
26-1167 9½ Dot Matrix Printer.....360.00
26-1166 Line Printer VI.....1080.00
26-1158 Daisy Wheel II.....1799.00
26-1165 Line Printer V.....1710.00

MODEL III



26-1061 4K I.....$629.00
26-1062 16K III.....865.00
26-1063 32K III
W/2 Drives, RS232.....2225.00

COLOR



26-3001 4K.....$353.00
26-3002 16K Ext. Basic.....533.00
26-3008 Joysticks.....22.50
26-3010 Color Video.....353.00
26-1206 Recorder.....54.00

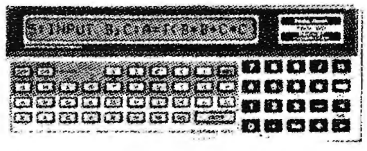
PRINTERS

CENTRONICS

BEST PRICES

Fast 100 CPS Centronics
730-1A Printer.....$577.00
Text Quality Centronics
737-1 Printer.....$737.00

Pocket Computer



26-3501 1.9K P.C.....$221.00
26-3503 Cassette IF.....45.00
14-812 Recorder.....72.00

commodore

AUTHORIZED DEALER
CALL FOR PRICES

ALL POCKET AND COLOR COMPUTER SOFTWARE SOLD AT DISCOUNT

WRITE US FOR A
FREE CATALOG

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. & Export

TRS-80 is a registered trademark of the Tandy Corp.

R.S. 90 Day Limited Warranty
F-48 Form Provided

Largest Inventory In The S.E. U.S.A.

and this is one of its major strengths! CP/M just doesn't go around making mistakes on where it put your information because errors have been found and corrected years ago.

CP/M is not static either. Digital Research is constantly improving and correcting CP/M and its utilities. They are developing new and better software for use on CP/M. In fact, they just announced a new PL/I language compiler to run with CP/M.

CP/M has another major strength. That is the number of people who use it and the number of systems on which it runs. It has been around long enough so that practically any application is available somewhere on it. There are also more and better languages available for CP/M than for TRSDOS.

Do you want to program in the C programming language, or FORTH on the Model II, or any one of a dozen special purpose languages? Do you want everything to read the same files on your disk? If so, then CP/M could be for you. For programmers, there is an additional advantage. Working in CP/M instead of TRSDOS opens your potential market. Since many computers run CP/M, and the Model II runs a "standard" CP/M, you can take your expertise to other computers as well.

As with so many other things, the advantages and disadvantages of CP/M are fluid: they don't give you a clear cut answer to the question, "Should I choose CP/M?" In any case, be sure to consider it before you decide to develop a new "wheel" that may already be running under CP/M.

Which CP/M should you choose?

In the preparation of this article, I questioned a number of suppliers of CP/M software and asked which one they would recommend for running their software on the TRS-80. About 50% indicated they were not personally familiar with TRS-80 Model II, and could not make an intelligent recommendation. Of the remaining half, about 70% said they would recommend Pickles & Trout. One company recommended Lifeboat Associates, the others had no opinion.

This was not a careful survey of opinions, just a shot in the dark approach as I asked about software compatibility for the Model II. Don't take these as hard recommendations for one system over another. There are hundreds of CP/M software dealers out there and I hit only a few.

Summary

In the final analysis, we can safely say that with the introduction of the TRS-80 Model II, CP/M really becomes a viable alternative for software development.

Despite the argument that goes on about which is better, the only real deciding factor between TRSDOS and CP/M is which has the programs you need to run. If you want PASCAL, CP/M has it, TRSDOS does not (as of this writing). If you want Automated Patient History, there is one through Cybernetics CP/M, I don't know of one for TRSDOS.

As to which of the three CP/M's is better, well there isn't one that is better as a CP/M, the Pickles & Trout version is enhanced the most, both inside the operating system and in the utilities. It seems to be preferred among the CP/M software suppliers who know the Model II. However, I would not hesitate to recommend any of the systems we tested.

It's up to you. Consider it seriously, Don't ignore its existence. But in the end, choose the system you *need*. It very well might turn out to be TRS-80 Model II CP/M. ●

Table 2
CP/M Software Tested with TRS-80 Model II CP/M Systems

Stackworks FORTH
VEDIT Text Editor
Nevada COBOL
Microsoft COBOL
PASCAL/MT+
Microsoft FORTRAN
Microsoft BASIC
CBASIC
Microsoft BASIC Compiler
FILETRAN
Code Works Small C
Microdata Membership Billing
Microsoft MAC80 Assembler
And various programs written in the languages above by the author.

Note:
A number of these systems are presently under review for publication in the Journal later this year. We are not covering their capabilities in this article for that reason.

Table 3
CP/M Ordering Information

Cybernetics
8041 Newman Ave., Suite 208
Huntington Beach, CA 92647
Price: $250.00

Lifeboat Associates
1651 Third Avenue
New York, NY 10028
Price: $170.00

Pickles & Trout
P.O. Box 1206
Goleta, CA 93116
Price: $185.00

Dragonquest!

In a desperate race against the sun you search for SMAEGOR Monarch of Dragonfolk, who has kidnapped the Princess of the Realm and holds her in a distant and unknown place. In a quest for Honor and glory, you must search the land, seeking out the tools needed for the ultimate confrontation. On The River Delta, in the abandoned Temple of Baathteski, Goddess of the Blade, everywhere, clues abound. But WHERE is the Princess?

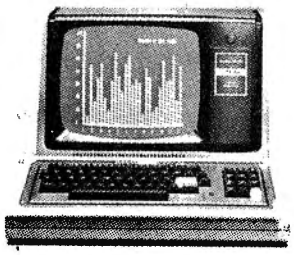
Now, as never before, the genius of CHARLES FORSYTHE shines in this new machine language ADVENTURE. DRAGONQUEST! Can YOU save M'lady from the iron clutches of SMAEGOR?

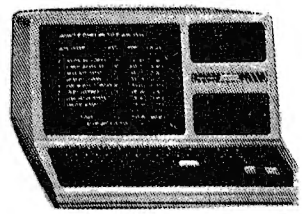
TAPE $15.95 *Dealer Inquiries Invited* **DISK $21.95**

THE PROGRAMMER'S GUILD
BOX 66, PETERBOROUGH, N.H. 03458
(603) 924-6065 AFTER 6 PM EST
MASTERCHARGE AND VISA ACCEPTED

Circle 30

Files and Foibles





TAKING THE RANDOM OUT OF RANDOM FILES

Model I, II, and III Disk Systems

T R Dettmann

Last issue, we covered the basics of Random Access file structure. The Basic access techniques are actually very simple. After you have opened a file for Random Access, you must create a FIELD statement to describe how the data will be put into the file. Then you LSET or RSET data into the FIELD and PUT the information into the file on the disk.

To get information out of the file, you simply GET the information from the file and then set your program variables equal to the appropriate fields (with conversion if they are numbers).

Sounds simple, yet there are many details to worry about and a lot of problems that you can get into if you do not take a little care. Look at the last issue's article for a more detailed description of the individual commands , or look over your DOS Basic manual. This issue, we are going to extend our ability by using Random Access files.

Where is it?

Random Access files present us with much power, but require us to do a lot more work than with sequential files. With a Random file, we can get at any piece of information in the file without delay - without searching. All we need to know is the record number for the information, that is, what was the number we used to put the information into the file?

The unfortunate problem for many applications is that the information we want to file has no natural ordering number. For example, let's say we are storing a list of our record collection in a random access file. We could assign a number to each record and store the information about the record by the record number. That's ok, but it has some problems: (1) The numbers have no *natural* relation to records. The numbering is *imposed*; (2) the information we want about the records has no relation to the numbers unless we also impose a numbering system; (3) we will still have to add some kind of system to handle things like alphabetical searches or category searches.

We might go on, but the primary problem should be clear, for many situations the record numbers don't bear a direct relation to the problem unless we impose a relation. This isn't necessarily bad, but is usually over-used.

Problems do exist where such an externally imposed relation is useful. For example, in an inventory with part numbers from 1 to 1000, we might simply assign space on a data disk so that each part number corresponds to the record number exactly. This is a very simple minded solution to a very tricky problem. The advantages are: (1) It is very simple to program; (2) it doesn't involve much special programming to make it work; (3) there is no translation to make between part and record number; and, (4) it is very fast.

The disadvantages are: (1) Part numbers not being used are just empty space on the disk; (2) part numbers *must* be forced into correspondence with the record numbers; (3) the largest part number is set by the maximum size of the disk file; and, (4) the information on a single part might not correspond to the size of one record.

If the problem is best solved without special tricks or programming techniques, then by all means try the simplest way. If some of the disadvantages are important it may be necessary to improve your programming skill to accomodate the requirements of your application.

Physical and Logical Records

In some Disk Operating Systems (DOS), it is possible to build Random Access files with a record length (number of bytes of disk space) of exactly the size needed for your problem. Model II and III TRSDOS can do this as can VTOS 4.0, NEWDOS and others for the Model I.

In the systems where you can specify the record length for a file, you can easily make your record length correspond to the amount of information you want to store for an item. The standard form for opening a file with a specified record length is: OPEN"X",N,"filename",L where

X is "R" or "D" depending on the system (check your manual); N is the file number and L is the record length.

If you have this kind of a DOS, you don't really need what follows, but read on anyway.

If you have a more limited system (earlier TRSDOS for the Model I for example), you can't define random access record lengths. Every record on the disk is 255 bytes (or 256, depending on the DOS).

The record that is actually stored on disk is called the *physical record* and it's length is the *physical record length (PRL)*. If we are limited to 255 bytes, or any other length for that matter, but the items we want to store are shorter, then we say the item's *logical record length (LRL)* is shorter.

In fact, if the LRL is short enough, we can fit more than one into a PRL, and fool the system by handling the *logical* records as items and packing them into *physical* records.

Let's see how we can do this with an example. Assume we have an inventory with the parameters given in Figure 1. Each logical record is 38 bytes long. If we leave space for future expansion and assign each logical record 51 bytes of space, then we can pack five logical records into one physical record.

Figure 1
Simplified Inventory Model

Variable	Program Name	Field Name	Size
Part Number	PN%	PN$(FD)	2
Description	DS$	DS$(FD)	20
Location	L%	L$(FD)	2
Cost	KS!	KS$(FD)	4
Price (selling)	PR!	PR$(FD)	4
No. on hand	OH%	OH$(FD)	2
Reorder number	RN%	RN$(FD)	
Supplier	SP%	SP$(FD)	2
	Total	Size =	38

If we start working with the first record of the file, then physical record 1 will hold logical records 1 through 5, and so on:

Physcial Record	Logical Records
1	1-5
2	6-10
3	11-15
4	16-20
5	21-25

Now the item number will be the same as the logical record number. But how can we locate a record in a simple way? We use an operation called the *Modulus*.

The modulus of a number is the remainder when divided by another number. To see how this works, take 13 divided by 5, which gives us 2 with a remainder of 3. So, 3 is the modulus of 13 with respect to 5. Mathematically this is expressed as:

13 MOD 5 = 3

Now consider the following table of the MOD of a whole series of numbers (ignore the third column for now):

N	N MOD 5	(N-1)MOD 5+1
1	1	1
2	2	2
3	3	3
4	4	4
5	0	5
6	1	1
7	2	2
8	3	3
9	4	4
10	0	5
11	1	1
12	2	2
13	3	3
14	4	4
15	0	5

As the table shows, the modulus of a number repeats itself over and over again at exactly the rate of the divisor 5. We want the numbers to go from 1 through 5 and repeat in that sequence. Consider what will happen with (N-1) MOD 5+1. Look at the third column in the table.

The TRS-80 Model II has this operation built-in. If your system does not, there is still an easy way to compute the modulus:

X MOD Y = X - INT(X/Y)*Y

Try using this formula to compute column two in the table. You can make it work very easily on your computer by defining a "Modulus" function:

```
DEF FNMD(X,Y) = X-INT(X/Y)*Y
```

Now, to get the number of the record within a physical record where a particular item number (N) is stored, we can use the statement:

```
FD = FNMD((N-1),5) + 1
```

FD is the field within the particular physical record where the item is stored. To find the physical record try computing: PRN =INT((N-1)/5)+1. You will find that this PRN corresponds exactly to the physical record where a logical record is located. For example, if N=5, then PRN = INT((5-1)/5)+1 = INT(4/5)+1 =1. And if N = 1, then PRN = INT((1-1)/5)+1 =1.

When we know the record number N, we then GET or PUT PRN = INT((N-1)/5)+1 to find the appropriate record. The simplest way to locate the logical record is to have each of our variable fields be an array from 1 through 5. Then we field the file like this:

```
FIELD #1,2ASPN$(1), 20ASDS$(1),
2ASL$(1), 4ASKS$(1), 4ASPR$(1),
2ASOH$(1), 2ASRN$(1), 2ASSP$(1),
13ASDM$(1), 2ASPN$(2),.....
```

The FIELD statement will continue until all the fields are defined. Note the use of the dummy string array DM$ to fill out each logical record.

This definition of the file's FIELD is rather cumbersome and well worth shortening. We can use the same arrays and simply write a loop to define the fields like this:

```
FORI=1TO5:FIELD#1,(I-1)*51ASDY$,
2ASPN$(I), 20ASDS$(I), 2ASL$(I),
4ASKS$(I), 4ASPR$(I), 2ASOH$(I),
2ASRN$(I), 2ASSP$(I):NEXTI
```

This will define the fields exactly the same as the first FIELD statement. To put the variables into the fields, we write:

```
LSET PN$(FD) = MKI$(PN%):LSET
DS$(FD) = DS$
:LSET L$(FD) = MKI$(L%):LSETKS$(FD)
= MKS$(KS!)
:LSET PR$(FD) = MKS$(PR!):LSET
OH$(FD) = MKI$(OH%): LSET RN$(FD) =
MKI$(RN%):LSET SP$(FD) = MKI$(SP%)
```

The variable types have been included so you can see the relation between them and the fields.

To recover the information from the file, we GET the record and then use the following:

```
PN% = CVI(PN$(FD)):DS$ = DS$(FD):L%
= CVI(L$(FD))
KS! = CVS(KS$(FD)):PR! = CVS(PR$(
FD))
OH% = CVI(OH$(FD)):RN% = CVS(RN$(
FD))
SP% = CVI(SP$(FD))
```

Now we have a way of packing the records to get more than one logical record into a physical record. It is the same, whether you are using a Model I or a more advanced Disk Basic.

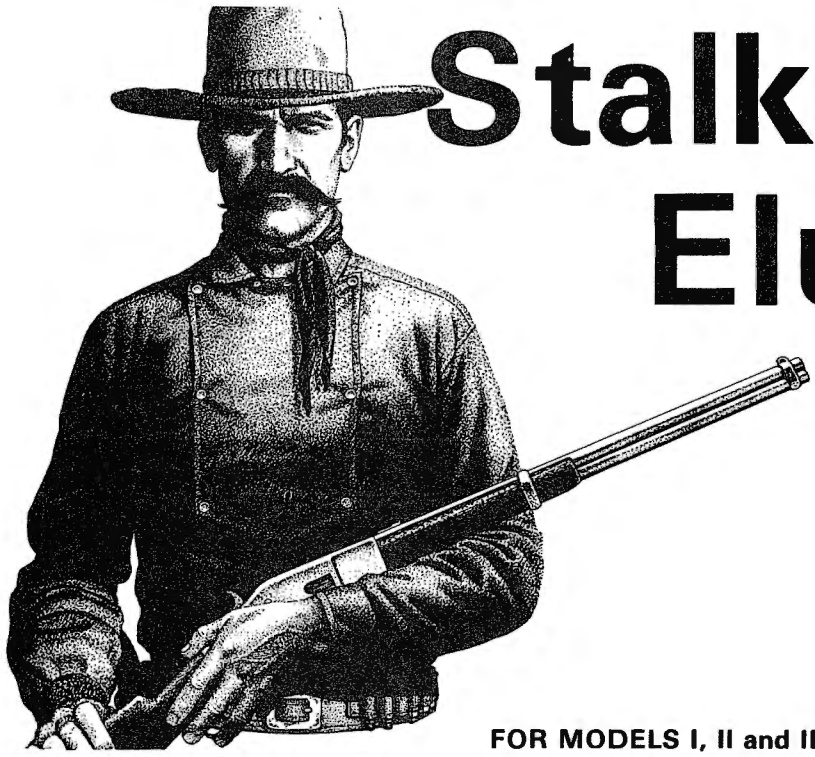
No matter what system you use, sometimes the logical record number will not relate to a number. An example: We want to store a random access file of time management information. We want to have tasks, date due, priority, date started, and for whom in the file.

Indexed Files

One method for getting at any piece of information in a file is to create an index. The index will contain the *key* which will be used to access the data contained in the file. Consider our time management file. A natural key for this file is the date the task is due. We simply store the date for each entry in an array (stored on disk between program runs) which is then sorted into order by date.

In order to use our key index, search the array until the correct key is found, and then print out the record corresponding to it. If there is more than one entry for the same date, print out all records for that date.

In our next installment, we will cover keyed files in more detail. It will also cover the maintenance of such an index. ●



Stalking the Elusive Print Using

FOR MODELS I, II and III

R C Bahn

Introduction

In the language of Level II, the statement: PRINT USING A$, is the most versatile member of the video screen (PRINT) and printer output (LPRINT) statements. A string variable, A$, defines what we shall call the line image. By this mechanism, one can specify the exact placement of numeric or string variables within a line. In addition, if the variable is numeric, the number of decimal places to be displayed to the left and to the right of the decimal point can be designated.

Numeric variables can be adorned by a variety of auxiliary symbols such as preceding asterisks, plus or minus signs, and the dollar sign. Commas can be placed between every third digit. Finally, numeric variables may be printed in exponential notation.

Alphabetical data may also be displayed as desired. Within the line image, one may designate the placement of a single string variable or series of them. The first character of a string variable, such as the initials of a name may be specified. String variables such as labels or special symbols may be embedded within the line image.

In the PRINT USING A$ statement, a semicolon follows A$. This is followed by a list of the names of the variables represented in the line image. An outline of the specific symbolic representation of each of the options in the line image appears in the summary.

When does one use the PRINT USING A$ statement? The easy answer is, anytime. The print using statement is so powerful, it can essentially simulate any of the other output statements, and more. However, from a practical standpoint, since the PRINT USING A$ statement necessitates preplanning, one might prefer to use this statement only when the other choices such as PRINT (with or without commas or semicolons), PRINT@, or PRINT TAB are inadequate.

```
100 CLEAR 1000
110 A$="% %  **$######.###+      +####,####.######      ##.######[[[["
115 REM 012345678901234567890123456789012345678901234567890123456789
120 CLS
130 B$=STRING$(63,"-")
140 C$="         %                                          %"
145 REM 012345678901234567890123456789012345678901234567890123456789012
150 D$=" DEMONSTRATION OF PRINT USING A$;"
160 E$(4)="SUM"
170 PRINT C$
180 PRINT USING C$;D$
190 PRINT:PRINT
200 INPUT" ENTER 3 NUMBERS(HIGH,MEDIAN,LOW)";A#(1),A#(2),A#(3)
210 SUM#=0
220 PRINT:PRINT A$
230 FOR I=1 TO 3
240 A(I)=A#(I)
250 SUM#=SUM#+A#(I)
260 E$(I)=STR$(I)
270 PRINT USING A$;E$(I),A(I),A#(I),A#(I)
280 NEXT I
290 PRINT B$
300 SUM=SUM#
310 PRINT USING A$;E$(4),SUM,SUM#,SUM#
320 PRINT@900," "
330 INPUT"TO PROCEED,PRESS ENTER";Z$
340 GOTO 120
```

Model II users - the percent sign in this program must be changed to a backward slash (use Control-9).
NOTE: Please interpret the [as an "up-arrow", or caret.

Figure 1

Spacing on lines 110 and 140 is critical. REMarks lines 115 and 145 have been added to provide a count guide.

UNIQUE !

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

REAL VALUE

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

MYSTERY REMOVED

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

- **FLIPPY** Allows the use of both sides of a diskette with a single-headed drive by simply turning the diskette over (model 40-1&80-1).
- **TRACK DENSITY** Specified in tracks per inch (TPI). Refers to the number of tracks per radial inch on the diskette. Typically 48 TPI=40 usable tracks and 96 TPI=80 usable tracks.
- **DOUBLE DENSITY** Refers to recording density in bits per inch (bpi). Typically single density means data can be recorded up to 2,938 bpi; double density means data can be recourded up to 5,876 bpi.
- **DOUBLE SIDED** Refers to number of read/write heads. Single-sided is one head, read/write one side only; double-sided is dual heads allowing read/write operations on both sides of the diskette. A double sided drive appears as two seperate drives to the controller.
- **ACCESS TIME** The time required for the head to move from one track to the next. Typecilly 5 to 40 milliseconds (ms).

SELECT YOURS!

- **40-Track "FLIPPY" $349.95**
(Model 40-1) 48 TPI, single dens. 125K bytes/side, double dens. 250K/side.
- **80-TRACK "FLIPPY" $459.95**
(Mod. 80-1) 96 TPI, single dens. 250K bytes/side, double dens. 500K/side.
- **40 TRACK DUAL HEAD $459.95**
(Mod. 80-2) 48 TPI, double-sided, 40 tracks/side; single dens. 250K bytes, double dens. 500K.
- **80 TRACK DUAL HEAD $599.95**
(Mod. 160-2) 96 TPI, double-sided, 80 tracks/side; single dens. 500K bytes, double dens. 1 megabytes.

NOTE: All capacity values are unformated All models capable of single or double density. All with power supply/silver enclosure 115 VAC, 50-60 Hz 115/230 VAC available.

SPECIAL PACKAGES

STARTER 40-Track FLIPPY drive,2-Drive cable,TRSDOS 2.3 Disk & Manual,Freight & Ins. $369.95 (reg. $401.00)

#1 40-Track FLIPPY drive
2-Drive cable
Newdos/80
Freight & Ins.$459.00(reg.$528 00)

#2 80-Track FLIPPY drive
2-Drive cable
Newdos/80
Freight & Ins. $569.00 (reg.$638 00)

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

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

- **DISK OPERATING SYSTEMS**
Newdos/80 $149.00
VTOS 4.0 $125.00
DOSPLUS 3.1 $99.95

- **DISKETTES, SOFT SECTOR,5 1/4"(box of 10)**

Single-sided, single density $29.95
Double-sided, double density $39.95

- **CABLES**
2-drive $24.95
4-drive $34.95
Extender $16.95

FREE TRIAL OFFER

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

WARRANTY

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

100% TESTED

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

ORDER NOW!!

To order by mail, specify Model Number(s) of Drive, cable, etc (above), enclose check, money order, VISA or MASTERCHARGE card number and expiration date, or request C O D shipment. Texas residents add 5% sales tax. Add $5 00 per drive for shipping and handling. Please allow 2 weeks for personel checks to clear our bank. No personel checks will be accepted on C.O.D. shipments-cash, money orders or certified checks only You will receive a card showing the exact C O D amount before your shipment arrives Be sure to include your name and shipping address WE SHIP PROMPTLY! In the event there is a slight delay, you will be notified of the shipping date and we will NOT charge your bankcard until the day we ship!

COMPARE AND BUY AEROCOMP!

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

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

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

*Trademark of Tandy Radio Shack.

WRITE AEROCOMP TODAY FOR MORE VALUES !!!

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

Dealers inqiries invited

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

Circle 13

One of the major differences between the PRINT USING A$ and the other output statements is that numeric variables of the former are right justified while the numeric variables of the latter are left justified. This means that when properly formed, the PRINT USING A$ statement aligns the decimal point on specific columns of the output.

One common problem associated with the PRINT USING A$ statement is that representation of the data as single precision variables may be inadequate to conform to the required range of output representation. This difficulty can be remedied by defining all the crucial numeric variables leading to the computer output as double precision variables. It should be remembered that in general in Level II, single precision variables are stored with seven digit accuracy and printed with six digit accuracy. Double precision variables are stored with 17 digit accuracy and printed with 16 digit accuracy. The arbitrary introduction of a formatted decimal point however, can influence the printed output. Thus, each application should be tested in the specific numerical range of use to assure oneself that the desired representation of the output has been achieved.

There are four common situations in which the PRINT USING A$ is specifically indicated. The first situation is when the decimal point should be aligned in columns. The second situation is when numeric variables need specific adornments such as the dollar sign, asterisks, signs, commas, or labels. The third situation is when the output must be divided into a number of columns which are not easily formed by the ordinary PRINT or LPRINT statements. The fourth situation is when a string field must be printed repetitively throughout the program in a specific portion of a line of the output.

Demonstration Program

It is easier to demonstrate the PRINT USING A$ statement than to describe it. The program in figure #1 allows you to designate various line images (A$) and to test its effect upon three samples of the data (A#(I)) you expect to process.

Discussion

The program demonstrates how your computer can teach you about programming. It is somewhat inefficient to write single isolated PRINT USING A$ statements to be tested by single numbers. This is particularly true when the intended use of this formatting technique is not only to print symbols but to relate a screen full of symbols to each other in a visually meaningful manner.

The program program provides the user with a full screen display for the effort of entering only three numbers. Subsequently, three different numeric formats may be tested and compared simultaneously. The actual format is printed on the line above the columns of data so that the relationship between format and data is clearly demonstrated. The program manipulates both string and numeric

Program Comments

LINE COMMENT

100 Clears memory and saves string space.
105 Remark line, added to allow for proper horizontal spacing.
110 A$ is the line image. The formats of four variables are defined. The first position identified by % % is the line label. The remaining variables are numerical. The last variable will be printed in the exponential format. Various adornments are associated with the first two numeric variables. This statement should be modified to explore the full potential of PRINT USING.
120 Clear screen, this is a return address from line 340.
130 B$ is the line drawn to separate the data from the sum.
140 C$ is the line image of the screen title.
145 Remark line, added (see line 105).
150 D$ is the screen title.
160 E$(I) are line labels. The other three labels are generated in 260.
170 The line image of the title is printed on the screen.
180 The screen title is printed. Note the alignment of the title with its format.
190 Skip two lines.
200 Input three test numbers. Explore large and small values in both the positive and negative range. A#(I) is a double precision numeric variable.
210 In addition to the input data, the sum of the input values is computed in double precision. To initialize summation, SUM# is set to zero.
220 Skip a line and print the line image (A$) for the data. This serves as a column header and allows you to see the relationship between the data format and the data. If the data is well formed, the decimal points will be aligned. If there is an overflow of the format, a percent sign (%) will precede the data and the position of the decimal point will be shifted on the video screen.
230 Loop to process 3 data points.
240 Conversion of a subscripted double precision variable (A#(I)) to a subscripted single precision variable (A(I)).
250 Sequential accumulation of the sum of the data in double precision.
260 Formation of the line label (E$(I)) from the loop index (I). The loop index is a numerical value. E$(I) is a string variable. The STR$ statement converts a numeric to a string variable.
270 Print the four variables designated in the line image A$. (E$(I) is the line label. A(I) and A#(I) are the single and double precision data variables respectively.
280 End of loop.
290 Print the line defined by B$.
300 Double to single precision conversion of SUM#.
310 Print the four variables designated in the line image A$. Note that the format properly aligns the label and the numbers. An important feature of the PRINT USING A$ statement is the capability to use the same line image repetitively.
320 Move the cursor to the bottom of the screen.
330 Hold screen image and provide user with prompt to continue.
340 Repeat program; to exit the program, press BREAK.

data and extends the use of formats to the results of a computation, the sum.

The program demonstrates the use of other means of video screen display such as the PRINT, PRINT@ and INPUT statements. Other programming techniques which are illustrated are: subscripted variables, a loop, the technique for the accumulation of a sum and the use of STRING$ and STR$ statements.

Summary

We can now summarize the various options for the PRINT USING A$ statement. Note that in the following examples the string variable A$, used as the line image, is introduced directly within the statement. Despite these examples, it is best in any major application, in which formats of entire lines are specified, to define the line images separately.

In general, the PRINT USING A$ statement formats strings and numbers. The following variations are allowed in the line image, A$. Combinations of most of these designations are possible.

1. # formats numbers: PRINT USING "####";9876.4

2. . designates position of the decimal point: PRINT USING "##.##";98.005 Note that if the format contains fewer significant figures than the stored variables, the printed number will be rounded.

3. , displays a comma to the left of every third digit: PRINT USING "####,";9876

4. ** fills leading spaces with asterisks. PRINT USING"**#####"; 98.46

5. $ fixed dollar sign: PRINT USING"$####.##"; 198.76

6. $$ floating dollar sign: PRINT USING"$$####.##"; 198.76

7. **$ floating dollar sign with leading spaces filled with asterisks: PRINT USING "**$####.##"; 8.76

8. ↑↑↑↑ exponential format PRINT USING"#.#####↑↑↑↑ ", 987654321

9. + the sign is printed either before or after the number.
PRINT USING"+####";987
PRINT USING"+####"-987
PRINT USING"####+;987
PRINT USING"####+";-987

10. - the minus sign is printed after the number. If the number is positive a space is printed:
PRINT USING"####-";987
PRINT USING"####-";-987

11. ! prints the first character of a string: PRINT USING"!";BOB

12. % % prints a string, the length if the number of spaces plus two: PRINT USING"% %";BOB

The PRINT USING A$ statement is not difficult to use. It gives one complete control over the formatting of a line. Care should be taken to assure that the printed number is exactly the desired representation. Differences in printed numbers are associated with single and double precision calculations, conversions of numbers from double to single precision and in rounding. You can only be confident of the results by thoughful planning and meaningful testing. ●

ANIMATION and the TRS-80

Works Without Modification on Model's I & III

Gary Sanderson
Quincy, Illinois

In the beginning Radio Shack created the TRS-80. In it they included hardware graphics capabilities and three graphics commands. Then came a "Video Display Worksheet", a "Graphics Character Chart", an indication that strings might be used with graphics, and in *Level II BASIC Reference Manual* (second edition, appendix C). As their latest contribution, a book *TRS-80 Graphics,* by Don Inman.

Utilization of the Basic commands provides a simple method to draw graphs and motionless figures. Animation requires not only graphics capabilities but also the capability of writing on the video in such a manner to provide the appearance of realistic motion. As you will soon see, the commands SET and RESET do not have the capability to write on the video fast enough to create realistic motion.

Through the use of several benchmark programs, included here to allow you to experience the results firsthand, it has been determined where some problems exist and how to work around them. The benchmark programs all are designed to perform the same animation task. This task is very simple to allow quick program writing and debugging. The animation task is to create a horizontal line in the upper third of the video and move it to the bottom third. No attempt will be made to make the motion smooth in the benchmark programs since it was immediately apparent that speed is the most evident problem.

Benchmark 1 uses SET and RESET to create the line and move it. Running this simple benchmark will immediately show that these two commands do not have the necessary writing speed to be used for animation purposes. Using the SET and RESET commands allows only one graphics dot to be modified each time these functions are run by the Basic interpreter. As you can see the interpretation speed of Level II Basic is not fast enough for animation programs using only SET and RESET.

```
100 REM BENCHMARK #1-ANIMATION USING SET/RESET
110 REM
120 CLS
130 FOR Y=1 TO 40
140 FOR X=24 TO 50
150 SET(X,Y)
160 NEXT X
170 FOR X=24 TO 50
180 RESET(X,Y-1)
190 NEXT X: NEXT Y
200 STOP
```

How can the writing speed be improved? One method is to write more graphics dots for each Basic command required. The CHR$(n) command could be used to create graphics characters equivalent in size to the alphanumeric characters. This graphics character consists of 6 graphics dots in a 2 X 3 matrix. Using CHR$(n) will write 6 graphics dots each time the Basic interpreter executes this command so it appears that a 6:1 improvement in writing speed

Add POWER to your TRS-80™

SOFTWARE by MiProg

XEDIT, a high powered compact disk based editor designed for the TRS-80™ 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 TRSDOS™ 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 TRSDOS™ 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 TRSDOS™ 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-80™ 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-80™. 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 continential U.S.A., add $3.50 postage and handling.

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

Circle 32

should be noted. Running benchmark 2 will show you the improvement obtained with CHR$(n).

```
100 REM BENCHMARK #2 - ANIMATION USING CHR$(N)
110 REM
120 CLS
130 FOR Y=1 TO 13: Y1=Y*64: Y2=Y1-64
140 FOR X=12 TO 24
150 PRINT@ X+Y1, CHR$(131);:

160 PRINT@ X+Y2, CHR$(128);
170 NEXT X
180 FOR X=12 TO 24
190 PRINT@ X+Y1, CHR$(140);
200 NEXT X
210 FOR X=12 TO 24
220 PRINT@ X+Y1, CHR$(176);
230 NEXT X: NEXT Y
240 STOP
```

In this example, for each increase in the Y value, three lines are written and erased. First, lines 140-170 write a line in the top elements of the matrix and erase the line in the bottom elements of the matrix above the new line. Second, lines 180-200 write a line in the middle elements of the matrix and in doing this, erase the line in the top elements. Third, lines 210-230 write a line in the bottom elements of the matrix erasing the middle elements. Graphics character 128 is a blank.

Benchmark 2 is an improvement over benchmark 1, but it is still much too slow (slower than the rate at which text is written to the video). What is the difference which makes it possible to write text so much faster? First, look at line 180-200 in benchmark 2. To write one graphics character, (six graphics dots), it is necessary to interpret five basic commands. Two of these create a FOR...NEXT loop, commonly used for delay. Now look at the simple PRINT statement:

PRINT "THE QUICK BROWN FOX JUMPED OVER THE LAZY DOG'S BACK"

Only one Basic command and 51 alphanumeric characters are written. No wonder text is faster!

Since there are no graphics keys on our keyboard, it is not possible to insert graphics characters into the print statement. By storing the data to be printed, we can still write the same 51 characters with only a slight increse in interpreter overhead:

A$="THE QUICK BROWN FOX JUMPED OVER THE LAZY DOG'S BACK"

The advantage to be noted here is that there are several commands to operate on strings and none to operate on "xxx". There exists a simple way to build strings using CHR$(n):

A$=CHR$(128)+CHR$(131)+CHR$(140)+CHR$(176)

This string can now be written to the video rapidly. It will take some time to create the strings but this can be done during the program initialization, and then output to the video as needed. Benchmark 3 shows a method of loading and writing the strings.

```
100 REM BENCHMARK #3 - ANIMATION USING STRINGS
110 REM
120 CLS
130 CLEAR 100
140 A$="": B$="": C$="": D$=""
150 FOR J=12 TO 24
160 A$=A$+CHR$(131)
170 B$=B$+CHR$(140)
180 C$=C$+CHR$(176)
190 D$=D$+CHR$(128)
200 NEXT J
210 FOR Y=1 TO 13: Y1=Y*64+12: Y2=Y1-64
220 PRINT@ Y1,A$;
230 PRINT@ Y2,D$;
240 PRINT@ Y1,B$;
250 PRINT@ Y1,C$;
260 NEXT Y
270 STOP
```

At last! Graphics writing speed that will allow animation. For large or complex animation this writing speed is not fast enough, but for simple animation it will be fine. It appears that further increases in writing speed will require the use of assembly language programs, or at least assembly language subroutines. The advantages of Basic over assembly language for the average user makes this an acceptable compromise.

Although benchmark 3 is capable of filling graphics characters into strings, it is obvious that considerable program memory will be required to store a matrix of different characters, since "+CHR$(n)" is required for each graphics character (2 X 3 element). A more efficient method of creating these strings is found in example 1. This example uses READ and DATA to enter the graphics characters into an array of strings.

```
100 REM EXAMPLE #1 - GRAPHIC STRINGS FROM DATA
      STATEMENTS
110 REM
120 CLS
130 CLEAR 100
140 DIM A$(3)
150 FOR I=0 TO 3
160 A$(I)=""
170 FOR J=1 TO 12
180 READ A: A$(I)=A$(I)+CHR$(A)
190 NEXT J: NEXT I
200 DATA 131,131,131,131,131,131,
      131,131,131,131,131,131
210 DATA 140,140,140,140,140,140,
      140,140,140,140,140,140
220 DATA 176,176,176,176,176,176,
      176,176,176,176,176,176
230 DATA 128,128,128,128,128,128,
      128,128,128,128,128,128
240 FOR Y=1 TO 13: Y1=Y*64+12: Y2=Y1-64
250 PRINT@ Y1,A$(0);
260 PRINT@ Y2,A$(3);
270 PRINT@ Y1,A$(1);
280 PRINT@ Y1,A$(2);
290 NEXT Y
300 STOP
```

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 filenaming, 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 explainations 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 compatable 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

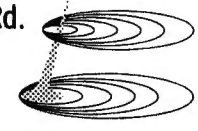
This is an all new concept for this type of game, and compares to the others like chess compares to checkers. ULTRA-TREK is a complex, logical game, intended for the serious contestant. It is doubtful that you will ever master this game, but you will certainly enjoy trying! This program requires a TRS-80 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.

Circle 33

Example 1 runs the same as benchmark 3, but provides a much easier method to load the graphic string array.

A more efficient method of loading the string variable is "packed strings", and is outlined in example 2. This method is memory efficient as well, and consumes less initialization time. Due to the complexity of the operations required this method is suggested for the experienced and patient programmer.

```
100 REM EXAMPLE #2 -GRAPHICS W/ PACKED STRINGS
110 REM
120 CLS
130 DIM A(3)
140 REM TYPE THE NEXT 4 LINES EXACTLY AS SHOWN
150 A$(0)="ABCDEFGHIJKL"
160 A$(1)="ABCDEFGHIJKL"
170 A$(2)="ABCDEFGHIJKL"
180 A$(3)="ABCDEFGHIJKL"
190 REM THE LINE LOCATOR STARTS AT LINE 30000
      THE GRAPHICS PACKER #1 STARTS AT 30100.
200 REM RUNNING THE GRAPHICS PACKER #1 WILL
    CHANGE LINES 150-180 TO:
 A$(0)="SETSETSETSETSETSETSETSETSETSETSET"
 A$(1)="LETLETLETLETLETLETLETLETLETLETLET"
 A$(2)="DEFDEFDEFDEFDEFDEFDEFDEFDEFDEFDEF"
 A$(3)="ENDENDENDENDENDENDENDENDENDENDEND"
210 FOR Y=1 TO 13: Y1=Y*64+12: Y2=Y1-64
220 PRINT@ Y1,A$(0);
230 PRINT@ Y2,A$(3);
240 PRINT@ Y1,A$(1);
250 PRINT@ Y1,A$(2);
260 NEXT Y
270 STOP
30000 REM LINE LOCATOR
30010 REM
30020 C=PEEK(16548) + PEEK(16549) * 256
30030 N=PEEK(C+0)+(PEEK(C+1)*256)
30040 M=PEEK(C+2)+(PEEK(C+3)*256)
30050 PRINT"LINE #";M,"MEMORY ADDRESS -";C;
30060 C=N: INPUT X: GOTO 30030
30100 REM GRAPHICS PACKER #1
30110 REM
30120 INPUT"LINE ADDRESS ";S
30130 FOR J=11 TO 22
30140 X=PEEK(S+J): PRINT"NOW - ";CHR$(X),X,;
30150 INPUT"GRAPHIC VALUE ";X
30160 POKE S+J,X
30170 NEXT J
30180 GOTO 30120
```

In example 2 there are actually three programs:

1. Animation program - this is where the final result of this example will be found. Lines 140 through 180 are initially entered as dummy statements. *It is very important that they be entered exactly as shown,* because the graphics packer assumes that the dummy characters within the quotation marks is offset from the start of the line by a fixed amount. The addition or deletion of even a *single space* will cause the graphics packer to place its data in the wrong loaction. After running the graphics packer you will find these lines have been modified and they will look like a string of Basic command words. The result of this operation is not editable.

2. Line location program - this program will locate the memory addresses required in the graphics packer. It displays the source line number and the memory address where it starts. Each time *ENTER* is pressed the next source line data will be displayed.

3. Graphics packer 1 program - this program is the workhorse for this method. It asks for the memory address of the line to be modified, calculates the offset to the first quote mark, displays the character it found for your verification, requests the graphics character desired, and places it in that location. This process continues until finishing the selected line. Note: This program is designed for a given offset from the start of the line and for a given number of characters. Any attempts to use this in any other manner may result in failure and possible loss of any program in memory. During the development of this program I destroyed several hours of work. I recommend that prior to running this program insure that you have saved a copy of everything in memory that you may need.

Entering the example is done in several phases and the instructions must be carefully followed:

1. Enter the source as shown in the listing. Lines 190 and 200 can be omitted since they do not affect the operations that follow.

2. Run the line locator.

RUN 30000 The program will display a line number and address each time you press *ENTER*. Notice the addresses shown here are samples. Different combinations of source program material in memory will cause the addresses to vary.

```
EXAMPLE #2 RUNNING UNDER NEWDOS80
LINE # 100      MEMORY ADDRESS - 27206 ?
LINE # 110      MEMORY ADDRESS - 27251 ?
LINE # 120      MEMORY ADDRESS - 27257 ?
LINE # 130      MEMORY ADDRESS - 27263 ?
LINE # 140      MEMORY ADDRESS - 27274 ?
LINE # 150      MEMORY ADDRESS - 27319 ?
LINE # 160      MEMORY ADDRESS - 27344 ?
LINE # 170      MEMORY ADDRESS - 27369 ?
LINE # 180      MEMORY ADDRESS - 27394 ? _
```

3. Run the graphics packer #1.

RUN 30100 This program will ask for the line address to be modified.

LINE ADDRESS? 18449 *(ENTER)* Enter the memory address that you obtained from the line locator for line 150. *Do not use the address shown here.*

```
LINE ADDRESS ? 27319
NOW - A          65          GRAPHIC VALUE ? 131
NOW - B          66          GRAPHIC VALUE ? 131
NOW - C          67          GRAPHIC VALUE ? 131
NOW - D          68          GRAPHIC VALUE ? 131
```

SOFTWARE UNLIMITED

presenting the LARGEST SELECTION OF SOFTWARE EVER ASSEMBLED...

the best available on the market today for TRS-80® Computers

at SUPER DISCOUNT PRICES!

AVALON HILL

- ☐ MIDWAY 13.50
- ☐ NUKE WAR 13.50
- ☐ PLANET MINERS 13.50
- ☐ CONVOY RAIDER 13.50
- ☐ B1 BOMBER 13.50
- ☐ LORDS OF KARMA 18.00

PROGRAMMERS GUILD

- ☐ DREADNAUGHT 13.55
- ☐ DREADNAUGHT [D] 17.95
- ☐ DUTCH GOLD [D] 17.95
- ☐ DUTCHMANS GOLD 13.55
- ☐ ELECTRA DRAW 2.0 ..... 17.95
- ☐ SPIDER MOUNTAIN 13.55
- ☐ SPIDER MOUNTAIN [D] 17.95
- ☐ THUNDER 13.55

APPARAT

- ☐ NEWDOS/80 ..... 125.00

ADVENTURE INTERNATIONAL

- ☐ ADVENTURE HINT BOOK 7.95
- ☐ ADVENTURE #0 ..... 6.25
- ☐ ADVENTURE (1,2,3) [D] 35.95
- ☐ ADVENTURE (4,5,6) [D] 35.95
- ☐ ADVENTURE (7,8,9) [D] 35.95
- ☐ ADVENTURE #10 [D] 18.95
- ☐ ADVENTURE (specify 1-10) 13.55
- ☐ MEAN CHECKERS MACHINE 17.95
- ☐ MEAN CHECKERS MACHINE [D] 22.95
- ☐ DR CHIPS 13.55
- ☐ DR CHIPS [D] 17.95
- ☐ INTER FICTION SAMPLER [D] 13.55
- ☐ INTER-LOCAL CALLS [D] 17.95
- ☐ INTER-TWO HEADS [D] 17.95
- ☐ INTER-IMPETUOUS [D] 17.95
- ☐ KID-VENTURE 1 ..... 13.55
- ☐ STAR TREK 3.5 ..... 13.55
- ☐ STAR TREK 3.5 [D] 17.95
- ☐ ZOSSED IN SPACE 13.55
- ☐ MACES #1, BARLOG 31.00
- ☐ MACES #1, SISYPHUS 31.00
- ☐ LUNAR LANDER 13.55
- ☐ LUNAR LANDER [D] 18.95
- ☐ POKER 13.55
- ☐ GALACTIC TRILOGY [D] 35.95

ACORN SOFTWARE

- ☐ ATERM 17.95
- ☐ SYSTEM SAVERS 13.55
- ☐ TING-TONG 8.95
- ☐ DISASSEMBLER 13.55
- ☐ DISK/TAPE UTILITY 17.95
- ☐ CHECKBOOK [D] 22.95
- ☐ STAR TREK SIMULATION 8.95
- ☐ CODEBREAKER 8.95
- ☐ OPERA THEATER 8.95
- ☐ GAMMON CHALLENGER 13.55
- ☐ BLACKADE 8.95
- ☐ PIGSKIN 13.55
- ☐ STRUCTURED BASIC [D] 26.95
- ☐ ULTRA TREK 13.55
- ☐ SPACE WAR 8.95
- ☐ WARP/LANDER 8.95
- ☐ BASKETBALL [D] 18.95
- ☐ BASKETBALL 13.55
- ☐ DUEL-N-DROIDS [D] 18.95
- ☐ DUEL-N-DROIDS 13.55
- ☐ INVADERS FROM SPACE 13.55
- ☐ INVADERS FROM SPACE [D] 18.95
- ☐ PIGSKIN [D] 18.95
- ☐ PINBALL 13.55
- ☐ PINBALL [D] 18.95
- ☐ QUAD 13.55
- ☐ QUAD [D] 19.95
- ☐ SUPERSCRIPT [D] 28.95

HAYDEN

- ☐ SARGON II 25.00
- ☐ SARGIN II [D] 30.00

AUTOMATED SIMULATION

- ☐ STAR WARRIOR [CASS. OR DISK] 35.95
- ☐ THREE PACK [D] 45.00
- ☐ STARFLEET [CASS, OR DISK] 22.50
- ☐ INVASION [CASS, OR DISK] 22.50
- ☐ APSHAI [CASS, OR DISK] 26.95
- ☐ RYN [CASS, OR DISK] 17.95
- ☐ MORLOC [CASS, OR DISK] 17.95
- ☐ RIGEL [CASS, OR DISK] 22.50
- ☐ HELLFIRE [CASS, OR DISK] 26.95

BIG FIVE SOFTWARE

- ☐ ATTACK FORCE (MODEL 1 OR 3) 14.55
- ☐ GALAXY INVASION (MODEL 1 OR 3) 14.55
- ☐ METEOR MISSION (MODEL 1 OR 3) 9.95
- ☐ SUPER NOVA (MODEL 1 OR 3) 14.55
- ☐ COSMIC FIGHTER (MODEL 1 OR 3) 14.55

COMPUTER SIMULATIONS COMPANY

- ☐ BATTLE OF BULGE-BASTGONES 17.95
- ☐ D-DAY INVASION OF FRANCE 17.95
- ☐ DARK KINGDOM 11.75
- ☐ DOG RACE, COLOR ONLY [C] 5.95
- ☐ ELECTRIC ALARM 4.95
- ☐ EMPIRE STRIKES BACK 13.55
- ☐ GREAT DICTATOR 7.25
- ☐ JEDI KNIGHT 11.75
- ☐ LUNAR ENCOUNTER 11.75
- ☐ MERCENARY 8.95
- ☐ MICRO ARCADE 13.55
- ☐ SHARK, COLOR ONLY [C] 5.95
- ☐ SLOT MACHINE 7.25
- ☐ BATTLE OF BULGE-ST VITH 13.55
- ☐ STAR CRUISERS 13.55
- ☐ TYPE WRITERS 17.95
- ☐ U-BOAT, COLOR ONLY [C] 5.95
- ☐ ORION WAR, COLOR ONLY [C] 5.95
- ☐ POLARIS, COLOR ONLY [C] 5.95

INSTANT SOFTWARE

- ☐ AIR FLIGHT SIMULATION 7.25
- ☐ AIRMAIL PILOT 8.95
- ☐ ALL STARS [D] 31.55
- ☐ ASTEROIDS [D] 17.95
- ☐ ASTEROIDS 13.55
- ☐ BASIC PROGRAMMING ASSISTANT 13.55
- ☐ BALL TURRET GUNNER 8.95
- ☐ BATTLEGROUND 8.95
- ☐ CHECK MANAGEMENT [D] 36.55
- ☐ CHESSMATE-80 ..... 17.95
- ☐ THE COMMUNICATOR 8.95
- ☐ COSMIC PATROL 13.55
- ☐ COSMIC PATROL [D] 17.95
- ☐ DAREDEVIL 8.95
- ☐ DISASSEMBLER 8.95
- ☐ DISK EDITOR [D] 36.55
- ☐ DISK SCOPE [D] 17.95
- ☐ DLDIS [D] 17.95
- ☐ DEVICE DRIVERS 17.95
- ☐ ENHANCED BASIC 22.50
- ☐ ELEC REPRT GEN 8.95
- ☐ FLIGHT PATH 8.95
- ☐ FLYING CIRCUS [D] 36.55

INSTANT SOFTWARE

- ☐ GEOGRAPHY [D] 45.00
- ☐ I.Q. TEST 6.95
- ☐ INVADERS 8.95
- ☐ INVESTORS PARADISE 8.95
- ☐ IRV 22.50
- ☐ IRV [D] 27.55
- ☐ JET FIGHTER PILOT 13.55
- ☐ MIND WARP 8.95
- ☐ MONEY MADNESS 8.95
- ☐ NIGHT FLIGHT 8.95
- ☐ ONE-D MAILING LIST [D] 22.50
- ☐ OTHELLO 8.95
- ☐ PROGRAMMER'S CONVERTER 8.95
- ☐ QSL MANAGER [D] 17.95
- ☐ SANTA PARAVIA FIUMACCIO 8.95
- ☐ SKIRMISH-80 ..... 8.95
- ☐ TERMINAL-80 ..... 22.50
- ☐ TLDIS 13.55
- ☐ UTILITY I 7.25
- ☐ UTILITY II 7.25
- ☐ WINNERS DELIGHT 8.95
- ☐ WORDSLINGER 26.00

PERSONAL SOFTWARE

- ☐ CCA MGMT-TRS-80 ..... 65.00

QUALITY SOFTWARE

- ☐ DEBUG 13.55
- ☐ FASTGAMMON 17.95
- ☐ LOWBALL POKER 10.45
- ☐ BANKSHOT 8.95
- ☐ POKER PETE 10.45
- ☐ RUMMY MASTER 10.45
- ☐ 3D TIC TAC TOE 13.55
- ☐ QS LIGHT PEN 17.95
- ☐ SKETCH 80 ..... 13.55

STRATEGIC SIMULATIONS

- ☐ COMPUTER BISMARCK [D] 51.50
- ☐ COMPUTER BISMARCK 42.00

SUB-LOGIC

- ☐ T80-FS1 FLIGHT SIMULATOR 22.00
- ☐ 3D GRAPHICS 26.50

BOTTOM SHELF

- ☐ ANALYSIS PAD [D] 90.00
- ☐ BASIC TOOLKIT 17.80
- ☐ BUSINESS MAIL [D] 100.00
- ☐ CHECKBOOK II [D] 44.50
- ☐ CHECK REGISTER [D] 67.00
- ☐ DATA MANAGER [D] 67.00
- ☐ HEAD CLEANER [D] 17.00
- ☐ INFO SYSTEM [D] 44.50
- ☐ LIBRARY 100 ..... 44.50
- ☐ SYSTEM DOCTOR [D] 35.50
- ☐ SYSTEM DOCTOR 26.00

MICROSOFT SOFTWARE

- ☐ ADVENTURE [D] 25.50
- ☐ ASSEMBLY DEVELOPMENT [D] 80.00
- ☐ BASIC COMPILER [D] 175.00
- ☐ EDITOR/ASSEMBLER 25.00
- ☐ FORTRAN COMPILER [D] 80.00
- ☐ LEVEL III BASIC 44.00
- ☐ MuMATH [D] 64.00
- ☐ OLYMPIC DECATALON [CASS OR DISK] 20.00
- ☐ TYPING TUTOR 13.55

If you don't see it listed, write...we probably have it in stock!

Check program desired.
Complete ordering information
and mail entire ad.
Immediate Shipments from stock.

KEY:
C-TRS-80 Color
D-on Disc
If not marked - Cassette

Ship the above programs as checked to:

Mr /Mrs _____

Address _____

City _____

State _____ Zip _____

I have a _____ name of Computer

with _____ K memory

Number of Programs Ordered ....

Amount of order .....

N Y residents add Sales Tax ....

Add shipping anywhere in the U S 2.00

Total amount enclosed .....

Charge my ☐ Master Charge ☐ Visa

Signature _____

Card No _____ Expires _____

Mail to: **DIGIBYTE SYSTEMS CORP.**

31 East 31st Street, New York, N.Y. 10016
(212) 889-8975

TRS-80 is a registered trademark of TANDY CORP.

Prices subject to change without notice.

Circle 34

```
NOW - E          69        GRAPHIC VALUE ? 131
NOW - F          70        GRAPHIC VALUE ? 131
NOW - G          71        GRAPHIC VALUE ? 131
NOW - H          72        GRAPHIC VALUE ? 131
NOW - I          73        GRAPHIC VALUE ? 131
NOW - J          74        GRAPHIC VALUE ? 131
NOW - K          75        GRAPHIC VALUE ? 131
NOW - L          76        GRAPHIC VALUE ? 131
LINE ADDRESS ? 27344
NOW - A          65        GRAPHIC VALUE ? 140
NOW - B          66        GRAPHIC VALUE ? _
```

Enter the graphics character value (decimal). If the "now" character is not the character wanted. Hit *BREAK* and start over if you did something wrong. The graphics values for line 150 is 131; line 160 is 140, line 170 is 176 and line 180 is 128.

```
150 A$(0)="SETSETSETSETSETSETSETSETSETSETSET"
160 A$(1)="LETBCDEFGHIJKL"
170 A$(2)="ABCDEFGHIJKL"
180 A$(3)="ABCDEFGHIJKL"
```

If this is what you find, you have done everything correctly. Note: The Basic commands listed are really not there. Basic thinks that the graphics characters are compressed commands and when listing the line it expands them as Basic commands. The 12 characters shown take 12 bytes of memory.

I have described a method to rapidly write graphics and to efficiently create Basic source programs for these graphics. All that remains now is to define an example set of graphics for animation, convert these to packed graphics strings, and write the necessary subroutines to provide the motion.

For a practical, yet simple example, I have created an Android. He is capable of walking left and right, looks up and down and moves his hands and arms. The Android is divided into three parts:

HEAD - two strings 12 X 3 points each
BODY - two strings - 12 X 3 points each
FEET - one string - 12 X 3 points each

This results in the Android contained within a 12 X 15 point matrix. Since the Android has been defined as three separate parts, it is possible to combine different selections of head, body and feet to create the animation desired.

Nine different positions are used for each part of the android. To creat motion from right to left, the following sequence is used:

HEAD 0,	BODY 0,	FEET 0
HEAD 1,	BODY 1,	FEET 1
HEAD 0,	BODY 0,	FEET 2
HEAD 1,	BODY 1,	FEET 2
HEAD 1,	BODY 1,	FEET 3

Motion in the opposite direction uses:

HEAD 8,	BODY 8,	FEET 8
HEAD 7,	BODY 7,	FEET 7
HEAD 8,	BODY 8,	FEET 6
HEAD 7,	BODY 7,	FEET 5

In addition to his 8 profile views the Android has five basic front views:

	HEAD	BODY	FEET
Half turn left	2	2	1
Looking front	3	3	4
Looking up	4	4	4
Looking down	5	5	4
Half turn right	6	6	7

The graphics must now be converted to graphic strings. The first step is to convert drawings to graphics characters. This is done by looking up each 2 X 3 character matrix of each drawing in the graphic character chart in appendix C of the Level II Basic Reference Manual. Next, enter these graphic characters into the initialization subroutine.

Two methods are shown in the "Arnold the Android" program listing. The first method, lines 24000-24430 use READ/DATA for the graphic string definition. This method allows straightforward entry by placing the graphic character values into DATA statements. (Using this method enter only lines 10000-24400 and 25280-25520. Change line 10030 as indicated in the remark in line 10020.)

The second method uses packed graphics strings. This method is more complex to enter since lines 25040-25260 cannot be directly entered. These lines must be entered in the dummy form shown in the listing and then modified to their final form. To do this proceed as follows:

Enter lines 30000 to 31130 (line locator and graphics packer #2) and lines 25040 to 25260 exactly as shown in the listing.

RUN30000 (line locator) and write down the address for each line from 25040 to 25260.

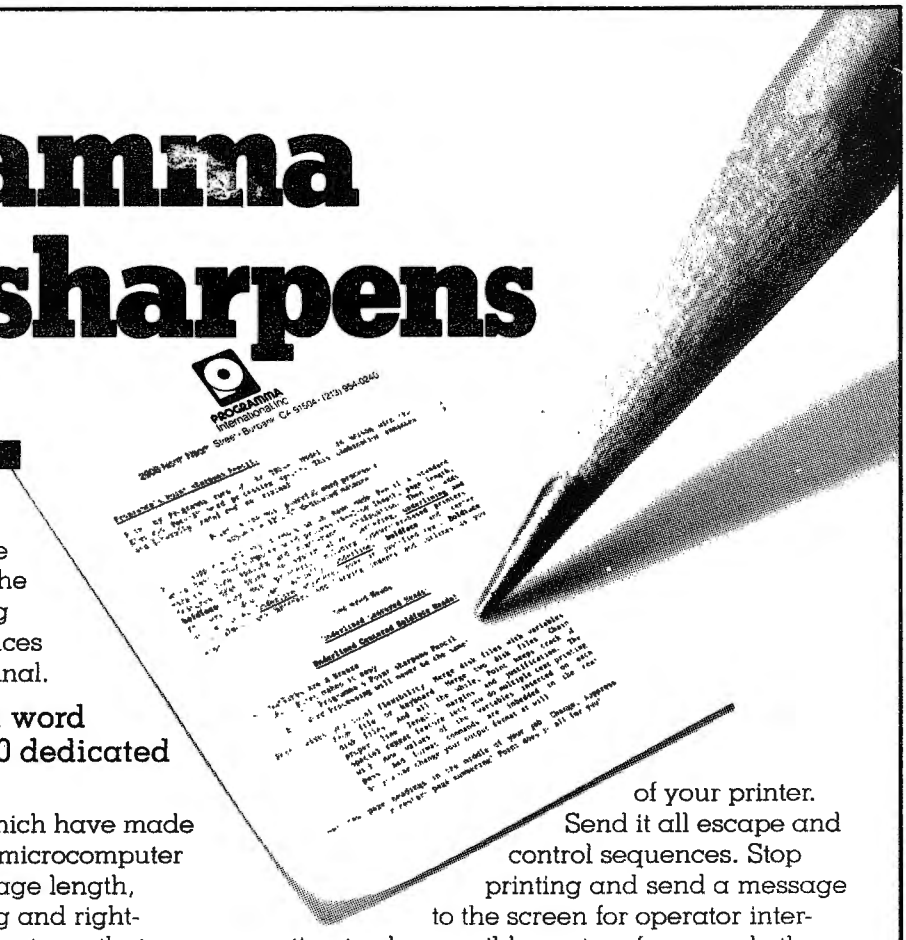
RUN31000 (graphics packer #2). Using the line addresses you just wrote down, enter the graphic values from the data statements in lines 24150 to 24410 of the listing.

Now enter the remaining lines from the program listing (lines 10000 - 25030 and 25270 - 25520). After verifying that lines 25040-25260 are correct, delete lines 30000 - 31130. Note that lines 25040 - 25260 cannot be edited.

The basic activity program, lines 10040-10160, has been provided as an example of how to control the Android. Using the subroutines provided, the Android can be put to use to add some life to your programs.

```
10000 ' ARNOLD THE ANDROID
        5-APR-80
        BY Gary Sanderson
10010 CLEAR 400  'CAN BE REMOVED IF PACKED
      STRING VARIABLES ARE USED
10020 REM For initialization from DATA
          statements change line # 10030 to
          "GOSUB 24010: CLS"
10030 GOSUB 25010: CLS
10040 POKE 16396,23 ' Disable break key
```

Programma Point™ sharpens Pencil.



Point by Programma runs on the TRS-80* Model I in unison with the Electric Pencil** word processing system. This combination enhances and literally transforms the original.

Point is the most powerful word processor since the $20,000 dedicated machines.

Point supports all the formats which have made Pencil a standard among basic microcomputer word processors—line length, page length, adjustable margins, line spacing and right-hand justification. Then it adds features that bring the system alive! Centering, underlining and boldface with most printers including typewriter-based printers.

You can actually underline, boldface and center all at once. Underline anywhere, even in justified text. Boldface with any printer. Add hanging indents and outlines as you wish.

Programma's Point gives you total flexibility. Merge disk files with variables from file or keyboard. Merge two disk files. Chain disk files. And all the while, Point keeps track of proper line length, margins and justification. The special repeat feature lets you do multiple text printing with new values of the variables inserted on each pass. And format commands are embedded in the text so you can change your output format at will.

Type as fast as humanly possible without losing characters—ever!

Even during wrap-around, Point supports loss-free, high-speed entry. You can halt, if you like, at the end of each page for single sheet insertion. While you're typing, you have full upper and lower case capability with any lower case mod—or none at all. You need no additional control key to maintain full Pencil control functions.

Point keeps you comfortably in command of your printer. Send it all escape and control sequences. Stop printing and send a message to the screen for operator intervention to change ribbons, typefaces and other housekeeping.

If you have hardware problems, Point is ready. There is a simple warm-start re-entry so you can recover files in the event of hardware crash or to let you install protectable machine language routines in high memory.

If you're basically happy with your Electric Pencil—but need a first-cabin word processor—the new Programma Point is exactly what you want. Move from ordinary limited word processing to the ultimate easily, quickly. Now available at your computer store or use this coupon for fast turn-around! An extraordinary bargain at only $49.95.

In California, call Programma at (213) 954-0240. Outside California, call toll-free (800) 423-2978.

Programma International, Inc.
2908 N. Naomi Street, Burbank, CA 91504

RUSH me the brand new Programma Point software system. Guaranteed to be fully compatible with Electric Pencil (disk version). Enclosed is $49.95 plus $1.50 for handling and postage. California residents add $3 sales tax. [ ] Check enclosed
[ ] Master Charge
or Visa number _____ Expiration date _____
Signature _____
Name _____
Street Address _____
City _____ State _____ Zip _____

PROGRAMMA International, Inc.

*TRS-80 is a trademark of Tandy Corp.
**Electric Pencil is a trademark of Michael Shrayer Software, Inc

```
10050 ' Draw walkway and roam around
10060 FOR C=0 TO 318: PRINT@ 704+C,CHR$(191);:
      NEXT C
10070 POKE 16383,191
10080 PRINT@910,"RADIO SHACK * TRS-80 * LEVEL II
10090 FOR C=0 TO 32: PRINT @ 846+C,CHR$(143);:
      NEXT C
10100 X1=29: Y=8 : I=3: J=4: K=4: GOSUB 25430:
      GOSUB 25280
10110 X2=RND(55)+1: X3=ABS(X1-X2)
10120 IF X3>20 GOSUB 10140
10130 GOSUB 25500: GOTO 10110
10140 FOR C=0 TO 10
10150 IF RND(4)=2 THEN J=RND(5)+1: GOSUB 25430
10160 IF RND(4)=2 THEN I=RND(7): GOSUB 25430
10170 FOR H=0 TO 40: NEXT H: NEXT C
10200 RETURN

24000 REM THE FOLLOWING LINES ARE USED FOR
       INITIALIZATION FROM DATA STATEMENTS
24010 DIM H$(8,1),B$(8,1),F$(8)
24040 N=8
24050 FOR I=0 TO N: FOR J=0 TO 1: H$(I,J)="":
      FOR K=1 TO 6
24060 READ A: H$(I,J)=H$(I,J)+CHR$(A)
24070 NEXT K: NEXT J: NEXT I
24080 FOR I=0 TO N: FOR J=0 TO 1: B$(I,J)="":
      FOR K=1 TO 6
24090 READ A: B$(I,J)=B$(I,J)+CHR$(A)
24100 NEXT K: NEXT J: NEXT I
24110 N=8
24120 FOR I=0 TO N: F$(I)="": FOR K=1 TO 6
24130 READ A: F$(I)=F$(I)+CHR$(A)
24140 NEXT K:  NEXT I
24150 DATA 128,128,181,176,176,128,
           128,142,174,191,159,133
24160 DATA 128,170,176,176,144,128,
           136,141,191,191,143,128
24170 DATA 128,170,176,181,144,128,
           128,141,179,190,143,128
24180 DATA 128,160,181,186,144,128,
           128,143,182,185,143,128
24190 DATA 128,160,164,152,144,128,
           128,143,189,190,143,128
24200 DATA 128,160,180,184,144,128,
           128,143,155,167,143,128
24210 DATA 128,160,186,176,149,128,
           128,143,189,179,142,128
24220 DATA 128,160,176,176,149,128,
           128,143,191,191,142,132
24230 DATA 128,176,176,186,128,128,
           138,175,191,157,141,128
24240 DATA 128,168,191,191,191,128,
           136,135,170,191,159,128
24250 DATA 128,190,191,191,149,128,
           142,129,191,191,133,128
24260 DATA 168,151,183,191,171,148,
           186,133,159,175,186,133
24270 DATA 184,135,183,187,139,180,
           130,173,159,175,158,129
24280 DATA 168,151,183,187,171,148,
           138,181,159,175,186,133
24290 DATA 168,151,183,187,171,148,
           186,133,159,175,138,181
24300 DATA 168,151,191,187,171,148,
           138,181,159,175,138,181
24310 DATA 128,170,191,191,189,128,
           128,138,191,191,138,181
24320 DATA 128,191,191,191,148,128,
           128,175,191,149,175,144
24330 DATA 128,176,150,160,185,128
24340 DATA 128,176,181,186,128,128
24350 DATA 128,128,176,191,128,128
24360 DATA 128,176,166,185,128,128
24370 DATA 128,176,149,170,176,128
24380 DATA 128,128,182,153,176,128
24390 DATA 128,128,191,176,128,128
24400 DATA 128,128,181,186,176,128
24410 DATA 128,182,144,169,176,128
24420 RETURN
24430 REM. . . . . .
25000 REM. . . . . .
       For initialization with packed strings,
       use lines 25010-25170 and change line
       # 10030 to read "GOSUB 25010: CLS"
25010 REM   LINES 25040-25260 MUST BE ENTERED
      EXACTLY AS SHOWN. THE LINE LOCATOR AND
      GRAPHICS PACKER #2 MUST BE USED AS
      DESCRIBED IN THE ARTICLE. USE THE DATA
      FROM THE ABOVE DATA STATEMENTS
25030 DIM H$(8,1),B$(8,1),F$(9)
25040 H$(0,0)="ABCDEF": H$(0,1)="ABCDEF"
25050 H$(1,0)="ABCDEF": H$(1,1)="ABCDEF"
25060 H$(2,0)="ABCDEF": H$(2,1)="ABCDEF"
25070 H$(3,0)="ABCDEF": H$(3,1)="ABCDEF"
25080 H$(4,0)="ABCDEF": H$(4,1)="ABCDEF"
25090 H$(5,0)="ABCDEF": H$(5,1)="ABCDEF"
25100 H$(6,0)="ABCDEF": H$(6,1)="ABCDEF"
25110 H$(7,0)="ABCDEF": H$(7,1)="ABCDEF"
25120 H$(8,0)="ABCDEF": H$(8,1)="ABCDEF"
25130 B$(0,0)="ABCDEF": B$(0,1)="ABCDEF"
25140 B$(1,0)="ABCDEF": B$(1,1)="ABCDEF"
25150 B$(2,0)="ABCDEF": B$(2,1)="ABCDEF"
25160 B$(3,0)="ABCDEF": B$(3,1)="ABCDEF"
25170 B$(4,0)="ABCDEF": B$(4,1)="ABCDEF"
25180 B$(5,0)="ABCDEF": B$(5,1)="ABCDEF"
25190 B$(6,0)="ABCDEF": B$(6,1)="ABCDEF"
25200 B$(7,0)="ABCDEF": B$(7,1)="ABCDEF"
25210 B$(8,0)="ABCDEF": B$(8,1)="ABCDEF"
25220 F$(0)  ="ABCDEF": F$(1)  ="ABCDEF"
25230 F$(2)  ="ABCDEF": F$(3)  ="ABCDEF"
25240 F$(4)  ="ABCDEF": F$(5)  ="ABCDEF"
25250 F$(6)  ="ABCDEF": F$(7)  ="ABCDEF"
25260 F$(8)  ="ABCDEF": DUMMYS$="ABCDEF"
25270 RETURN
      '. . . . . .
25280 '
      Wait some
25290 FOR W=0 TO 30: NEXT W: RETURN
25300 '
      Turn front to left
25310 I=2: J=2: K=1: GOSUB 25430: GOSUB 25280
      I=0: J=0: K=0: GOSUB 25430: RETURN
25320 '
      Walk left ( X1=start, X2=end, Y=line )
25330 I=0: J=0: K=0: GOSUB 25430:
      I=1: J=1: K=1: GOSUB 25430: X1=X1-1:
      I=0: J=0: K=2: GOSUB 25430:
```

```
      I=1: J=1: K=3: GOSUB 25430: X1=X1-1:
      IF X1>X2 GOTO 25320 ELSE RETURN
25340 '
        Turn left to front
25350 I=2: J=2: K=1: GOSUB 25430: GOSUB 25280:
      I=3: J=4: K=4: GOSUB 25430: GOSUB 25280:
      RETURN
25360 '
        Turn front to right
25370 I=6: J=6: K=7: GOSUB 25430: GOSUB 25280:
      I=8: J=8: K=8: GOSUB 25430: GOSUB 25280:
      RETURN
25380 '
        Walk right
25390 I=8: J=8: K=8: GOSUB 25430:
      I=7: J=7: K=7: GOSUB 25430: X1=X1+1:
      I=8: J=8: K=6: GOSUB 25430:
      I=7: J=7: K=5: GOSUB 25430: X1=X1+1:
      IF X1<X2 GOTO 25380 ELSE RETURN
25400 '
        Turn rignt to front
25410 I=6: J=6: K=7: GOSUB 25430: GOSUB 25280:
      I=3: J=4: K=4: GOSUB 25430: RETURN
25420 '
     'Display ANDRIOD
     'I - head, J - body, K for feet
25430 L=X1+Y*64:
 PRINT@ L     ,H$(I,0);: PRINT@ L+ 64,H$(I,1);:
 PRINT@ L+128,B$(J,0);: PRINT@ L+192,B$(J,1);:
 PRINT@ L+256,F$(K);: RETURN
25480 '
25490 ' Determine direction and move.

25500 IF X1<X2 GOTO 25520
25510 GOSUB 25300: GOSUB 25320: GOSUB 25340:
      RETURN
25520 GOSUB 25360: GOSUB 25380: GOSUB 25400:
      RETURN
30000 REM. . . . . .
      LINE LOCATOR
30010 C=PEEK(16548) + PEEK(16549) * 256
30020 N=PEEK(C+0)+(PEEK(C+1)*256)
30030 M=PEEK(C+2)+(PEEK(C+3)*256)
30040 PRINT"LINE #";M,"MEMORY ADDRESS -";C;
30050 C=N: INPUT X: GOTO 30020

31000 REM. . . . . .
       GRAPHICS PACKER #2
31010 ' STRING FORMAT:
        A$(N,M)="ABCDEF": A$(W,Z)="ABCDEF"
    OR: A$(N)  ="ABCDEF": A$(W)  ="ABCDEF"
    *** EXACT SPACING IS IMPORTANT ***
31020 INPUT"LINE ADDRESS ";S
31030 FOR J=13 TO 18
31040 X=PEEK(S+J): PRINT "NOW - ";CHR$(X),X,;
31050 INPUT "GRAPHICS VALUE ";X
31060 POKE S+J,X
31070 NEXT J
31080 FOR J=31 TO 36
31090 X=PEEK(S+J): PRINT "NOW - ";CHR$(X),X,;
31100 INPUT"GRAPHICS VALUE ";X
31110 POKE S+J,X
31120 NEXT J
31130 GOTO 31020
```

●

@NEWS

By Jim Perry

A few years ago I never dreamed that I would earn my living having fun with a TRS-80. If I can do it so can you! There are lots of ways of getting cash from your TRS-80, the main thing is to actually do it - and not procrastinate.

I get my income mainly from writing TRS-80 related material, with a bit of editing and consulting work as well. A couple of years ago my background in printing and publishing led me into the frozen wastes of New Hampshire, and the Wayne Green empire. I became the founding editor of *80 Microcomputing* and then left to live in sunny Southern California, making a pleasant living with my TRS-80.

You don't have to be a literary genius to earn money writing (although it doesn't do any harm). The main things to concentrate on are interest and clarity - don't blind people with buzzwords and jargon, or bore them to tears.

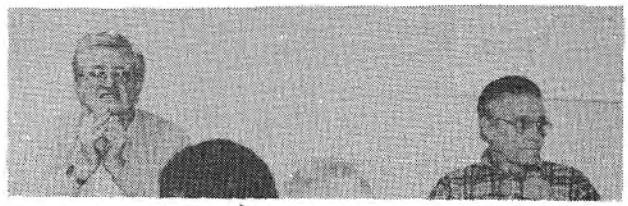
Another way of earning cash with your computer is to sell programs. There are dozens of software publishing companys, all looking for good software on a royalty basis, or you can try and become your own publisher. If you have written some good software that uses the Stringy Floppy then Exatron may be interested in publishing it for you.

Currently Exatron has two methods of publishing software, the first being a standard 20 percent royalty arrangement, the second being the *Exatron Stringy Floppy Owners Association Library*. The library arrangement is a bit like roulette, if a program is selected for the *Wafer Of the Month* you get a $500.00 equipment credit, but unlike roulette you always get at least a $9.95 credit - if your program is accepted into the library. Library programs are given away for the cost of the wafer and duplication, just like in most other user groups. Programs published by Exatron on a royalty basis are sold on a wafer, and you can usually sell them on cassette or disk through another publisher.

No matter how you decide to promote your programs you have to provide documentation. As Michael Commendul (my successor at *80 Microcomputing*) once pointed out, English is a difficult second language for most programmers, hence the terrible standard of documentation for most programs. Remember that there are upwards of 300,000 TRS-80 users, and of the thousands of programs being submitted to publishers only about 5 percent ever make it. The other 95 percent are usually rejected because of poor documentation, or even a complete absence of documentation! Just think how annoyed you would be if you bought a jigsaw puzzle, and the picture was missing!

If you would like further details about the Exatron library, or royalty publishing arrangements, send a business size SASE to Bill Burnham at Exatron, 181 Commercial Street, Sunnyvale, CA 94086.

A third method of cashing in on your computer is to sell Stringy Floppies. To do this you need to become a *Program Chairman* (or woman), and act as a liason between Exatron and potential buyers in your area. Several thousand of you now own Stringys, most of you bought them after reading about them in magazines. However a lot of potential owners want to actually see and use a Stringy before buying one, this is where you can come in to the picture.



This page top: John Onemos (left) and Fred Waters chair a Saturday morning workshop. Bottom: Between 30 and 40 users turn up every week, some say that they actually look forward to the meetings!

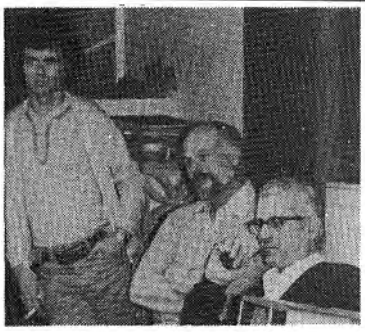
Opposite page, top left: Your friendly author props up a doorway while he wakes up, center is Ron Engdahl workshop coordinator at Exatron. Last, but by no means least, is Bob Howell Snr (inventor of the Stringy Floppy and chairman of Exatron). Top right, a typical programmers den at Exatron (look familiar?), this one belongs to Jim Maynard , who is said to dream in assembly language. Bottom, Exatron World Headquarters, drop in and see them sometime.

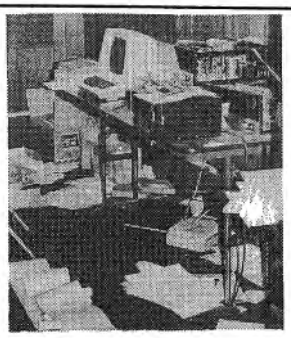
Exatron gets hundreds of reader enquiries every month from its magazine advertising, and wants owners like you to arrange local demonstrations. In my own experience demonstrating Stringy Floppies is like selling water in the Sahara - everyone buys. The product is unique, and excellent. Anyone who still uses cassette can't believe their eyes when they see the Stringy whirr into action, at half the cost of upgrading to disks. A lot of people (myself included) use both Stringy Floppy and disk at the same time. So if you would like to find out more about demonstrating Stringy Floppies send a SASE to Ron Engdahl at Exatron.

Weekly Workshops

If you are in the Sunnyvale area, any Saturday morning, drop into the Owners Association Workshop held at the Exatron factory. The photographs on this page were taken during a recent gathering, anywhere from 30 to 50 people attend from 9.00am until lunch time. You don't have to be a member or owner to turn up, and admission is absolutely free (so is the coffee).

The meetings are usually chaired by John Onemos and Fred Waters (the ESFOA secretary), and have guest speakers plus a general question and answer session. Demonstrations for the curious take place at the end of the meeting, with no high pressure selling (of course you can buy a Stringy or software if you want to!).





Photographs by John Barry.

If you are interested in starting a workshop in your area then contact Fred Waters or Ron Engdahl at Exatron, and they will help get you started. It's a very pleasant and rewarding experience, you meet all sorts of interesting people - doing all sorts of odd things with their Stringys. If you are using your Stringy in a novel project then drop me a line and I'll make you famous by writing about you in a future @NEWS.

New ROMs For Old

If your Stringy is getting on a bit, and has the old 3.2 (or even 3.1) EPROM still in it, then it's about time you upgraded to the current 4.1 ROM. To get your Stringy brought upto date the EPROM and another component need replacement. The cost of the upgrade is a nominal $25.00, which includes giving your unit a thorough 'wash & brush-up' service. All units returned to Exatron, for this upgrade, will be processed and dispatched back to you within 5 days of arrival at Exatron. So you won't be without your unit for very long, suffering with cassettes.

The 4.1 operating system is far superior to the 3.2, and some software written for 4.1 will not work reliably on the older versions. Don't delay, upgrade today!

Where Is Randy Burt ?

Somewhere in America is a frustrated TRS-80 user by the name of Randy Burt. He's frustrated because he is waiting for his Stringy Floppy and some software. Exatron are frustrated because they cashed his check, and then realized that he did not put his address on his order. So if you are Randy, or know him, please call Ruth Howell on the Hotline (800 538 8559) and tell her your address (plus how much your check was for). Moral of this story, always put your address on any mail order - don't rely on your printed check, because most companies seperate the check and order.

One of the more active ESFOA members, who prefers to be known by the name 'Wee Willy', has penned a theme song for use at official functions - I print it without further comment, except that he didn't tell me what tune goes with it!

Hail to ESFOA here we are;
We'll be true whether near or far;
No more cassette corders will we use;
We'll @SAVE our programs and @LOAD them too!

We love our Stringy Floppies, they're for us;
No more volume settings 'n all that fuss;
Our headaches and heartbreaks now are few;
Just throw in a wafer and give it @NEW!

So here's to EXATRON our mentor true;
They gave us Stringy and the RS232;
Our software oh ... it's a delight to run;
Thanks to EXATRON computers are fun.

Slow Disk Software

The standard Stringy Floppy operates like an extremely fast, and reliable, cassette system. Wouldn't it be nice to make it operate like a slow disk system? Tom Wheeler, from Missouri, has released an enhancement to BASIC that does just that. Called ESF BASIC 1.4, the software gives you the ability to name programs, writes a directory file on each wafer and generally makes the Stringy behave like a slow disk. Costing only $25.00, direct from Exatron, this software opens up lots of interesting possibilities. I'll have an indepth review in the next @NEWS.

Son of Visicalc

Any serious user of a microcomputer needs two types of program; a word processor (such as Electric Pencil, and a number processor (Visicalc from Personal Software being excellent). Now there is a rival to Visicalc, written by Dan Haney, called Electric Spreadsheet. Written in BASIC (you don't have to write in assembly language to be fast), it comes with a 50 page manual and allows you to process numbers in dozens of ways. I've been using it for about 2 weeks and will include a complete review in my next column. If you don't want to wait, it's available now for $49.95 direct from Exatron.

Inflation Strikes

For those of you that have been thinking of buying a Stringy Floppy, but kept putting it off, I have some bad news - on April 1st the price of a starter kit went up to $349.50 (plus shipping and tax). However do not despair, the starter kit contents have also gone up! The new improved starter kit contains much more software and a trial subscription to @LOAD (a wafer based magazine), plus a $16.00 credit towards your *80-US Journal* subscription.

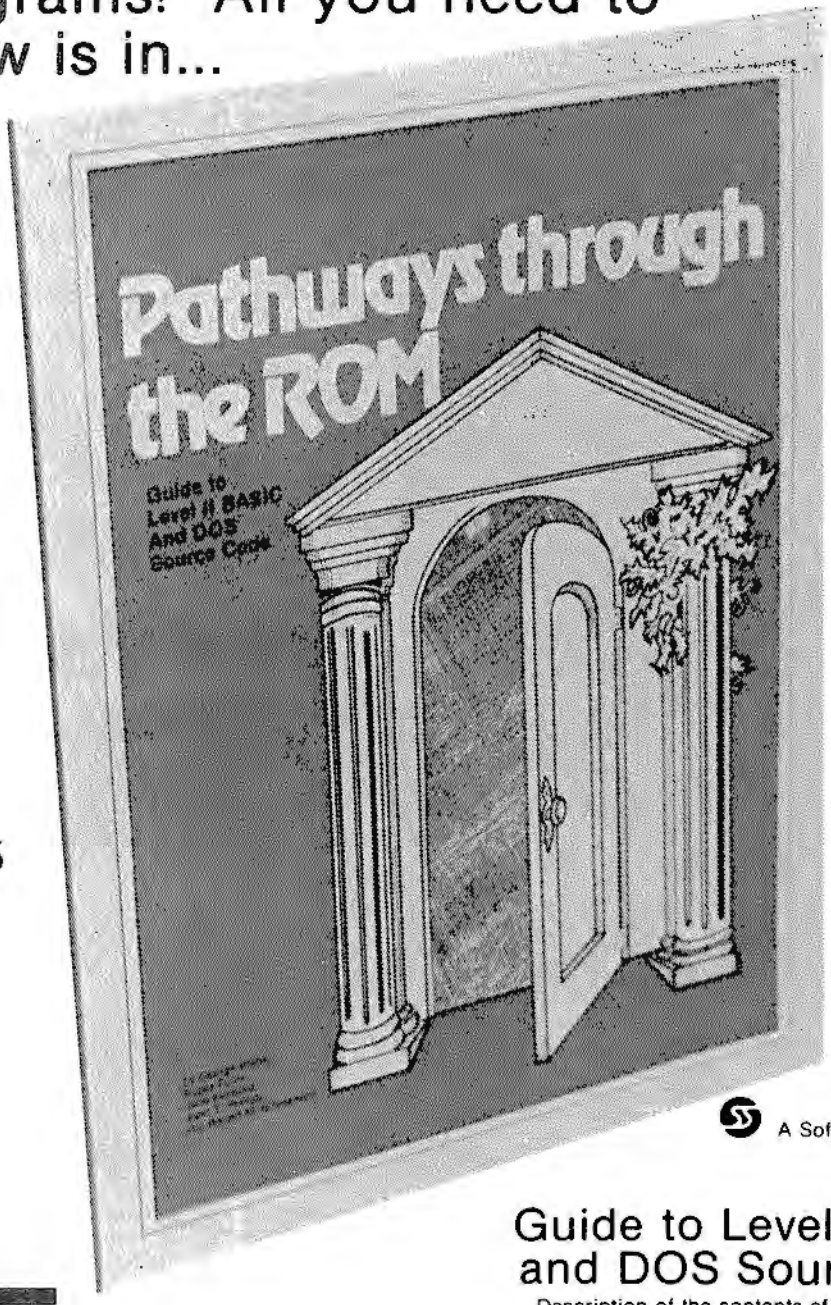
Next Issue

Well I hope you found my debut column interesting; future columns will include detailed software reviews, feedback from you (please!), details on new workshops and anything else that I think you'll find useful and informative.

Feel free to write to me at @NEWS, 1016 W Pine Street, Upland, CA 91786. I look forward to hearing from you.

If you have any questions about Exatron, the Exatron Stringy Floppy or the Exatron Stringy Floppy Owners Association, please call the Exatron Hotline **800-538 8559** or in California 408-737 7111. ●

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...



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

ALL ONLY $19.95
plus $1 shipping

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.

MasterCard VISA

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

The Software Exchange

APL

by Phelps Gates

Now a high-level, scientific programming language for the home computer that doesn't cost $200 or $300. 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.

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 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 and provides additional insight into APL programming.

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 workshapes 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)

LIMITATIONS

Due to the absence of the special APL character set on the TRS-80™, APL-80 uses shifted letters to represent the various APL characters.
In addition to the keyboard 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.

TSE:HARDSIDE

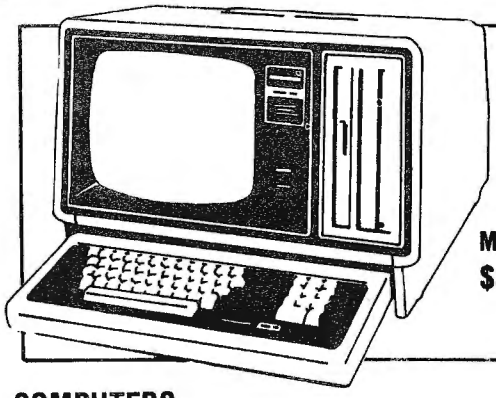
MasterCard VISA

6 South St., Milford, N.H. 03055

ORDER TOLL FREE: 1-800-258-1790

(in NH call 673-5144)

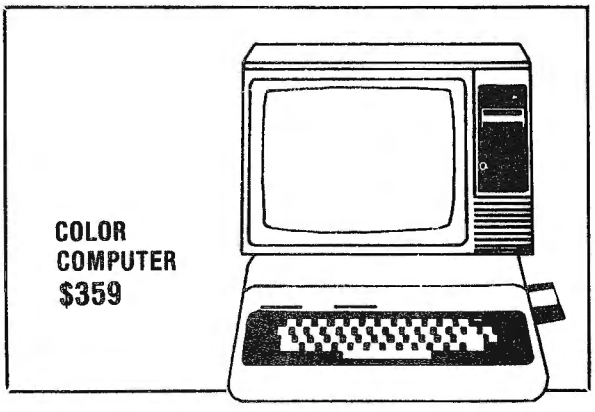
Circle 31



MODEL II
$3599.

COMPUTERS

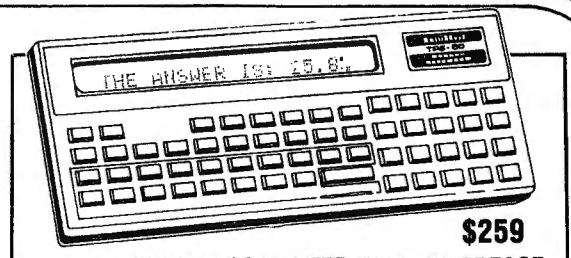
TRS-80 Model II 64K RAM (#26-4002) $3599.00
TRS-80 Model III, 16K RAM (#26-1062) $919.00
TRS-80 Model II, 48K RAM (#26-1062+) $1039.00
TRS-80 Mod. III 48K RS232 2-dr. (#26-1063) $2299.00
TRS-80 Pocket Comp. w/Interface (#26-3501+) . $259
TRS-80 Video Tex (#26-5000) $359.00
TRS-80 Color Computer, 4K RAM (#26-3001) . $359.00
TRS-80 Color Comp. 16K RAM (#26-3001+) . $399.00
TRS-80 Color Comp. Ext. BASIC (#26-3002) . . $529.00



COLOR
COMPUTER
$359

MODEL I DISK DRIVES

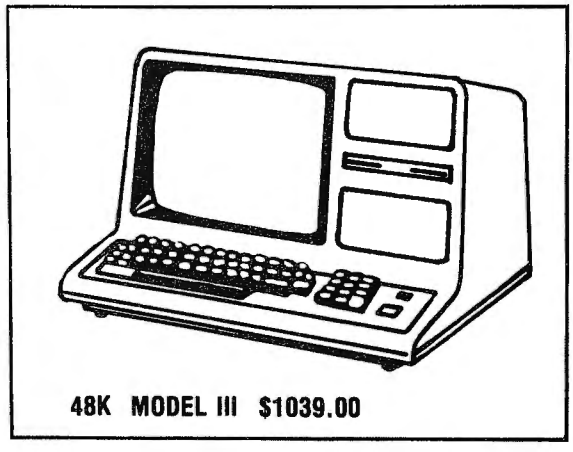
HARDSIDE 40-track Drive (#7-40) $329.00
PERCOM TFD-40 Drive (#7-99) $379.00
PERCOM TFD-100 Drive (#7-100-1) $399.00
HARDSIDE 80-track Drive (#7-80) $449.00
PERCOM Dual TFD-100 Drives (#7-100-2) $799.00
PERCOM Data Separator (#7-03) $29.95
PERCOM Doubler (#7-07) $199.95
HARDSIDE Extender Cable (#7-02) $15.95
HARDSIDE 2-Drive Cable (#7-04) $29.00
HARDSIDE 4-Drive Cable (#7-05) $39.00



$259
POCKET COMPUTER WITH INTERFACE

MODEL I PERIPHERALS

COMM-80 Interface (#4-80) $159.00
CHATTERBOX Interface (#4-81) $239.00
Disk-80 Interface, 16K RAM (#4-82) $339.00
DISK-80+ Interface, 16K RAM (#4-83) $369.00
BUSY BOX Interface (#4-01) $99.95
LYNX Communications Interface (#19-80) . . . $229.00
RS Exp. Interface, 32K RAM (#26-1140-32) . . $399.00
16K Memory Kit, TRS-keyboard (#5-1102-1) . . $59.00
16K Memory Kit, TRS-Interface (#5-1102) $59.00
ORCHESTRA-80 (#15-03) $79.95
Upper/Lower Modification Kit (#15-02) $24.95
CPU Speed-up Modification kit (#15-04) $37.50
Video Reverse Modification kit (#15-05) $23.95
2-port TRS-BUS Extender (#15-12) $29.95
3-port TRS-BUS Extender (#15-13) $39.95
TRS-80 Model I Dust Cover Set (#16-01) $7.95
TRS-80 Model I Carrying Case (#7-201) $109.00
TRS-80 Monitor Carrying Case (#17-202) $84.00
Dual Joysticks for Color Computer (#26-3008) . $24.95
VISTA Model II 8'' Disk Drive 1(#7-4001) $939.00
VISTA Model II 8'' Disk Drive, 3(#7-4002) . . $1795.00
CTR-80A Cassette Recorder & Cable (#26-1206) $59.95
TRS-80 Model III Dust Cover (#16-05) $7.95
TRS-80 Color Dust Cover (#16-06) $7.95



48K MODEL III $1039.00

MasterCard VISA

TSE:HARDSIDE

6 South St., Milford, NH 03055 (603) 673-5144
TOLL FREE OUT-OF-STATE 1-800-258-1790

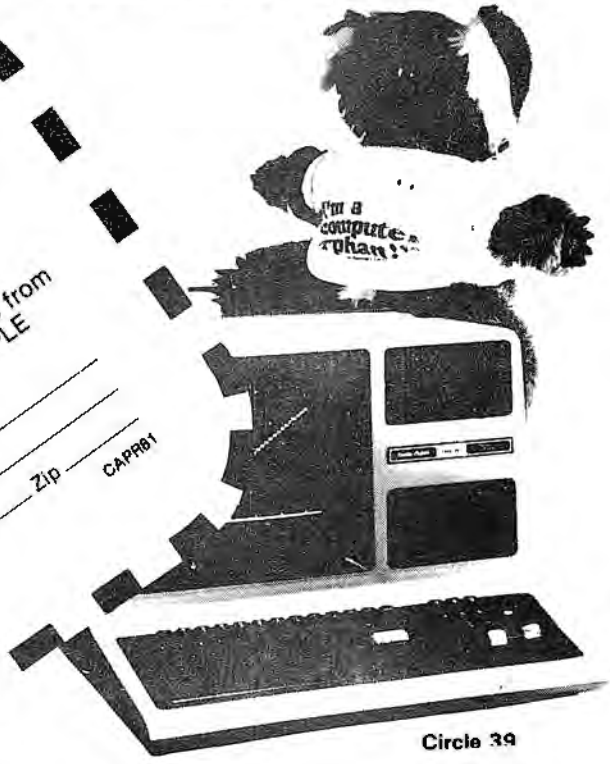
TERMS: Prices and specifications are subject to change. HARDSIDE accepts VISA & MASTERCARD. Certified checks and Money Orders; Personals checks accepted (takes 3 weeks to clear). HARDSIDE pay all shipping charges (within the 48 states) on all PREPAID orders OVER $100.00. On all orders under $100 a $2.50 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. TRS-80 is a trademark of Tandy Corp.

Circle 82

TSE-HARDSIDE HAS IT ALL IN ONE!

How many times have you wished that there was a single source for your personal computer needs? Well look no further, TSE-HARDSIDE, located in pleasant New Hampshire, has virtually every conceivable item for your micro. Whether you're shopping for your Apple, Pet, TRS-80 or Atari, TSE-HARDSIDE has it all. We stock hardware, software, books, magazines and specialty items for all of the popular machines. So the next time you're out shopping for your system don't be surprised, be satisfied. Remember the name TSE-HARDSIDE as your choice for quality, service and reliability.

TSE/HARDSIDE
6 South St., Milford, NH 03055 (603) 673-5144
TOLL FREE OUT-OF-STATE 1-800-258-1790



save $4.00

If you or a friend would like to subscribe to our TSE/HARDSIDE Complete Computer Catalog, simply send $2.00 and receive a $4.00 certificate good toward your first purchase of software, hardware, books or specialty items from TSE/HARDSIDE. I have a (check one) ☐ TRS-80, ☐ APPLE ☐ ATARI ☐ PET. A photo-copy of this coupon will be accepted.

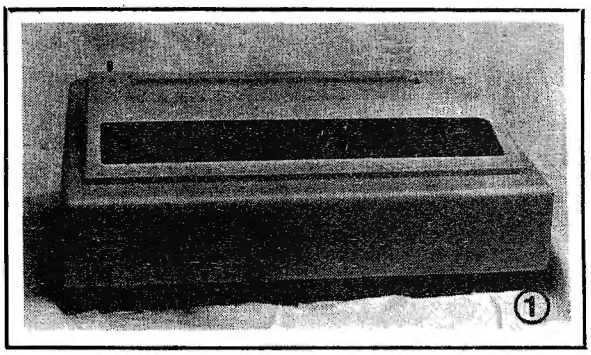
Name _____

Address _____

City _____ State _____ Zip _____ CAPR81

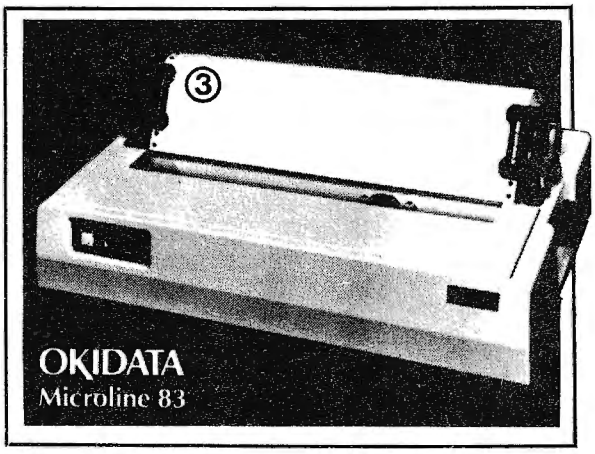
MAIL TO: TSE/HARDSIDE
6 South Street
Milford, NH 03055

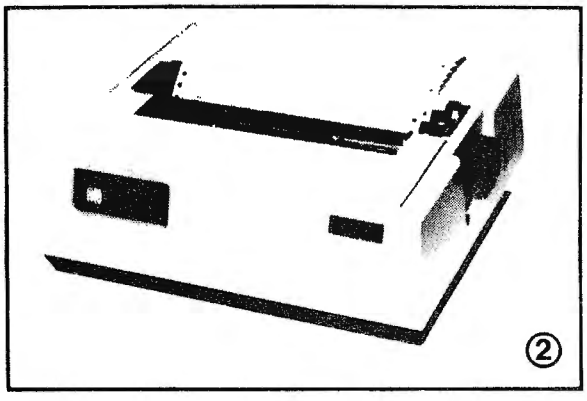
Circle 39



PICK YOUR PRINTER FROM TSE•HARDSIDE TODAY!

	RS Plotter/Printer	(26-1190) $1339.00
	RS Quick Printer II	(26-1155) $209.00
	TYPRINTER 221	(9-221) $2595.00
	VISTA 25-cps Daisy Wheel	(9-4000) $1795.00
	NEC 5510 SpinWriter	(9-5520) $3095.00
	NEC 5530 SpinWriter	(9-5530) $2595.00
	EATON-LRC 7000 +	(9-7000-64) $279.00
①	CENTRONICS 730	(9-730) $659.00
	CENTRONICS 737	(9-737) $769.00
	OKIDATA Microline-80	(9-80) $499.00
②	OKIDATA Microline-82	(9-82) $669.00
③	OKIDATA Microline-83	(9-83) $999.00
	OKIDATA SL300 (300 1pm)	(9-SL300) $3795.00
	EPSON MX-70	(9-MX70) $449.00
	EPSON MX-80	(9-MX80) $519.00





RS MOD-I Printer (36-pin) Int. Cable	(26-1411) $59.00
RS MOD-I Printer (40-pin) Int. Cable	(26-1416) $59.00
RS MOD-I & III Printer (36-pin) Cable	(26-1401) $29.00
RS MOD-I & III Printer (40-pin) Cable	(26-1415) $29.00
RS MOD-I & III LRC Printer Cable	(9-10) $29.00
APPLE Parallel Int. & Cable (36-pin)	(47-936) $100.00
APPLE Parallel Int. & Cable (40-pin)	(47-940) $100.00
APPLE LRC Interface & Cable	(47-LRC) $100.00
APPLE Asynchronous RS-232C Int	(47-7710A) $159.00
ATARI-Macrotronics Print. (36-pin) Int	(36-936) $69.95
ATARI-Macrotronics Print. (40-pin) Int	(36-940) $69.95

NEC Tractor-Feed Option	(9-5000) $229.00
BDT Sheet-Feeder (NEC only) Option	(9-5005) $1849.00
Microline-80 or 82 Tractor-feed Option	(9-80-T) $129.95
Centronics Zip-Pack Ribbons (3-pack)	(21-01) $15.95
SpinWriter Multi-Strike Ribbons (3)	(21-02) $18.95
Microline Printer Ribbon (3-pack)	(21-04) $12.95
LRC Printer Ribbons (3-pack)	(21-03) $6.95
EPSON Printer Ribbons (2-pack)	(21-05) $25.95

MasterCard VISA

TSE•HARDSIDE

6 South St., Milford, NH 03055 (603) 673-5144

TOLL FREE OUT-OF-STATE 1-800-258-1790

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 48 states) on all PREPAID orders over $100.00. On all orders under $100.00 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, Air Freight available at extra cost

Circle 40

Enter our world of microcomputing...

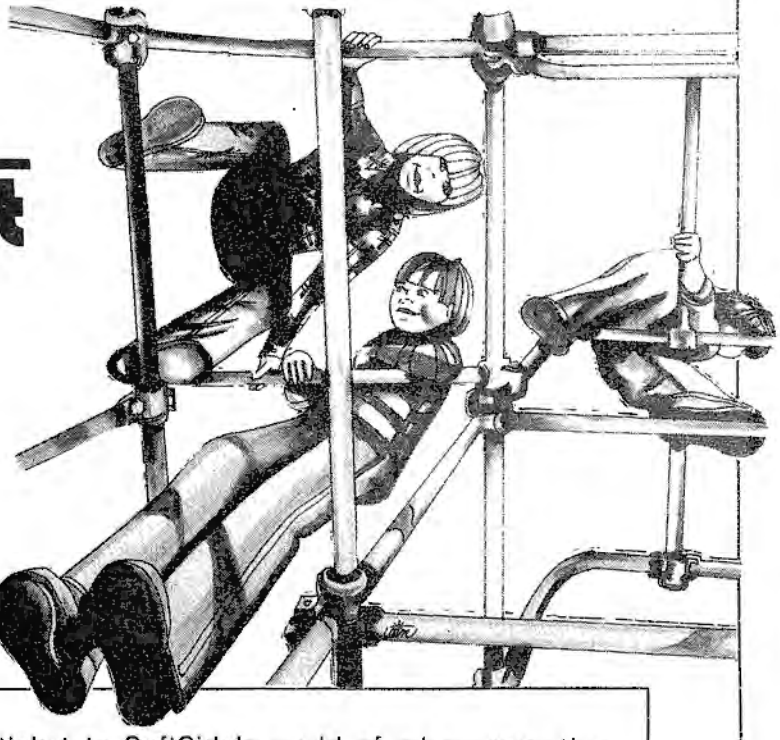
Discover our exciting world of microcomputing by reading SoftSide Magazine, the monthly that helps you learn to use your S-80 based system more effectively and enjoyably. In SoftSide, you'll read helpful programming hints, follow regular columns by such well-known authors as Scott Adams, and explore the wide range of S-80 compatible software and hardware.

The focus of SoftSide is lots of entertaining software — great games, simulations, and educational programs all ready to type right into your computer, each one fully documented and carefully explained. Every month you'll find at least three programs for the S-80. The value of these programs is impressive — one reader told us that he estimates the value of the S-80 software in SoftSide at $18 per issue!

SoftSide isn't for everyone. We're looking for the kind of reader who feels challenged by tackling all kinds of programs, long or short, and is rewarded with consistently fine graphics, great adventures and simulations, and arcade games that don't lose their appeal after one play.

If you're the kind of person who ought to be reading SoftSide, take advantage of our special offer while it lasts — 13 issues for the price of 12 (you must order with the coupon in this ad or a photocopy). For a small admission price, you can become a part of SoftSide's world of microcomputing!

SoftSide™ our viewpoint is strictly BASIC.



Here's my ticket to SoftSide's world of microcomputing.
Please send me 13 issues for only $18
(at a savings of 40% over cover price).

☐ Check enclosed ☐ Bill Me ☐ Bill my Visa ☐ MasterCard

Card # _____ Expiration date _____

I own a(n) _____

Name _____

Address _____

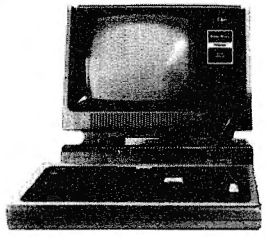
City _____ State _____ Zip _____

SoftSide Publications 6 South Street, Milford, New Hampshire M181

Circle 41

SYSTEM/COMMAND

Omni Load: Program Relocation Revisited



The 15th in a series.

Phil Pilgrim
Discovery Bay Software Co.
Port Townsend, Washington

A while back (Nov/Dec 1979 *80.U.S.-Journal*) I did a column on machine language program relocation. Here is another column on the same subject.

"Why make such a big deal out of it?", you ask. Well, as you know, the Radio Shack assembler is an absolute assembler. You have to decide at assembly time where in memory you want your program to reside. That means that if two programs are targeted to the same area, one will have to be reassembled not to conflict with the other. Once this is accomplished, you then have two (or more) non-conflicting and separate modules to load, one at a time. This can be tedious, time after time. Of course you *could* assemble them all together into one object file. But then *everything* has to be reassembled just to modify one routine.

Is all this juggling really necessary? Not at all. My original column on the subject showed a way to relocate a program in memory after loading it. This column carries the process a step further. The technique presented here allows you to merge already assembled object code into *one* program. This program will then load under Basic from tape or disk. Runing it will relocate each module therein and activate its startup code (for linking the keyboard, etc.). The program also contains the means for loading additional Editor/Assembler SYSTEM tapes and merging them with those extant. This loader converts the modules into Basic REMARK statements which may later be deleted like any other Basic statement.

The program which does all this (Omni-Load) is listed in Figure 1. Enter it under Level II or Disk Basic *exactly* as shown. Make no omissions or changes, and double check everything. Admittedly, this will be an awesome typing task. But getting this core of routines into the form shown in Figure 1 required an even tougher process bordering on trial and error. So concentrate on accuracy and use an expendable disk for the first few runs.

Once you have the beast typed in, *and SAVED or CSAVED*, run it. If it bombs completely, recheck the Basic code and the values in the DATA statements. If the message FD (or Bad File Data) IN nnnn appears, check the hex values in the remark statements. If the screen clears and the READY prompt appears at the top, you are ready for phase two.

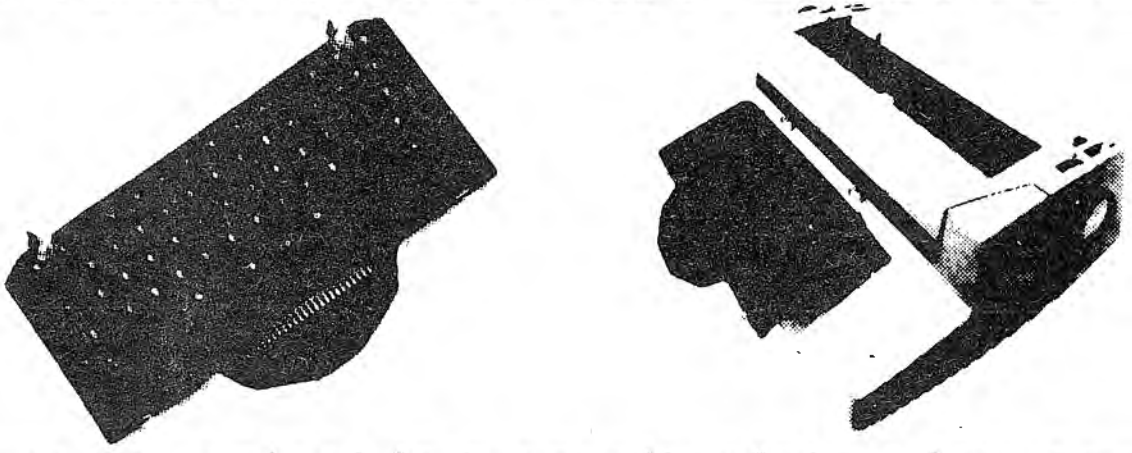
Look at the program in Figure 2. This is a routine for adding, auto repeat to the keyboard (i.e., hold a key down, and after a short delay it repeats about eight times per second). You will be loading this program into Omni-Load and saving it. From then on, whenever Omni-Load is run, you will have an auto-repeating keyboard. But first, let's look at the program.

There are several key elements essential to any module written for Omni-Load. Those elements are marked with asterisks (*) in Figure 2. The first is the ORG0. Following this is a two byte word equal to the Basic line number where the module should begin. Omni-Load will convert the program to a series of remark statements

Figure 1

```
10 DIMP%(141):GOSUB60000
20 P%(0)=1000:PRINT
30 '(C) COPYRIGHT 1980  DISCOVERY BAY SOFTWARE CO.
100 DATA0,0,11005,16414,26365,-513,-402,7714,10816,1655
  1,-32767,2304,11001,16561,8739,16614,22269,-515,-930,
  11469,24603,361,245,2557,-6659,8995,12926,16546,32291
  ,-23758,9024,-386
101 DATA-15725,1740,14,-7389,3277,18176,3277,22272,3277
  ,24320,3277,-18688,18464,10973,16614,6621,8368,-8942,
  -7195,-6622,-4800,-677,-8384,12513,7694,6156,-13030,1
  2,30685,-8960,4131,-12810
102 DATA12,-18567,1824,-16669,8227,6396,7845,-10966,-63
  67,4358,16405,6145,-4864,-12880,457,-15407,6562,3277,
  -7424,-18567,-7392,23533,16614,10973,16614,485,5,2525
  ,32477,-18688,10311,-8940
103 DATA-8925,110,9181,26333,6400,-31874,9079,-30082,42
  15,-8724,-6614,-8896,110,26333,31745,10421,6404,3277,
  -8960,-6614,-8896,606,22237,-8957,-8935,-6622,-7872,6
  319,-626,-7199,-10626,-464
104 DATA14346,-10750,8967,1799,1799,-10618,-13520,10358
  ,-10750,8967,-2589,20353,-527,20457
1000 REM00FFDB0026
1001 REM250000000900160000011000002A1640221B40211600221
  640C9CDE303FE1AC03A0838E604C83E1AC95E
1002 REM00000001FF
50000 REM00FFB00051
50001 REM40000000013002C000006170001D0026002D0046004B0000
  2A16402238002A1B40223300214B49221B40212C00221640C93A4
  F00B72005CD00001803CD0000FE01C03A803824
50002 REM10004000A720023CC93A4F00EE01324F00AFC90170
50003 REM00000001FF
56000 REM00FFBB0046
```

At last --- the DYNATYPER TYPEWRITER INTERFACE!



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

DYNATYPER -- Rochester Data's patented* Computer/Typewriter Interface is the industry standard for typewriter output.

- 2 minutes to initially install and 5 seconds to remove or replace.
- You do not have to modify your typewriter. All factory warranties and maintenance agreements on your typewriter will be honored.
- Compatible with all power carriage return typewriters having standard U.S. keyboard. The Dynatyper works with Selectrics (model 1) and most non-Selectrics (model 2). Please specify. Typewriter conversion between models takes 2 minutes and the kit (17 plungers) is available for a nominal fee.
- The Dynatyper is compatible with all major word processing software. (Scripsit, Pencil, Applewriter, Easywriter, Magic Window, Visi-calc CCADBM, Supertext, Write On)
- Interfaces available for TRS-80, APPLE, PET/CBM, OSI, Northstar, HP-85, H-89. Weighs only 3 lbs. Extremely portable.
- Delivery: Stock to two weeks. Price $499 for complete system. F.O.B. Rochester, Domestic. VISA and Master Card accepted. Call Ken Yanicky at 716-244-7804

*Pat. Pending

ROCHESTER DATA

incorporated

3000 Winton Road South, Rochester, N.Y. 14623

beginning at that line number and incrementing by one. The choice of a line number is most important, because modules are activated in the order in which they appear in Omni-Load. Therefore, if several modules link into the keyboard calling sequence, those first in line will be "closest to the keyboard", i.e., called last. No module should have a line number greater than 49999.

Next is a sixteen-bit value equal to *minus* the total number of bytes in the program. This tells Omni-Load how much space to reserve at the top of memory when it begins converting that module's remark statements into machine code. (An Omni-Load module is limited to a 2000 byte length). After another ORG0 come two addresses. The first points to the startup code, executed *once* when Omni-Load is run, and destroyed. The second points to the resident code, which OmniLoad protects in upper memory.

Next comes a DEFB 0, followed by a relocation table. The relocation table needs some explanation. All those instructions (e.g., JP and CALL) and data in the startup and resident routines referring absolutely (with a two byte address) to other addresses *within those routines* need to be singled out so Omni-Load can add a displacement value to them. This is the relocation process. The table is just a list of pointers to those instructions and data. If, in writing an assembly program, such an instruction ends up without a label, you will need to add one so you can refer to it in the table. Most (but not all) absolute address references begin in an instruction's second byte. Hence all the +1's. To get the correct displacement for any instruction in question, consult your Editor/ Assembler manual. The relocation table must begin with a byte equal to its length in two-byte words and must end with a DEFB0.

Following the relocation table is the startup block. This handles the preliminary linking and initialization tasks. Most programs in SYSTEM/ COMMAND have had such a block, ending with JP 1A19H or 402DH. Omni-Load startup blocks must end with RET.

Finally comes the resident code. Omni-Load protects this code in upper memory by readjusting Basic's MEMORY SIZE.

Now, type the auto-repeat example into the Editor/Assembler, assemble it and make an object tape. (Yes, disk users, get out the old CTR-80 and clean the heads - you are gonna make a *tape*!) Now, with SYSTEM tape in hand, load Omni-Load and type RUN 56789 *ENTER*. You should get the message

READY CASSETTE—

If not, proofread the REM's, after 57000, correct and try again. Once you get the right message, stick the SYSTEM cassette in the recorder, press PLAY and hit ENTER. The tape will begin to read, displaying its name, and signalling you with asterisks. When loaded, the REM's will be generated

```
56001 REM40000000007004500000000111E00CD2C1BD20000AF6069
  2323063862310FCFEA4C20000ED5BE6401B2AB14023232323010
  400EDB8EB22B14022494011CEFF1922A040CD06
56002 REM050040004A1BC3CC06C1
56003 REM00000001FF
56789 DIMP%(141):GOSUB60000
56790 P%(0)=-8536:PRINT
56800  STOP
57000 REM00F4E2002A
57001 REM40000000A300B000004EA700AA00B200B500BC00C000C3
  00CA00E000FC0005011A011D01200123013601 3A013E0143014A0
  150015801 5E016B017201790181019101940 1B3
57002 REM40004000970 19A019F01A501AB01AF01B501BB01C001C4
  01C801CC01D301D901DF01E301E601EB01F901FF01020206020A0
  2140218021B021F0225022B02310236023D025A
57003 REM40008000400243024702 4D02530258025E0262028A028F
  029902A102A502AD02BA02C002C502002A164022D70221B000221
  640C9ED7312033AD902B7280B3DCAC7013DCAE1
57004 REM4000C000DE01C3C701AF329C4021DA02CDA728CDE303B7
  28FAFE01200F3A8038B728F02AD702221640C3CC06FE0D20E3CD3
  3003E0FCD3300AFCD1202CD9602CD6201FE55E2
57005 REM400100002 0F60606CD6201CD330010F83E0DCD33003E2A
  323E3C323F3C111D0B211D03221B03CD5401FE782853FE3C20223
  A3F3CEE0A323F3CCD540147CD54014FCD5401BA
57006 REM40014000814FCD5401814F10F9CD6201B928D321F20218
  19DF3006CD62017723C9210303180B3A4038E604CA350221EB02C
  DA728ED7B1203CDF801AF32D902C9CDF801FDC6
57007 REM40018000211D03FD7E01FE0420C5FD6E04FD6605221403
  CD5C0221250322 1B03AF4FCDA902FD7E07CDA902FD7E06CDA902A
  FCDA90279ED44CDA9022AA7402217033E013267
57008 REM4001C000D902AF321603C92A1703232217032B7EB7C03A
  D9023CE60332D9023E0DC93A1603B7CC8E02CD5C0238503A16034
  7B72804FE4038020640903216 03AF4F78CDA928
57009 REM40020000022A19037CCDA9027DCDA9027D806F7CCE0067
  221903AFCDA9022A1B037ECDA9022310F9221B0379ED44CDA9022
  AA7402217033E0132D902AFC9AF4FCDA902CDEB
57010 REM40024000A902CDA9023CCDA90279ED44CDA9022AA74022
  17033E0332D902AFC9082A1403232214032B11FF7FAFED52CDD20
  BCDBD0F23ED5BA7407EB7280512231318F7EBEC
57011 REM400280003652233645233 64D2322170308C92A1B037EFE
  7837C8237E321603235E2356ED53190323221B03AFC9E5D5F52A1
  7035F814F7B0F0F0F0FE60FCDCB027BE60FCD45
57012 REM4002C000CB023600221703F1D1E1C9FE0A3802C607C630
  7723C9C30000000D52454144592043415353455454450E0042524
  5414B0D00544150452052454144204552524FEC
57013 REM1203000052 0D00544150452054 4F4F204C4F4E470D00F3
57014 REM00000001FF
60000 V0=0:V1=0:RESTORE:FORI=0TO141:READP%(I):NEXT
60010 A=PEEK(16414)+256*PEEK(16415):IFA>32767THENA=A-65
  536
60020 P%(1)=A:V=VARPTR(P%(2)):IFV<0THENV=V+65536
60030 V1=INT(V/256):V0=V-V1*256:POKE16414,V0:POKE16415,
  V1
60040 RETURN
```

Figure 2

```
00100          ORG     Ø                ;*ALWAYS ORG Ø
00110          DEFW    3ØØØ             ;*BASIC LINE NO.
00120          DEFW    -KBDE            ;*-TOTAL NO. BYTES
00130          ORG     Ø                ;*ALWAYS ORG Ø
00140          DEFW    START            ;*POINT TO STARTUP CODE
00150          DEFW    AKBD             ;*POINT TO RESIDENT CODE
00160          DEFB    Ø                ;*ALWAYS DEFB Ø
00170          DEFB    RELOCE-RELOC<-1  ;*NO. 2-BYTE WDS IN TABLE
00180 RELOC    DEFW    AKBD+1           ;RELOCATION TABLE
00190          DEFW    ARØ1+1
00200          DEFW    ARØ2+1
00210          DEFW    ARØ3+1
00220          DEFW    ARØ4+1
00230          DEFW    ARØ5+1
00240          DEFW    ARØ6+1
00250          DEFW    ARØ7+1
00260          DEFW    ARØ8+1
00270          DEFW    ARØ9+1
00280          DEFW    AR1Ø+1
00290 RELOCE   DEFB    Ø                ;*ALWAYS DEFB Ø
00300 ;
00310 ;   STARTUP CODE: MUST PRECEDE RESIDENT CODE.
00320 ;
00330 START    LD      HL,(4Ø16H)       ;STARTUP CODE; LINK INTO
00340 ARØ9     LD      (SKBD+1),HL      ;KEYBOARD CALLING SEQUENCE
00350 AR1Ø     LD      HL,AKBD          ;       .
00360          LD      (4Ø16H),HL       ;       .
00370          RET                      ;ALWAYS END W/ RET
00380 ;
00390 ;   RESIDENT CODE
00400 ;
00410 AKBD     LD      A,(CSAVE)        ;CHAR BEING SAVED?
00420          OR      A                ;       .
00430          JR      Z,KBDØ           ;  NO: DON'T DELAY
00440 ARØ1     LD      A,(COUNT)        ;255 TIMES THRU?
00450          CP      255              ;       .
00460          JR      C,KBDØ           ;  NOT YET: FORGET DELAY
00470          LD      BC,5ØØØ          ;DELAY: ADJUST THE 5ØØØ TO SUIT
00480          CALL    ØØ6ØH            ;ROM DELAY ROUTINE
00490 KBDØ     LD      HL,4Ø36H         ;COPY LV II KBD SAVE VECTOR
00500 ARØ2     LD      DE,VSAVE         ;       .
00510          LD      BC,8             ;       .
00520          LDIR                     ;       .
00530 SKBD     CALL    $-$              ;CALL OLD KBD ROUTINE
00540          PUSH    AF               ;RESERVE CHARACTER
00550          LD      HL,4Ø36H         ;COMPARE NEW VECTOR TO OLD
00560 ARØ8     LD      DE,VSAVE         ;       .
00570          LD      B,8              ;       .
00580 ACLOOP   LD      A,(DE)           ;       .
00590          CP      (HL)             ;DIFFERENT?
00600          JR      NZ,DIFFER        ;  YES
```

and echoed on the screen. Finally, READY CASSETTE— will again be displayed. You may either load another tape or hit *SHIFT-BREAK* to return to Basic. At this point reSAVE Omni-Load, which now contains the auto repeat module.

Not all tape loads will be successful. You may get a TAPE LOAD ERROR, which means your SYSTEM tape was not of the prescribed format. You may also get a TAPE TOO LONG error, which means your module exceeds 2000 bytes. Pare it down or break it up.

Once merged successfully, you may use the auto repeat by running OmniLoad. Now whenever you hold down a key it will repeat itself. To temporarily disable this feature and use a previously loaded (NEWDOS for example) keyboard routine, type *SHIFT-BREAK*. To re-enable hit *SHIFT-BREAK* again.

Omni-Load should handle most small machine language utilities you want to write. Programs from previous SYSTEM/COMMAND columns are ripe candidates for conversion to the Omni-Load format. Not all combinations of these routines will be successful though, because of timing interactions and sharing of certain function keys. But experiment anyway. You may come up with the perfect utility for your needs.

```
00610           INC   HL
00620           INC   DE
00630           DJNZ  ACLOOP
00640 AR03      LD    A,(COUNT) ;ENOUGH DELAY?
00650 DLYL      CP    255       ;       .
00660           JR    C,COUNTI  ;  NO: COUNT SOME MORE
00670           POP   HL        ;  YES: FORGET RESERVED CHAR
00680 AR04      LD    A,(CSAVE) ;GET SAVED CHAR
00690           RET             ;AND RETURN IT
00700 COUNTI    INC   A         ;INC (COUNT)
00710 AR05      LD    (COUNT),A ;       .
00720           XOR   A         ;& RETURN A 0
00730           POP   HL        ;(DISCARD RESERVED CHAR)
00740           RET
00750 DIFFER    XOR   A         ;ZERO COUNT
00760 AR06      LD    (COUNT),A ;       .
00770           POP   AF        ;SAVE & RETURN RESERVED CHAR
00780 AR07      LD    (CSAVE),A ;       .
00790           RET             ;       .
00800 CSAVE     DEFB  0         ;CHAR SAVE AREA
00810 COUNT     DEFB  0         ;PASS COUNTER
00820 VSAVE     DEFS  8         ;KBD VECTOR SAVE AREA
00830 KBDE      EQU   $         ;=NO. BYTES IN PROGRAM
00840           END
```

A Basic Self-Programmer and Interpreter

Wallace Havenhill
Cleveland Heights, Ohio

As a teacher of mathematics, I have been interested in programs that graph mathematical functions. One such program was printed in this magazine (*Function Grapher/Root Finder*, Nov/Dec 1979, page 20). However, it, like most function graphing programs, required the user to enter equations by retyping the program line holding the mathematical function and then running the entire program.

I feel that the computer should do as much as possible for the user, so I set about writing a program that would take a string entered with an INPUT statement and write that string into a program line. In short, to make the computer partially self-programming. To do this, I had to write a routine which is a partial Basic interpreter in order to change the ASCII character codes of mathematical operatiors and functions to their ASCII function codes. The Sep/Oct 1979 issue of this magazine contains a list of the ASCII function codes.

Run this program, entering different equations. Try X=Y; T=SIN(X); and Y=SQR(3*X+4). After the program displays the value for Y, break the program and list line 200. The equations entered via the INPUT statement will be in this line.

Program Analysis

Subroutine 1000

Line 1000 locates the start of the program in RAM by peeking the pointer at 16548 and 16549. Normally this location points to 17129, but disk Basic pushes this location upwards in memory. Line 1010 holds the line number where the equation is going to be placed.

Line 1020 converts LN to MS (most significant byte) and LS (least significant byte).

Lines 1030-1050 search for the memory location of line LN in RAM and places the memory location of the start of that line's content into ML.

Subroutine 2000

This is the partial Basic interpreter that converts ASCII character codes to their ASCII function codes.

Lines 2040-2090 convert +, -, *, /, = and the up arrow, respectively.

Lines 2100-2140 convert COS, TAN, LOG, INT, and EXP. Note that this routine only checks the first letter. A "C" implies COS, "T" implies TAN, etc.

Lines 2150-2160 convert SIN, SQR, ATN, and ABS. Here, the routine must check the second letter as well as the first to get the correct function.

Line 2200 fills the remaining RAM locations for line LN with blanks.

Things to Watch

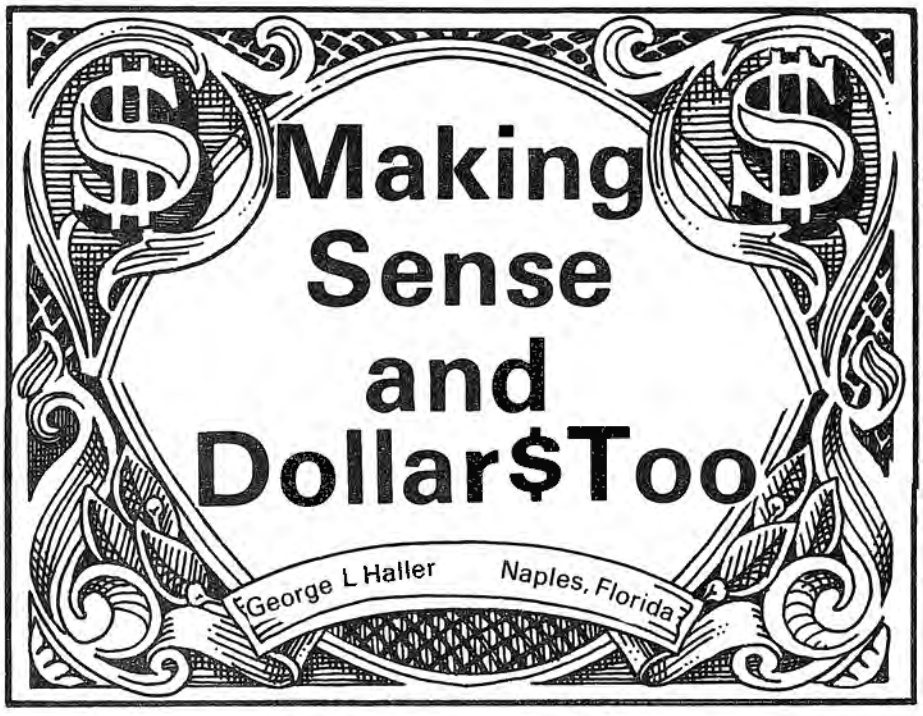
In using this technique, the programmer must observe the following rules:

1. Line number searched for *must* be pre-defined.
2. Line number must not reside in a portion of memory that lies in the expansion memory (beyond the 16K boundary).

In creating the formulas, beware of COS(CT) types. (Try it to see the results).

```
100 CLS
110 GOSUB 1000
120 GOSUB 2000
130 INPUT "VALUE FOR X"; X
200 REM PAD THIS LINE WITH ENOUGH CHARACTERS TO
  EXCEED YOUR LONGEST EQUATION
300 PRINT E$
310 PRINT X, Y
320 PRINT : GOTO 120
1000 MM = PEEK(16548) + 256 * PEEK(16549)
1010 LN=200
1020 MS = INT(LN / 256) : LS = LN - 256 * MS
1030 M = MM
1040 IF PEEK(M+2)<>LS OR PEEK(M+3)<>MS THEN
  M=PEEK(M) + 256 * PEEK(M+1) : GOTO 1040
1050 ML = M+4
1060 RETURN
2000 INPUT "ENTER YOUR EQUATION, Y=F(X)"; E$
2010 M = ML
2020 FOR N = 1 TO LEN(E$)
2030 Q = ASC(MID$(E$,N,1))
2040 IF Q=43 THEN Q=205 : GOTO 2170
2050 IF Q=45 THEN Q=206 : GOTO 2170
2060 IF Q=42 THEN Q=207 : GOTO 2170
2070 IF Q=47 THEN Q=208 : GOTO 2170
2080 IF Q=61 THEN Q=213 : GOTO 2170
2090 IF Q=91 THEN Q=209 : GOTO 2170
2100 IF Q=67 THEN Q=225 : N = N + 2 : GOTO 2170
2110 IF Q=84 THEN Q=227 : N = N + 2 : GOTO 2170
2120 IF Q=76 THEN Q=223 : N = N + 2 : GOTO 2170
2130 IF Q=73 THEN Q=216 : N = N + 2 : GOTO 2170
2140 IF Q=69 THEN Q=224 : N = N + 2 : GOTO 2170
2150 IF Q=83 THEN N = N + 2 : IF MID$(E$,N-1,1)
  = "I" THEN Q = 226 ELSE Q = 221
2160 IF Q=65 THEN N = N + 2 : IF MID$(E$,N-1,1)
  = "T" THEN Q = 228 ELSE Q = 217
2170 POKE M, Q
2180 M = M + 1
2190 NEXT N
2200 IF PEEK(M) <> 0 THEN POKE M,32 : M = M + 1
  : GOTO 2200
2210 RETURN
```

●



In writing checks and other financial papers it is customary to put the amount in the Arabic numerals and also in English words. There are probably some good reasons for this. However, a discussion of them is beyond the scope of this article. We **will** show you a Basic language program to convert the usual Arabic number to English words.

This program is a demonstration which accepts continuing numeric inputs in line 90 (variable A) and converts them into English word equivalents. By some simple changes, the program can be used as a subroutine for use in any program which requires this type of output. Any program which writes checks (such as payroll or accounts payable) could use this routine.

The program will accept any number from 1.00 to 99,999.99. For amounts higher than this, the program will have to be modified. Other variations are possible, too. For instance, if the printed line is too long for your checks, it could be shortened by substituting the "&" character for "AND". The "xx/100" could be replaced with "CENTS", requiring the relocation of the word, "DOLLARS".

The examples shown here also include variable values as a debugging aid.

```
10 REM       THIS PROGRAM WILL CONVERT
20 REM       ARABIC NUMBER SYMBOLS TO
30 REM       ENGLISH NUMBER WORDS
40 REM       BY G. L. HALLER
50 DIM T$(27)
60 DEFDBL A
70 V$="##"
80 LPRINT
90 INPUT "NUMBER = ";A
100 IF A>99999.99 THEN
    PRINT"NUMBER IS TOO BIG" :
    GOTO 90
110 LPRINT "INPUT NUMBER A IS ";A
120 LPRINT
130 T=INT(A)
140 B=100*(A-T)
150 T1=INT(T/1000)
160 T2=INT(T1/10)
170 T3=T1-T2*10
180 T4=T-1000*T1
190 T5=INT(T4/100)
200 T6=T4-100*T5
210 T7=INT(T6/10)
220 T8=T6-10*T7
230 RESTORE
240 DATA ONE,TWO,THREE,FOUR,FIVE,SIX
250 DATA SEVEN,EIGHT,NINE,TEN,ELEVEN
260 DATA TWELVE,THIRTEEN,FOURTEEN
270 DATA FIFTEEN,SIXTEEN,SEVENTEEN
280 DATA EIGHTEEN,NINETEEN,TWENTY
290 DATA THIRTY,FORTY,FIFTY,SIXTY
300 DATA SEVENTY,EIGHTY,NINETY
310 FOR I=1 TO 27
320 READ T$(I)
330 NEXT
340 T$(0)=""
350 IF T1=>20 THEN 410
360 IF T<100 THEN 460
370 IF T<1000 THEN 440
```

```
380 LPRINT T$(T2*10+T3);
390 GOSUB 580
400 GOTO 430
410 LPRINT T$(T2+18);" ";T$(T3);
420 GOSUB 580
430 IF T5=0 THEN 460
440 LPRINT T$(T5);
450 GOSUB 620
460 IF T6=>20 THEN 530
470 IF T6=0 THEN PRINT
480 LPRINT T$(T6);" AND ";
490 IF B=0 THEN LPRINT "NO/100 DOLLARS":
    GOTO 90
500 LPRINT USING V$;B;
510 LPRINT "/100 DOLLARS"
520 GOTO 90
530 LPRINT T$(T7+18);" ";T$(T8);" AND ";
540 IF B=0 THEN LPRINT "NO/100 DOLLARS":
    GOTO 90
550 LPRINT USING V$;B;
560 LPRINT "/100 DOLLARS"
570 GOTO90
580 LPRINT " THOUSAND";
590 IF T4=0 AND T6=0 THEN RETURN
600 LPRINT " AND ";
610 RETURN
620 LPRINT " HUNDRED";
630 IF T6=0 THEN RETURN
640 LPRINT " AND ";
650 RETURN
```

For use with the Model I, II, and III

```
INPUT NUMBER A IS  12345.67

T= 12345          B= 67
T1= 12            T2= 1             T3= 2             T4= 345
T5= 3             T6= 45            T7= 4             T8= 5
TWELVE THOUSAND AND THREE HUNDRED AND FORTY FIVE AND 67/100 DOLLARS

INPUT NUMBER A IS  89999.99

T= 89999          B= 99
T1= 89            T2= 8             T3= 9             T4= 999
T5= 9             T6= 99            T7= 9             T8= 9
EIGHTY NINE THOUSAND AND NINE HUNDRED AND NINETY NINE AND 99/100 DOLLARS

INPUT NUMBER A IS  50

T= 50             B= 0
T1= 0             T2= 0             T3= 0             T4= 50
T5= 0             T6= 50            T7= 5             T8= 0
FIFTY  AND NO/100 DOLLARS
INPUT NUMBER A IS  100

T= 100            B= 0
T1= 0             T2= 0             T3= 0             T4= 100
T5= 1             T6= 0             T7= 0             T8= 0
ONE HUNDRED AND NO/100 DOLLARS
INPUT NUMBER A IS  215.58

T= 215            B= 58
T1= 0             T2= 0             T3= 0             T4= 215
T5= 2             T6= 15            T7= 1             T8= 5
TWO HUNDRED AND FIFTEEN AND 58/100 DOLLARS
INPUT NUMBER A IS  219

T= 219            B= 0
T1= 0             T2= 0             T3= 0             T4= 219
T5= 2             T6= 19            T7= 1             T8= 9
TWO HUNDRED AND NINETEEN AND NO/100 DOLLARS
```

(Line Packing...Cont'd from page 26)

Line #6: Data For Zero Move

STEP#	DECIMAL	HEX	STATEMENT
0	68	44	"D"
1	1	01	COUNTER
2	132	84	DATA
3	129	81	"
4	129	81	"
5	1	01	"
6	7	07	"
7	129	81	"
8	129	81	"
9	4	04	"
10	7	07	"
11	65	41	"
12	129	81	"
13	7	07	"
14	1	01	"
15	7	07	"
16	129	81	"
17	135	87	"
18	4	04	"
19	71	47	"
20	1	01	"
21	65	41	"
22	65	41	"
23	1	01	"
24	132	84	"
25	1	01	"
26	1	01	"
27	7	07	"
28	71	47	"
29	135	87	"
30	129	81	"
31	135	87	"
32	135	87	"
33	7	07	"
34	132	84	"
35	71	47	"
36	129	81	"
37	135	87	"
38	135	87	"
39	135	87	"
40	1	01	"
41	71	47	"
42	1	01	"
43	129	81	"
44	4	04	"
45	132	84	"
46	4	04	"
47	7	07	"
48	7	07	"
49	7	07	"
50	135	87	"
51	135	87	"
52	1	01	"
53	65	41	"
54	135	87	"
55	1	01	"
56	7	07	"
57	129	81	"

Line #2: Line Finder

STEP#	DECIMAL	HEX	STATEMENT
0	121	79	LDA,C
1	254	FE	CP,N
2	13	0D	N
3	200	C8	RET,Z
4	42	2A	LDHL,(NN)
5	164	A4	N
6	64	40	N
7	84	54	LDD,H
8	93	5D	LDE,L
9	35	23	INC,HL
10	35	23	INC,HL
11	190	BE	CP,(HL)
12	40	28	JRZ
13	7	07	e
14	235	EB	EXDE,HL
15	94	5E	LDD,(HL)
16	35	23	INC,HL
17	86	56	LDE,(HL)
18	235	EB	EXDE,HL
19	24	18	JR
20	242	F2	e
21	35	23	INC,HL
22	35	23	INC,HL
23	126	7E	LDA,(HL)
24	35	23	INC,HL
25	254	FE	CP,N
26	68	44	N
27	32	20	JRNZ
28	4	04	e
29	34	22	LD(NN),HL
30	40	28	N
31	64	40	N
32	201	C9	RET
33	254	FE	CP,N
34	83	53	N
35	192	C0	RET,NZ
36	233	E9	JP,(HL)

Line #5: Screen Painter

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	33	21	LDHL,NN
2	1	01	N
3	60	3C	N
4	84	54	LDD,H
5	93	5D	LDE,L
6	43	2B	DEC,HL
7	112	70	LD(HL),B
8	1	01	LDBC,NN
9	255	FF	N
10	3	03	N
11	237	ED	LDIR
12	176	B0	-
13	201	C9	RET

Line #32: CLS

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	1	01	LDBC,NN
2	5	05	N
3	128	80	N
4	205	CD	CALL
5	37	25	N
6	64	40	N
7	201	C9	RET

Line #33: All On

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	1	01	LDBC,NN
2	5	05	N
3	191	BF	N
4	205	CD	CALL
5	37	25	N
6	64	40	N
7	201	C9	RET

Line #35: Screen Flash

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	6	06	LDB,N
2	32	20	N
3	62	3E	LDA,N
4	1	01	N
5	217	D9	EXX
6	8	08	EXAF,AF'
7	33	21	LDHL,NN
8	1	01	N
9	60	3C	N
10	84	54	LDD,H
11	93	5D	LDE,L
12	43	2B	DEC,HL
13	1	01	LDBC,NN
14	255	FF	N
15	3	03	N
16	237	ED	LDA,R
17	95	5F	-
18	230	E6	AND,N
19	63	3F	N
20	40	28	JRZ
21	250	FA	e
22	203	CB	SET 7,A
23	255	FF	-
24	119	77	LD(HL),A
25	237	ED	LDIR
26	176	B0	-
27	8	08	EXAF,AF'
28	238	EE	XOR,N
29	3	03	N
30	211	D3	OUT(N),A
31	255	FF	N

(Listings continued on page 96)

a subroutine Steps 1 - 3 load the HL register with the address of the start of the screen memory plus one. Steps 4 and 5 place this address in the DE registers also In step 6, HL is decremented by one so as to point to the first byte of the screen memory

Step 7 places whatever is stored in register B into the first byte of the screen memory Steps 8 - 10 set the BC register to be a counter which will count a total of one less than the number of bytes in the screen memory (this number in BC is 1023D or 3FFH) We have already loaded the first byte of the screen memory in step 7. Steps 11 and 12 copy the first byte of the screen (which contains whatever was in the B register originally) into each next byte of the screen memory using BC as a counter to know when the job is done

This means that if we want all the screen turned on we would load a 191D, or BFH, into register B and call line 5

To pack line 5 into our program, do the following: Delete lines 255-1003, then enter the following new lines and then save the program After the program is saved, run the program *once only!*

```
18) 5D              (13 PERIODS)
19) 255 LPRINTCHR$(5)
20) 1000 DATA83,33,1 60,84,93,43,112,1,255 3,237,176,201
21) 1010 L PEEK(16424)+PEEK(16425)*256-1
22) 1011 RESTORE FORN: OTO13 READD POKEL+N,D NEXTN
```

Line 5 should now be packed. Notice that we had to start the line in (18) with a "D" so that Line Finder would give us line 5' address. The minus one at the end of (21) causes the first item of data, an 83D, or an "S" for subroutine, to replace the "D" at the start of line 5 If you run the program again (try it if you like), line 5 is now packed as a subroutine, and the Line Finder will jump to the "paint screen" subroutine. This will cause the screen to be filled with whatever is in register B. This subroutine is not really meant to be called from Basic. It will normally be called from another Z-80 subroutine. In fact, the next two lines we will pack will be subroutines that call line 5.

Line 33 turns the screen on. Look at the listing in Figure 3 for line 33. All On. Step 0 identifies the line as a subroutine Steps 1 - 3 load the C register with the number of the line we wish to find, in this case, line 5. The B register is loaded with the byte we wish to have fill the screen, in this case a 191D (the graphics code which turns on a print position). Steps 4 - 6 call to the jump in the LPRINT vector at address 16421D, or 4025H. From there the program is sent to the Line Finder, where the line whose number is stored in the C register will be searched for and found. Since the line being found, line 5, is a subroutine, the program jumps from line 2 to line 5 where the content of the B register is loaded into the screen. Thus, to turn on the screen from Basic, we would LPRINT"!" or LPRINTCHR$(33).

Change or add the lines (at the beginning of the text on page 99) to our program and then save it again.

(Text continued on page 99)

Line #35: Screen Flash *(Cont'd)*

STEP#	DECIMAL	HEX	STATEMENT
32	217	D9	EXX
33	16	10	DJNZ
34	226	E2	e
35	1	01	LDBC,NN
36	5	05	N
37	128	80	N
38	205	CD	CALL
39	37	25	N
40	64	40	N
41	201	C9	RET

Line #36: Set Zero Direction

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	14	0E	LDC,N
2	6	06	N
3	205	CD	CALL
4	37	25	N
5	64	40	N
6	78	4E	LDC,(HL)
7	52	34	INC,(HL)
8	62	3E	LDA,N
9	57	39	N
10	190	BE	CP,(HL)
11	32	20	JRNZ
12	2	02	e
13	54	36	LD(HL),N
14	1	01	N
15	6	06	LDB,N
16	1	01	N
17	5	05	DEC,B
18	9	09	ADDHL,BC
19	70	46	LDB,(HL)
20	14	0E	LDC,N
21	101	65	N
22	205	CD	CALL
23	37	25	N
24	64	40	N
25	35	23	INC,HL
26	35	23	INC,HL
27	112	70	LD(HL),B
28	201	C9	RET

Line #37: Up Sweep

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	62	3E	LDA,N
2	1	01	N
3	14	0E	LDC,N
4	100	64	N
5	65	41	LDB,C
6	16	10	DJNZ
7	254	FE	e
8	238	EE	XOR,N
9	3	03	N
10	211	D3	OUT(N),A
11	255	FF	N
12	13	0D	DEC,C
13	32	20	JRNZ
14	246	F6	e
15	201	C9	RET

Line #50: Preset Plus and Zero

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	14	0E	LDC,N
2	100	64	N
3	205	CD	CALL
4	37	25	N
5	64	40	N
6	17	11	LDDE,NN
7	153	99	N
8	60	3C	N
9	115	73	LD(HL),E
10	35	23	INC,HL
11	114	72	LD(HL),D
12	14	0E	LDC,N
13	101	65	N
14	205	CD	CALL
15	37	25	N
16	64	40	N
17	17	11	LDDE,NN
18	159	9F	N
19	62	3E	N
20	115	73	LD(HL),E
21	35	23	INC,HL
22	114	72	LD(HL),D
23	14	0E	LDC,N
24	32	20	N
25	205	CD	CALL
26	37	25	N
27	64	40	N
28	14	0E	LDC,N
29	51	33	N
30	205	CD	CALL
31	37	25	N
32	64	40	N
33	14	0E	LDC,N
34	52	34	N
35	205	CD	CALL
36	37	25	N
37	64	40	N
38	201	C9	RET

Line #51: Set Plus On Screen

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	14	0E	LDC,N
2	100	64	N
3	205	CD	CALL
4	37	25	N
5	64	40	N
6	94	5E	LDE,(HL)
7	35	23	INC,HL
8	86	56	LDD,(HL)
9	98	62	LDH,D
10	107	6B	LDL,E
11	54	36	LD(HL),N
12	140	8C	N
13	35	23	INC,HL
14	54	36	LD(HL),N
15	191	BF	N
16	35	23	INC,HL
17	54	36	LD(HL),N
18	140	8C	N
19	62	3E	LDA,N
20	1	01	N
21	211	D3	OUT(N),A
22	255	FF	N
23	6	06	LDB,N
24	50	32	N
25	16	10	DJNZ
26	254	FE	e
27	62	3E	LDA,N
28	2	02	N
29	211	D3	OUT(N),A
30	255	FF	N
31	201	C9	RET

Line #52: Set Zero On Screen

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	14	0E	LDC,N
2	101	65	N
3	205	CD	CALL
4	37	25	N
5	64	40	N
6	94	5E	LDE,(HL)
7	35	23	INC,HL
8	86	56	LDD,(HL)
9	98	62	LDH,D
10	107	6B	LDL,E
11	54	36	LD(HL),N
12	191	BF	N
13	35	23	INC,HL
14	54	36	LD(HL),N
15	179	B3	N
16	35	23	INC,HL
17	54	36	LD(HL),N
18	191	BF	N
19	201	C9	RET

Line #53: Count Dots

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	62	3E	LDA,N
2	140	8C	N
3	1	01	LDBC,NN
4	255	FF	N
5	3	03	N
6	17	11	LDDE,NN

Line #53: Count Dots *(Cont'd)*

STEP#	DECIMAL	HEX	STATEMENT
7	42	2A	N
8	64	40	N
9	33	21	LDHL, NN
10	1	01	N
11	60	3C	N
12	237	ED	CPIR
13	177	B1	-
14	224	E0	RET, PO
15	32	20	JRNZ
16	251	FB	e
17	235	EB	EXHL, DE
18	52	34	INC, (HL)
19	235	EB	EXHL, DE
20	24	18	JR
21	246	F6	e

Line #55: Game Control

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	14	0E	LDC, N
2	70	46	N
3	205	CD	CALL
4	37	25	N
5	64	40	N
6	14	0E	LDC, N
7	60	3C	N
8	205	CD	CALL
9	37	25	N
10	64	40	N
11	14	0E	LDC, N
12	100	64	N
13	205	CD	CALL
14	37	25	N
15	64	40	N
16	94	5E	LDE, (HL)
17	35	23	INC, HL
18	86	56	LDD, (HL)
19	213	D5	PUSH, DE
20	14	0E	LDC, N
21	101	65	N
22	205	CD	CALL
23	37	25	N
24	64	40	N
25	94	5E	LDE. (HL)
26	35	23	INC, HL
27	86	56	LDD, (HL)
28	225	E1	POP, HL
29	183	B7	OR, A
30	237	ED	SBCHL, DE
31	82	52	-
32	32	20	JRNZ
33	6	06	e
34	62	3E	LDA, N
35	2	02	N
36	50	32	LD(NN), A
37	43	2B	N
38	64	40	N
39	201	C9	RET
40	58	3A	LDA, (NN)
41	44	2C	N
42	64	40	N
43	79	4F	LDC, A
44	71	47	LDB, A
45	16	10	DJNZ
46	254	FE	e
47	13	0D	DEC, C
48	32	20	JRNZ
49	250	FA	e
50	24	18	JR
51	205	CD	e

Line #60: Key Press?

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	58	3A	LDA, (NN)
2	255	FF	N
3	56	38	N
4	183	B7	OR, A
5	200	C8	RET, Z
6	58	3A	LDA, (NN)
7	64	40	N
8	56	38	N
9	254	FE	CP, N
10	2	02	N
11	32	20	JRNZ
12	2	02	e
13	225	E1	POP, HL
14	201	C9	RET
15	38	26	LDH, N
16	1	01	N
17	37	25	DEC, H
18	254	FE	CP, N
19	8	08	N
20	32	20	JRNZ
21	4	04	e
22	46	2E	LDL, N
23	64	40	N
24	24	18	JR
25	53	35	e
26	254	FE	CP, N
27	72	48	N
28	32	20	JRNZ
29	4	04	e
30	46	2E	LDL, N
31	61	3D	N
32	24	18	JR
33	45	2D	e
34	254	FE	CP, N
35	64	40	N
36	32	20	JRNZ
37	4	04	e
38	46	2E	LDL, N
39	3	03	N
40	24	18	JR
41	45	2D	e
42	254	FE	CP, N
43	80	50	N
44	32	20	JRNZ
45	4	04	e
46	46	2E	LDL, N
47	67	43	N
48	24	18	JR
49	37	25	e
50	254	FE	CP, N
51	16	10	N
52	32	20	JRNZ
53	4	04	e
54	46	2E	LDL, N
55	64	40	N
56	24	18	JR
57	29	1D	e
58	254	FE	CP, N
59	48	30	N
60	32	20	JRNZ
61	4	04	e
62	46	2E	LDL, N
63	61	3D	N
64	24	18	JR
65	21	15	e
66	254	FE	CP, N
67	32	20	N
68	32	20	JRNZ
69	4	04	e
70	46	2E	LDL, N
71	3	03	N
72	24	18	JR
73	5	05	e
74	254	FE	CP, N
75	40	28	N
76	192	C0	RET, NZ
77	46	2E	LDL, N
78	67	43	N
79	6	06	LDB, N
80	1	01	N
81	229	E5	PUSH, HL
82	14	0E	LDC, N
83	61	3D	N
84	195	C3	JP
85	37	25	N
86	64	40	N
87	6	06	LDB, N
88	2	02	N
89	24	18	JR
90	246	F6	e

Line #61: Move Plus

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	14	0E	LDC, N
2	100	64	N
3	205	CD	CALL
4	37	25	N
5	64	40	N
6	94	5E	LDE, (HL)
7	35	23	INC, HL
8	86	56	LDD, (HL)
9	225	E1	POP, HL
10	235	EB	EXHL, DE
11	5	05	DEC, B
12	40	28	JRZ

Line #61: Move Plus *(Cont'd)*

STEP#	DECIMAL	HEX	STATEMENT
13	3	03	e
14	25	19	ADDHL,DE
15	24	18	JR
16	3	03	e
17	183	B7	OR,A
18	237	ED	SBCHL,DE
19	82	52	-
20	62	3E	LDA,N
21	59	3B	N
22	188	BC	CP,H
23	200	C8	RET,Z
24	62	3E	LDA,N
25	64	40	N
26	188	BC	CP,H
27	200	C8	RET,Z
28	6	06	LDB,N
29	64	40	N
30	125	7D	LDA,L
31	184	B8	CP,B
32	56	38	JRC
33	3	03	e
34	144	90	SUB,B
35	24	18	JR
36	250	FA	e
37	254	FE	CP,N
38	61	3D	N
39	208	D0	RET,NC
40	229	E5	PUSH,HL
41	42	2A	LDHL,(NN)
42	40	28	N
43	64	40	N
44	94	5E	LDE,(HL)
45	35	23	INC,HL
46	86	56	LDD,(HL)
47	26	1A	LDA,(DE)
48	254	FE	CP,N
49	191	BF	N
50	40	28	JRZ
51	7	07	e
52	62	3E	LDA,N
53	128	80	N
54	18	12	LD(DE),A
55	19	13	INC,DE
56	18	12	LD(DE),A
57	19	13	INC,DE
58	18	12	LD(DE),A
59	209	D1	POP,DE
60	114	72	LD(HL),D
61	43	2B	DEC,HL
62	115	73	LD(HL),E
63	19	13	INC,DE
64	26	1A	LDA,(DE)
65	254	FE	CP,N
66	140	8C	N
67	40	28	JRZ
68	5	05	e
69	14	0E	LDC,N
70	51	33	N
71	195	C3	JP
72	37	25	N
73	64	40	N
74	62	3E	LDA,N
75	3	03	e
76	50	32	LD(NN),A
77	43	2B	N
78	64	40	N
79	225	E1	POP,HL
80	201	C9	RET

Line #70: Move Zero

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	14	0E	LDC,N
2	101	65	N
3	205	CD	CALL
4	37	25	N
5	64	40	N
6	94	5E	LDE,(HL)
7	35	23	INC,HL
8	86	56	LDD,(HL)
9	35	23	INC,HL
10	126	7E	LDA,(HL)
11	6	06	LDB,N
12	1	01	N
13	5	05	DEC,B
14	14	0E	LDC,N
15	68	44	N
16	98	62	LDH,D
17	107	6B	LDL,E
18	183	B7	OR,A
19	237	ED	SBCHL,BC
20	66	42	-
21	79	4F	LDC,A
22	9	09	ADDHL,BC
23	62	3E	LDA,N
24	59	3B	N
25	188	BC	CP,H
26	32	20	JRNZ
27	7	07	e
28	14	0E	LDC,N
29	71	47	N
30	205	CD	CALL
31	37	25	N
32	64	40	N
33	24	18	JR
34	222	DE	e
35	62	3E	LDA,N
36	64	40	N
37	188	BC	CP,H
38	40	28	JRZ
39	244	F4	e
40	6	06	LDB,N
41	64	40	N
42	125	7D	LDA,L
43	184	B8	CP,B
44	56	38	JRC
45	3	03	e
46	144	90	SUB,B
47	24	18	JR
48	250	FA	e
49	254	FE	CP,N
50	61	3D	N
51	48	30	JRNC
52	231	E7	e
53	229	E5	PUSH,HL
54	98	62	LDH,D
55	107	6B	LDL,E
56	54	36	LD(HL),N
57	128	80	N
58	35	23	INC,HL
59	54	36	LD(HL),N
60	140	8C	N
61	35	23	INC,HL
62	54	36	LD(HL),N
63	128	80	N
64	42	2A	LDHL,(NN)
65	40	28	N
66	64	40	N
67	209	D1	POP,DE
68	115	73	LD(HL),E
69	35	23	INC,HL
70	114	72	LD(HL),D
71	14	0E	LDC,N
72	52	34	N
73	195	C3	JP
74	37	25	N
75	64	40	N

Line #71: Reject Direction

STEP#	DECIMAL	HEX	STATEMENT
0	83	53	"S"
1	14	0E	LDC,N
2	36	24	N
3	205	CD	CALL
4	37	25	N
5	64	40	N
6	14	0E	LDC,N
7	33	21	N
8	205	CD	CALL
9	37	25	N
10	64	40	N
11	14	0E	LDC,N
12	37	25	N
13	205	CD	CALL
14	37	25	N
15	64	40	N
16	14	0E	LDC,N
17	32	20	N
18	205	CD	CALL
19	37	25	N
20	64	40	N
21	33	21	LDHL,NN
22	42	2A	N
23	64	40	N
24	53	35	DEC,(HL)
25	32	20	JRNZ
26	3	03	e
27	225	E1	POP,HL
28	225	E1	POP,HL
29	201	C9	RET
30	14	0E	LDC,N
31	51	33	N

Line #71: Reject Direction *(Cont'd)*

STEP#	DECIMAL	HEX	STATEMENT
32	205	CD	CALL
33	37	25	N
34	64	40	N
35	14	0E	LDC,N
36	52	34	N
37	195	C3	JP
38	37	25	N
39	64	40	N

Line #100: Plus Screen Address

STEP#	DECIMAL	HEX	STATEMENT
0	68	44	"D"
1	46	2E	(PERIOD)
2	46	2E	(PERIOD)

Line #101: Zero Screen Address

STEP#	DECIMAL	HEX	STATEMENT
0	68	44	"D"
1	46	2E	(PERIOD)
2	46	2E	(PERIOD)
3	46	2E	(PERIOD)

```
(23) 33D.....      (7 PERIODS)
(24) 255 LPRINT"I"
(25) 1000 DATA83,1,5,191,205,37,64,201
(26) 1011 RESTORE:FORN=0TO7:READD:POKEL+N:NEXTN
```

And now, after running this program, add the following lines and run the program again to test line 33:

```
(27) 254 FORN=1TO20
(27) 255 FORM=1TO80:NEXTM:CLS:FORM=1TO80:NEXTM,N:
     STOP
```

If all is well, the screen should flash on and off 20 times each time the program is run. *If this does not happen, load the last saved program and check for errors!*

The only difference between the contents of line 33 and line 32 is that step 3 loads the B register with 128D or 80H, which is the graphics "clear" block when printed.

To add line 32 to our program, make the following additions or changes and delete lines 254-256 and then save the program:

```
(29) 32D.....      (7 PERIODS)
(30) 255 LPRINT" "     (A SPACE IN QUOTES)
(31) 1000 DATA83,1,5,128,205,37,64,201
```

And, to test this line, run the program and then add the following line:

```
(32) 260 STOP
```

When the program is run again, the screen should be "cleared". After running the program, try this:

```
(33) LPRINT"! ! ! ! !"     (NO LINE NUMBER)
```

You should see the screen flash very quickly!

If your program is working fine so far, we are now ready to build our game around the line packing framework we have set up.

The Game *"Plus Zero"*

In the game, *"Plus Zero"*, the player moves a plus around the screen with the arrow keys. The machine moves a zero. The player tries to cause the plus and zero to collide. At a collision, the player's score is increased by the number of dots on the screen. These dots are left behind as the zero moves. A "high score" is also kept from game to game. When the zero comes to a screen boundary, all dots are erased and the zero seeks a new direction. It may "bounce" at the screen's edge two or three times while seeking this new direction. Only 20 of these bounces are allowed until the game is over. However, if a "hit" or collision occurs, the counter is reset to 0. If the plus runs into one of the dots left by the zero, the player loses and the game is over. Five playing speeds are offered, with the fifth being rather fast. Pressing CLEAR will drop the program from the machine language routines back to Basic and represents giving up.

Line Packing *"Plus Zero"*

Since this game is offered as an example of line packing only, a description of what each packed line does will not be included. But, since the complete machine language listing for each line is included with descriptive titles, it should be possible for the reader to discover how the game works.

From here on, all of the packed lines will interrelate and can't be tested separately as packed. So we are going to pack all remaining lines, one after the other. Then the actual Basic of the program will be added, after which the program should be done. If all goes well, you will have a nice little program otherwise, you will have to try again!

Let's get started - delete lines 250 to 1020. Then type in these lines:

```
(24)    6D.....(57 PERIODS)...
(25)  250Q0=PEEK(16548)+PEEK(16549)*256+13:
      Q2=INT(Q0/256):Q1=Q0-Q2*256:
      POKE16421,195: POKE16422,Q1:
      POKE16423,Q2
(26)  255LPRINTCHR$(6)
(27)  1000DATA68,1,132,129, and so on from
machine line listing of line #6...
(28)  1010L=PEEK(16424)+PEEK(16425)*256-1
(29)  1011M=57:RESTORE:FORN=0TOM:READD:
      POKEL+N,D:NEXTN
(30)  1012FORN=0TOM:PRINTPEEK(L+N);:NEXTN:STOP
```

When this program is run line 6 will be packed. *Do not run this program more than once,* since the packed line would be called without other necessary lines having been packed as yet!

We see (at 34) the line of periods where the packing will occur. LPRINT is connected to line 2 in (35). Since the line in (34) is a "D" line, (36) returns with the address of the "D" line. The data to be packed is in (37). (More than one data line may be necessary for some packing). The line in (38) finds the starting address for the line to be packed, including the "D" or "S" at its start. The line is packed by (39) and, as a

double check, (40) prints out what has just been packed. It's a good idea to check this list against the Z80 listing each time a line is packed.

When line 6 seems to be packed, go on to the next unpacked line, line 35. Do this to pack it:

```
ENTER:   35D....(41 PERIODS).....
EDIT:    255LPRINTCHR$(35)
ENTER: 1000DATA83,6,32,62, and so on from
machine language listing of line#35
EDIT:  1011M=41: The rest of line stays the
same
```

Now, run the program and line 35 should be packed (run only once!).

Follow this process of entering a "D" line with periods for each line to be packed. Change lines 255, 1000 and 1011 each time. Run the program once each time.

Lines 100 and 101 need not be packed. They may be entered as follows:

```
100D..
101D...
```

Lastly, delete lines 255 through 1020 and enter the Basic lines as shown in the Basic listing for "Plus Zero" (Figure 13). Be sure to save the program before you run it!

The game should now be ready to run. If things go wrong, you may have to start over. Or you could try to see what kind of thing went wrong, and redo the line or lines that seem related to that problem. Using machine language can be very frustrating since every byte has to be correct. There are no error traps, such as Basic has, and things can really crash!

Good luck, and have fun!

```
1 GOTO250
2   (PACKED SUBROUITNE: LINE FINDER.
    36 STEPS.)
5   (PACKED SUBROUTINE: SCREEN PAINTER.
    13 STEPS.)
6   (PACKED DATA:       DATA FOR ZERO MOVE.
    57 STEPS.)
32  (PACKED SUBROUTINE: CLS.
    7  STEPS.)
33  (PACKED SUBROUTINE: ALL ON.
    7  STEPS.)
35  (PACKED SUBROUTINE: SCREEN FLASH.
    41 STEPS.)
36  (PACKED SUBROUTINE: SET ZERO DIRECTION.
    28 STEPS.)
37  (PACKED SUBROUTINE: UP SWEEP.
    15 STEPS.)
50  (PACKED SUBROUTINE: PRESET PLUS AND ZERO.
    38 STEPS.)
51  (PACKED SUBROUTINE: SET PLUS ON SCREEN.
    31 STEPS.)
52  (PACKED SUBROUTINE: SET ZERO ON SCREEN.
    19 STEPS.)
53  (PACKED SUBROUTINE: COUNT DOTS.
    21 STEPS.)
55  (PACKED SUBROUTINE: GAME CONTROL.
    51 STEPS.)
60  (PACKED SUBROUTINE: KEY PRESS.
    90 STEPS.)
61  (PACKED SUBROUTINE: MOVE PLUS.
    80 STEPS.)
70  (PACKED SUBROUTINE: MOVE ZERO.
    75 STEPS.)
71  (PACKED SUBROUTINE: REJECT DIRECTION.
    39 STEPS.)
100 (PACKED DATA:       PLUS SCREEN ADDRESS.
    2  STEPS.)
101 (PACKED DATA:       ZERO SCREEN ADDRESS.
    3  STEPS.)
250 Q0=PEEK(16548)+PEEK(16549)*256+13:Q2=INT(Q0/
    256):Q1=Q0-Q2*256:POKE16421,195:POKE16422,Q1
    :POKE16423,Q2
260 H=0
270 CLS:PRINT@202,CHR$(34);"PLUS ZERO";CHR$(34);
    "  (C) 1981 LEO CHRISTOPHERSON":PRINTTAB(7)"CO
  NNECT TAPE AUX PLUG TO AMP FOR SOUND EFFECTS."
280 PRINT@325,"TO START, ENTER SPEED FACTOR: 1 (
  SLOW) TO 5 (FAST)."
290 PRINT:PRINTTAB(23)"INSTRUCTIONS:":PRINT"(1)
  PRESS ";CHR$(34);"CLEAR";CHR$(34);" TO END ACT
  ION."
300 PRINT"(2) USE THE ARROW KEYS TO MOVE THE PLU
  S."
310 PRINT"(3) IF THE PLUS AND ZERO HIT EACH OTHE
  R, YOUR SCORE INCREASES BY THE NUMBER OF DOTS
  LEFT ON THE SCREEN BEHIND THE ZERO."
320 PRINT"(4) IF YOU HIT A DOT, YOU'RE OUT!"
330 PRINT"(5) IF YOU ALLOW THE ZERO TO BOUNCE OF
  F THE SCREEN EDGE 20 TIMES WITHOUT SCORING A H
  IT, THE GAME IS OVER!"
340 K$=INKEY$:IFK$=""THEN340
350 IF(K$<"1")OR(K$>"5")THEN340ELSEONVAL(K$)
    GOTO360,370,380,390,400
360 S=255:GOTO410
370 S=200:GOTO410
380 S=150:GOTO410
390 S=100:GOTO410
400 S=70
410 POKE16428,S:CLS:T=0:
    PRINT@405,"ONE MOMENT PLEASE!"
420 LPRINTCHR$(6):L=PEEK(16424)+PEEK(16425)*256:
    FORN=1TO56:R=RND(56):A=L+N:B=L+R:C=PEEK(A):
    POKEA,PEEK(B):POKEB,C:NEXTN
430 CLS:POKE16426,20:POKE16427,1:LPRINT"$27":
    ONPEEK(16427)GOTO440,450,490,480
440 CLS:IFPEEK(16426)=0THEN480ELSE510
450 POKE16426,0:LPRINT"5":T=T+PEEK(16426):
    LPRINT"#":PRINT@408,"HIT: SCORE ";T
460 IFT>HTHENH=T
470 PRINT@472,"HIGH SCORE ";H:GOTO420
480 LPRINT"#":PRINT@405,"GAME OVER: SCORE ";T:
    GOTO500
490 LPRINT"#":PRINT@404,"YOU'RE OUT: SCORE ";T
500 PRINT@472,"HIGH SCORE ";H
510 PRINT:
    PRINT@527,"ANOTHER GAME?  (Y)ES  OR  (N)O"
520 PRINTTAB(22)"(C)HANGE SPEED?"
530 K$=INKEY$:IFK$=""THEN530ELSEIFK$="Y"THEN410
    ELSEIFK$="C"THEN270ELSEIFK$="N"THENEND
540 GOTO530
```

SPACE MARAUDER

Charles A Quante

Have you ever had the urge to put on your space helmet, grab your laser sword and stomp the living daylights out of a few green meanies? Well, here is your chance. Marauder is based on a popular arcade game. You must move and fire your cannon, destroying as many of the marauders as possible. Since the program is written in Basic, naturally it is not as fast as the arcade version. To make up for this, it is somewhat more difficult to destroy the creatures than it is in their version.

Several of the lines of the program are interesting in themselves. After the first page of instructions is printed, line 280 waits for a key to be pressed to continue. This in itself is not unusual, but instead of using INKEY$, this line peeks at location 14591. This location contains zero, until a key is pressed. This technique is also used in line 520 to determine if a key has been pressed. By making Q$ equal to the control keys, (J,K,L), it is possible to use a FOR...NEXT loop to find the key that has been pressed. This method eliminates three IF...THEN statements, and improves the speed of the game.

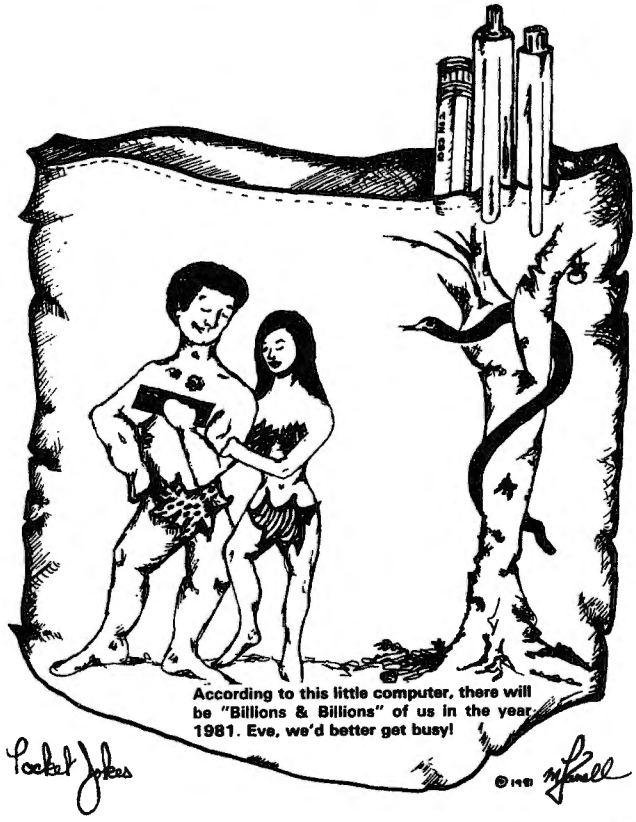
The sound producing routine is stored in line 430, and is from the article, *"Sounding Off!"* in the May/Jun 1980 issue of 80-U.S. Journal. This routine is actually a machine language routine that is packed into a Basic string. The line is packed into the string by lines 6000 to 6080. The program tests to see if you are in disk Basic or Level II Basic, and sets up the USR command accordingly.

The program is complete, but like all programs, there is plenty of room for improvement. To help those who wish to customize the program, here is a more detailed look:

10	Clear memory for strings and define integer variables.
20-200	Graphic Cover-draws a saucer, using poke graphics.
230-270	first page of instructions
280	Peeks at 14591 until a key is pressed
290-320	Second page of instructions
400-420	Strings for the cannon and Marauders
430-	Machine language routine packed in string X$
450-460	Sets up USR for DOS or L2 Basic
470-480	Chooses random creature
490-	Fills L( ) with the print locations for the creatures
500-510	Prints playing field
520-	Peeks location 14591 again
530-	Checks to see if pressed key s is a control key.
540-	If key was control key, branch to command
550-610	If key was not control key, move creature and branch to fire routine
620-	Move cannon left, but not off the screen
630-	Produce cannon fire sound
640	Wait till key is released, then branch to user fire routine
650-	Move cannon right, but not off the screen
660-690	User fire routine, user must hit the creature dead center when its 'mouth' is open.
700-780	Creature fire routine

```
10 CLS:CLEAR200:DEFINTA-Z:AP=15360
20 DL=63:FORI=0TO31:POKEI+AP,191:POKEAP+DL,191:D
   L=DL-1:NEXT
30 DL=127:FORI=64TO90:POKEI+AP,191:POKEDL+AP,191
   :DL=DL-1:NEXT:POKE91,135:POKE100,139:POKE95+AP
   ,188:POKE96+AP,188
40 DL=191:FORI=128TO148:POKEI+AP,191:POKEDL+AP,1
   91:DL=DL-1:NEXT:DL=170:FORI=149TO150:POKEI+AP,
   143:POKEDL+AP,143:DL=DL-1:NEXT:DL=168:FORI=151
   TO154:POKEI+AP,131:POKEDL+AP,131:DL=DL-1:NEXT
50 DL=255:FORI=192TO208:POKEI+AP,191:POKEDL+AP,1
   91:DL=DL-1:NEXT:DL=238:FORI=209TO210:POKEI+AP,
   131:POKEDL+AP,131:DL=DL-1:NEXT:DL=233:FORI=214
   TO220:POKEAP+I,176:POKEDL+AP,176:DL=DL-1:NEXT
60 DL=319:FORI=256TO270:POKEAP+I,191:POKEDL+AP,1
   91:DL=DL-1:NEXT:POKE271,143:POKE304,143:POKE27
   6,176:POKE299,176:DL=298:FORI=277TO284:POKEAP+
   I,191:POKEDL+AP,191:DL=DL-1:NEXT
70 DL=383:FORI=320TO333:POKEI+AP,191:POKEDL+AP,1
   91:DL=DL-1:NEXT:POKE369,131:POKE334,131:POKE33
   9,188:POKE364,188:DL=363::FORI=340TO348:POKEI+
   AP,191:POKEDL+AP,191:DL=DL-1:NEXT
80 DL=447:FORI=384TO386:POKEI+AP,191:POKEDL+AP,1
   91:DL=DL-1:NEXT:DL=444:FORI=387TO396:POKEI+AP,
   143:POKEDL+AP,143:DL=DL-1:NEXT
90 POKE448+AP,191:POKE511+AP,191:POKE449+AP,131:
   POKE510+AP,131
100 POKE512+AP,191:POKE575+AP,191:POKE513+AP,180
   :POKE574+AP,184:POKE514+AP,176:POKE573+AP,176
110 DL=639:FORI=576TO579:POKEI+AP,191:POKEDL+AP,
   191:DL=DL-1:NEXT:POKE580+AP,189:POKE635+AP,190
   :POKE581+AP,188:POKE634+AP,188:POKE582+AP,176:
   POKE633+AP,176:POKE583+AP,144:POKEAP+632,160:P
   OKE603+AP,160:POKE612+AP,144
120 POKE604+AP,190:POKEAP+611,189:DL=610:FORI=60
   5TO607:POKEI+AP,191:POKEDL+AP,191:DL=DL-1:NEXT
130 DL=703:FORI=640TO648:POKEI+AP,191:POKEDL+AP,
   191:DL=DL-1:NEXT:POKE649+AP,188:POKE694+AP,188
   :POKE650+AP,180:POKE693+AP,184:POKE651+AP,176:
   POKE692+AP,176:POKE666+AP,184:POKE677+AP,180:D
   L=676:FORI=667TO671:POKEI+AP,191:POKEDL+AP,191
   :DL=DL-1:NEXT
140 DL=767:FORI=704TO735:POKEI+AP,191:POKEDL+AP,
   191:DL=DL-1:NEXT
150 DL=831:FORI=768TO781:POKEI+AP,191:POKEDL+AP,
   191:DL=DL-1:NEXT:POKE817+AP,191:POKE816+AP,191
   :POKEAP+815,191:DL=814:FORI=782TO799:POKEI+AP,
   143:POKEDL+AP,143:DL=DL-1:NEXT
160 DL=895:FORI=832TO845:POKEI+AP,191:POKEDL+AP,
   191:DL=DL-1:NEXT:FORI=1TO3:POKEDL+AP,191:DL=DL
   -1:NEXT
170 DL=959:FORI=896TO927:POKEI+AP,191:POKEDL+AP,
   191:DL=DL-1:NEXT
180 DL=1023:FORI=960TO991:POKEI+AP,191:POKEDL+AP
   ,191:DL=DL-1:NEXT
190 PRINT@527,"<M> <A> <R> <A> <U> <D> <E> <R>";
200 PRINT@847,"<M> <A> <R> <A> <U> <D> <E> <R>";
210 FORI=1TO10
220 PRINT@847,"                                  ";
   :FORJ=1TO300:NEXT:PRINT@847,"<M> <A> <R> <A> <
   U> <D> <E> <R>";:FORJ=1TO300:NEXT:NEXT
230 CLS:PRINTTAB(14)"* <M> <A> <R> <A> <U> <D> <
   E> <R> *":PRINTSTRING$(64,"=");
240 PRINT:PRINT"   YOU ARE PITTED AGAINST SIXTY
   OF THE MOST HORRIBLE MARAUDERS":PRINT"OF SPACE
   .  YOUR MISSION, SHOULD YOU DECIDE TO ACCEPT I
   T, IS TO":PRINT"DESTROY ALL SIXTY OF THEM, (OR
    AS MANY AS YOUR SKILL ALLOWS YOU"
250 PRINT"TO), THUS SAVING DEAR OLD MOTHER EARTH
   !":PRINT:PRINT"   THERE IS A PROBLEM THOUGH.
    WHILE YOU ARE TRYING TO DESTROY":PRINT"THEM,
   THEY WILL TRY TO DESTROY YOU! IF THEY DESTROY
   YOUR THREE"
260 PRINT"CANNONS, THE GAME IS OVER AND THE EART
   H IS LOST!  BUT DON'T":PRINT"DESPAIR, YOU MIGH
   T DO BETTER NEXT TIME.":PRINT
270 PRINTTAB(19)"PRESS ANY KEY TO CONTINUE."
280 IFPEEK(14336+255)=0THEN280ELSECLS
290 PRINT"   OH YES, ONE MORE THING, TO MOVE YOU
   R CANNON TO THE LEFT":PRINT"PRESS THE <U> KEY.
    TO MOVE RIGHT, THE <O> KEY, AND TO FIRE THE"
   :PRINT"<I> KEY.  ALSO SINCE YOUR EQUIPMENT IS
   NOT EAXCTLY NEW, IT MAY"
300 PRINT"BE A LITTLE SLOW TO RESPOND.":PRINT
310 PRINTTAB(16);:INPUT"DO YOU ACCEPT THE CHALLE
   NGE";G$:G$=LEFT$(G$,1)
320 IFG$="Y"THEN330ELSEPRINT"OKAY, C-H-I-C-K-E-N
   -!-!  GOODBYE!":END
330 CLS
340 CLEAR200
350 RANDOM
360 DEFINTA-Z
370 DIMP(60),L(60),K(60),E(12)
380 Z=960:MI=3:Q$="UIO"
390 S$=STRING$(5,191)
400 C$=CHR$(32)+CHR$(184)+CHR$(191)+CHR$(180)+CH
   R$(32)
410 A$(1)=CHR$(32)+CHR$(167)+CHR$(140)+CHR$(155)
   +CHR$(32):A$(2)=CHR$(32)+CHR$(182)+CHR$(140)+C
   HR$(185)+CHR$(32)
```

```
420 A$(3)=CHR$(32)+CHR$(182)+CHR$(140)+CHR$(185)
  +CHR$(32):A$(4)=CHR$(32)+CHR$(182)+CHR$(188)+C
  HR$(185)+CHR$(32)
430 X$="1:1:1:1:1:::1::1:1:1:1:1:1:1:1:1:1:1:1:1:
  :1:1:1:1:1:1:1:1:1:1:1:1:1:1:1:1:1"
440 X1=PEEK(VARPTR(X$)+1):X2=PEEK(VARPTR(X$)+2):
  XO=X2*256+X1
445 GOTO6000
450 IFPEEK(16396)=201THENL=0ELSEL=1
460 IFL=1THENDEFUSR=XO:CMD"T":ELSEPOKE16526,X1:P
  OKE16527,X2
470 PRINTTAB(14);:INPUT"RAPID-FIRE OR SINGLE SHO
  T CANNON";G$:G$=LEFT$(G$,1):IFG$="R" RP=1:CLSE
  LSECLS
480 FORI=1TO60:A1=RND(4):P(I)=A1:NEXT
490 X=64:S=1:E=12:FORJ=1TO5:FORI=STOE:L(I)=X:X=X
  +5:NEXT:X=X+128-60:S=E+1:E=E+12:NEXT
500 FORI=1TO60:PRINT@L(I),A$(P(I));:NEXT
510 PRINT@897,S$;:PRINT@915,S$;:PRINT@934,S$;:PR
  INT@952,S$;:PRINT@Z,C$;
520 IFPEEK(14591)<>0 G$=INKEY$:POKE16537,ASC(G$)
  :ELSEPOKE16537,0:GOTO550
530 FORI=1TO3:IFMID$(Q$,I,1)=G$THEN540ELSENEXTI
540 ON IGOTO620,630,650
550 T=RND(66)+34:R=RND(60):POKEXO+6,1
560 IFPEEK(L(R)+15361)=32THEN520
570 ONP(R)GOTO580,590,600,610
580 POKEXO+16,T:M=USR(0):P(R)=2:PRINT@L(R),A$(P(
  R));:GOTO700
590 POKEXO+16,T:M=USR(0):P(R)=1:PRINT@L(R),A$(P(
  R));:GOTO700
600 POKEXO+16,T:M=USR(0):P(R)=4:PRINT@L(R),A$(P(
  R));:GOTO700
610 POKEXO+16,T:M=USR(0):P(R)=3:PRINT@L(R),A$(P(
  R));:GOTO700
620 IFZ=960PRINT@Z,C$;:GOTO550ELSEZ=Z-1:PRINT@Z,
  C$;:GOTO520
630 POKEXO+6,3:POKEXO+34,5:M=USR(0)
640 IFPEEK(14591)=0THEN660:ELSEIFRP=1THEN660:ELS
  E640
650 IFZ=1018PRINT@Z,C$;:GOTO550ELSEZ=Z+1:PRINT@Z
  ,C$;:GOTO520
660 AA=(Z-64)+2
670 PRINT@AA,"[";:TT=(AA-64)+15360:TP=PEEK(TT):I
  FTP=32ORTP=140THEN680:ELSEPRINT@AA,CHR$(32);:G
  OTO520
680 IFTP=140PRINT@AA,CHR$(32);:PRINT@TT-15361,ST
  RING$(3,32);:GOSUB860:ST=ST+1:PRINT@0,"HITS SO
   FAR:";;ST;:IFST=60THEN800:ELSE:GOTO520
690 PRINT@AA,CHR$(32);:AA=AA-64:IFAA<63THEN520EL
  SEPRINT@AA,"[";:GOTO670
700 PRINT@45,"CANNONS:";;MI;
710 W=RND(12):IFE(W)=1THEN710ELSES=48+W
720 FORI=STOWSTEP-12:IFPEEK(15360+L(I)+1)=32NEXT
  ELSE740
730 E(W)=1:GOTO710
740 MM=L(I)+15360+66
750 IFPEEK(MM)<>32THEN770
760 PRINT@MM-15361,CHR$(92);:PRINT@MM-15361,CHR$
  (32);:MM=MM+64:IFMM-15361>1023GOTO520ELSEGOTO7
  50
770 PP=MM-15360
780 IFPP=Z+1 ORPP=Z+2 OR PP=Z+3 ORPP=Z+4PRINT@MM
  -15360,CHR$(32);:PRINT@Z,STRING$(5,32);:FORI=1
  TO50:NEXT:PRINT@Z,C$;:FORI=1TO50:NEXT:PRINT@Z,
  STRING$(5,32);:Z=960:PRINT@Z,C$;:MI=MI-1:GOTO7
  90ELSEGOTO520
790 IFMI=0PRINT@54,MI;:ELSEGOTO820
800 PRINT@74,"GAME OVER - TRY AGAIN (<Y>ES OR <N
  >O)?";
810 G$=INKEY$:IFG$="Y"THENRUN330:ELSEIFG$="N"THE
  NEND:ELSEGOTO810
820 IFMI=2THENH=129:FORI=1TO60:L(I)=L(I)+H:NEXT:
  ELSEH=128:FORI=1TO60:L(I)=L(I)+H:NEXT
830 POKEXO+6,1:POKEXO+16,4
840 FORI=1TO60:IFI<13THENPRINT@L(I)-H,STRING$(5,
  32);
850 IFK(I)=0PRINT@L(I),A$(P(I));:M=USR(0):NEXT:G
  OTO520:ELSENEXT:GOTO520
860 TZ=TT-15362:FORI=1TO60:IFL(I)<>TZNEXT:RETURN
  :ELSEK(I)=1:RETURN
6000 DATA 74
6010 DATA 33,1,2,14,255,62,1,254,2,40,21,254
6020 DATA 3,40,38,17,200,100,237,97,67,16,254,23
  7
6030 DATA 105,67,16,254,21,32,243,201,17,1,5,123
6040 DATA 237,97,71,16,254,237,105,71,16,254,60,
  32
6050 DATA 243,21,32,239,201,17,100,5,123,237,97,
  /1
6060 DATA 16,254,237,105,71,16,254,61,32,243,21,
  32
6070 DATA 239,201
6080 RESTORE:READH:FORN=0TO73:READD:POKEXO+N,D:H
  =H-1:NEXT:IFH<>0PRINT"CHECK-SUM ERROR. PLEASE
  CHECK YOUR DATA STATEMENTS. (6000-6070)":END:E
  LSEPRINT"WHEN 'READY' APPEARS, DELETE LINE 44
  5,THEN TYPE RUN330.":DELETE6000-6080
```



Custom Operating System

How to build a Custom Operating System with NEWDOS/80

David Busch
Ravenna, Ohio

TRS-80 Model I owner: would you like to own a 44 track drive, with 85 free granules to play with on data diskettes? Your Radio Shack drive just might have nine extra tracks that Tandy didn't tell you about.

Or, do you wish that your system had the intelligence to realize that you already know what programs are contained on your system disk in drive 0, and that when you type DIR, what you would really like to see is a directory of drive 1?

Perhaps you own drives with different track counts, say one 35 track and one 40 track unit, and get hung up making full diskette copies from one to the other. Is the biggest problem in your life remembering to disable the automatic keyboard debounce routine when you try to play a cassette ADVENTURE which you have transferred to disk?

With NEWDOS80, you can build a custom disk operating system that will take some of these little hassles out of your life. Take, for example, the discovery of a 44 track Tandy disk drive.

Before NEWDOS80, we had used a 35 track Disk Operating System, and a 40 track DOS on our twodrive system. The hardware consists of one older Shugart-type 35 track Radio Shack disk drive, and one newer Tandon-built Radio Shack drive. The Shugart is used as drive 0, with a 35 track system disk containing the 40 track NEWDOS+, plus some frequently used utilities and applications programs. The Tandon drive, which functioned well as a 40 track unit, was used with data diskettes almost exclusively. The configuration was fairly convenient, with several exceptions. One of these was the need to switch to a 35 track DOS whenever I wanted to copy 35 track disks.

Enter NEWDOS80. This disk operating system is smart enough to remember the track counts of every drive in your system. In our case, we entered the following PHYSICAL DRIVE (PDRIVE) command to tell the system that our first drive had only 35 tracks:

PDRIVE,:0,:0,DTC=35

The first :0 indicated the drive in which the SYSTEM diskette that was being modified was located, and the second :0 pointed to the drive for which the new specification was being made. With the DTC (drive track count) set equal to 35, the system will never try to look for or FORMAT more than 35 tracks on that drive, unless the user specifically says to. Only DIR and FREE will see the extra tracks, if these have been formatted during some previous incarnation of the disk.

For instance, a 40 track drive could be configured as with a 35 track default value. The operator could still access the other five tracks (if present) by one means or another, if desired. For Example:

COPY :1=40 TO :2=40 11/18/80

This would allow a full copy of a 40 track disk even if drives 1 and 2 had been set to 35 tracks via PDRIVE. Why would anyone use a drive to less than its full capacity? Someone with three 35 track units, and one 40 track unit might want to have all drives the same for convenience. Or, an operator waiting to install a data separator might wish to lock out the inner tracks to avoid problems in the meantime.

This leads us to the discovery of the 44 track disk drive. Like most, we had heard of rumors that the latest Radio Shack drives were 77 track units in disguise, but had no 77 track DOS available to check this out. NEWDOS80 supports track counts up to 96. When trying to FORMAT a disk to 77 tracks, it was found that the drive hung up somewhere after 40 tracks.

After 40 tracks? FORMAT the same disk to 41 tracks! It worked. Then we tried 42, 43, 44 and 45 tracks. Only at the last did the drive finally hang up again. When we had our first 88-gran data diskette (85 grans after the system information had been

placed on it), we didn't really believe that we had those extra usable tracks.

SUPERZAP agreed with us. It refused to look beyond 40. We had forgotten about PDRIVE. Once we reconfigured our SYSTEM for the new 44 track drive, SUPERZAP was willing to look at the extra sectors and in fact, copies some data from one of the lower sectors to the new. It worked.

NEWDOS80 allows the operator to build the sort of oddball disk operating system desired. There is more to PDRIVE, but let's digress to the new SYSTEM command first to show just how it can be used to build a custom DOS.

SYSTEM allows the changing of certain default system options to other parameters. For example, on booting up, such features as keyboard debounce, JKL printing, and the new DFG entry into Mini-DOS are all enabled. (JKL and DFG are keys which are simultaneously pressed to activate the feature.) Passwords are automatically disabled. These options can be changed, for specific system diskettes so that, for example, on booting up the keyboard debounce routine is automatically disabled, with no further operator steps required. If all your ADVENTURES and other debounce-hating programs are contained on such a disk, you can run them without giving the shift-up-arrow (normal debounce disabling during boot-up) a second thought. You only need to remember to reboot for the new disk.

The SYSTEM options are given labels, currently AA to AS. These are flagged with numeric or alpha characters to show how the options are set, much like the J=Y (justify=yes) in Scripsit shows that right justification has been turned on (or off), or the PL=88 (Page Length is 88 lines) can indicate the page length. SYSTEM is invoked merely by typing SYSTEM, the password if any, the drive number in which the disk being updated is mounted, and the options being changed. For example:

SYSTEM,:0,AA=Y

This would re-enable password checking any time that particular system disk is in use. More than one parameter can be changed at a time:

SYSTEM,:0,AA=Y,AI=Y,AM=20

With these changes, passwords would be enabled, the system would know that lowercase had been installed (SUPERZAP and DEBUG use this information), and the system would make 20 tries to complete a given disk I/O task before declaring an error (10 is the original value).

After the SYSTEM command has been entered correctly, NEWDOS80 displays a several-line listing of all the current SYSTEM option flags and their values, allowing a quick check. The new options are put into effect the next time that disk is booted.

The SYSTEM options, and their uses, are too numerous to cover here, but I'd like to discuss a few of them and how they lend themselves to custom DOS building.

AL tells the system the number of drives. If you have only two drives, then AL=2 will limit the system from checking more than two when searching for programs, space for SAVE's, etc. This saves some time that can build up if a lot of disk I/O is done.

AN is the default drive number for the DIR command. If your system disk remains in drive 0 most of the time, and all your swapping is done between your other drive(s), it makes little sense to see a directory of drive 0 every time you type in DIR. If AN=1, then DIR will give you a directory of drive 1 every time you enter it. AUTO DIR,CMD"DIR", and Mini-DOS will also show you drive 1's directory. If you still want to see the directory of drive 0 you can enter DIR :0. This option can save a few keystrokes and some frustration.

Under normal conditions, when you type SAVE"filename" without a drive specification, the system will search for a file by that name and if none are found, attempt to create the file on the lowest number drive on which there is space. That's not so great if you have space on your SYSTEM disk, but would prefer that all the programs be saved on one of the others. The previous solution had been to remember to add the drive specification to the file name: SAVE"TEST/BAS:1". The AO system option eliminates that hassle. The lowest number drive that will be searched and used for program saving can be specified. For example, if AO=1, then drive 0 will be ignored and your SYSTEM disk remains in a pristine state in future saves.

There are other SYSTEM options. The system can be set to operate in RUN ONLY mode (AB), the BREAK key can be disabled (AG). The features in Basic that converts input lowercase characters to uppercase (SET), can be enabled or disabled. Or, even though passwords are enabled, COPY can be used without password checking (AR). With the latter, protected files can be copied by users without the correct password, even though they cannot be KILLED, LISTED, or even RUN (depending upon the level of protection.)

Custom system building with NEWDOS can range beyond the convenient to the essential. PDRIVE has additional features that allow using the TRS-80 with a variety of non-Tandy hardware. More than the drive track count (DTC) can be specified - the type of interface and drive itself can be configured using the command. Like SYSTEM, PDRIVE displays its current flag values whenever it is invoked. For example the most frequently used system disk displays the following PDRIVE information for drive 0:

DTC=35 TI=A TD=A DDST=17 DDGA=2

We have already discussed DTC. TI (type of interface) shows the type disk controller used in the system. The A used in this case indicates that the standard TRS-80 Model I interface is used. Implemented in the current release of NEWDOS80 is OMIKRON's 8 inch disk drive interface. Those using MAPPER I and MAPPER II will be pleased to note that they can use NEWDOS80, and mix 5 and 8 inch drives without difficulty. If the OMIKRON interface is used, TI=B.

NEWDOS80's documentation notes that an additional interface, C, has been defined for LOBO's expansion interface, but that this has not yet been implemented. LOBO users will have to check with Apparat to see if/when their hardware will be compatible with NEWDOS80.

TD (type of drive) is used to specify the type of drive each unit is. The A in the example indicates that the drive is a 5 inch single density, single sided drive. B is used with 8 inch single density single sided drives. Not implemented are options C through H, which are various combinations of single and double sided, single and double density drives.

It is possible that the next option to be implemented will be for 5 inch single sided, double density drives. The necessary zaps should be made available either by Apparat, some independent source, or a manufacturer of double density controllers, such as PERCOM.

The other two options available under PDRIVE, DDST (Default Directory Sector Track) and DDGA (Default Directory Granule Allocation) allow additional system flexibility for TRS-80 Model I owners, but also hold forth the possibility of making disks crated under NEWDOS80 incompatible with TRSDOS, NEWDOS+, or other operating systems. DDST indicates the track used for the beginning of the disk directory. Track 17 is the standard for non-NEWDOS/80 systems, because it is halfway across the disk (just about) and is thus fairly close to the read/write head at either extreme of the disk. When DDST does not equal 17, other DOS's will not know how to find the directory. Use this option only if you understand what you are doing.

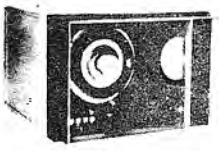
DDGA allows changing the number of granules allocated to the directory. Two granules is standard, and should be used when a diskette will be accessed by non-NEWDOS80 systems. But, if you find yourself needing extra FDE's (file directory entries), for example, when putting a lot of programs on your 77 track disk, this option will allow you to allocate up to six granules for the directory, for a total of 222 usable FDE's on a data diskette.

All the changes discussed in this article are options available to the operator under NEWDOS80. The entries need to be made just once, and then every time a disk is booted up, the combination of features selected by the user become active. Disk operating systems must obviously be written to handle the needs of a wide variety of hardware configurations and operator needs. But, equally as obvious, some features not needed by every user, or needed every time the system is operated. Building a custom DOS saves the operator time and frustration by tailoring the system's response to the needs of the moment.●

9 TRACK TAPE DRIVE FOR A MICRO-COMPUTER ?

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.



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).

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

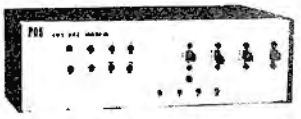
POS-100 NRZ1 TAPE DRIVE CONTROLLER/FORMATTER – For over 2 years Pacific Office Systems has been quietly selling its POS-100 NRZ1 Tape Drive Controller/Formatter for "Pertec" industry standard NRZ1 tape transports and the S-100 computer. The POS-100 operates with 2K of 8080 or Z-80 subroutines to Write 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 Pertec-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.

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

Circle 52

EDAS EDAS

A sophisticated Editor & Assembler setting the standard for the '80 Model I & Model III. All EDAS commands and SOURCE text can be entered in either upper case or lower case. Direct assembly form memory or disk by means of *GET assembler directives. This gives text buffer capacity equal to your drive configuration! 30,000 bytes of symbol table.

Direct assembly to disk or memory for faster debugging operations! DOS "system" command functions KILL, DIR, FREE, and LIST are available from within the environment of EDAS.

The Editor, with renumber, maintains command syntax identical to the BASIC editor. Global change permits you to alter a string throughout a designated range of lines while block move relocates lines of text.

EDAS is priced at $79 plus $3 S&H. A 72-page manual included.

cmdfil

Now you can append two or more CMD files and/or SYSTEM tapes. Perform transfer to & from disk/tape of SYSTEM/CMD modules with offset capabilities. Read VTOS ISAM overlays. More! $20

MISOSYS SERIOUS SOFTWARE

LDOS is the latest generation of operating systems for the TRS-80 computers. LDOS is completely documented in an extensive operating manual and total on-going support from some of the most knowledgeable people in the industry is provided. In short, LDOS is THE operating system to use with the TRS-80 computers!

$139.00
+$3 S&H

LDOS 4.1

dsmblr

Complement your assembly language tools with this Z80 disassembler which produces screen, printer, cassette, or disk file output. A two-pass process provides SYMBOLS for 16-bit address and 8-bit relative references. EQUates & ORG are generated. Read SYSTEM programs & display load address range. $20 (DSMBLR I for non-disk use is $15)

Specify Model I or Model III

master charge

VISA

MISOSYS - Dept U
5904 Edgehill Drive
Alexandria, Virginia 22303
703-960-2998 MicroNET 7C140,310
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 BØØK

THE BØØK must be a part of your Z-80 language tools. Volume I gives you access to all math operations in your Level II ROM including ASCII-Binary conversions. Included is a symbol table of the entire machine noting over 500 addresses. Volume II tells you everything you wanted to know about the Level II I/O - printer, keyboard, video, and cassette routines are fully explained. Each volume has a fully-commented listing of all the routines discussed. THE BØØKs will save you hours of assembler program development time. Don't start programming without THE BØØKs. Each volume is priced at $14.95 + $1.50 S&H or buy both for $24.95.

Circle 53

X-Reference

For Models I and III

Tom Huber

There are times when programming can be a nightmare: Your client wants a minor change to his accounting package; your program crashes after months of use because an unusual condition triggered a previously unused/untested routine; a project you set aside three months ago needs to be finished; or even worse, you inherit someone else's program to debug and make work correctly. If these and other problems plague you as a programmer, then you need Radio Shack's Cross Reference Utility #26-2008 $14.95!

The program will produce a cross reference table of line numbers, variables, or reserved words, or a combination of the three for any Basic program that has been saved on disk or tape. The output may be optionally sent to any printer with 80 or more columns.

Documentation and program operation is straightforward and easy to use. The manual explains all options, including how to get the disk version on disk and how to back up your diskettes. The disk version resides in the 5500H to 5D40H range; tape, 4300H to 4932H.

Once loaded from SYSTEM or DOS, the program will provide the user with a series of prompts, to be answered with a Y for yes and a N for no. The prompts, in order, are:

PRINTER?
LINE NUMBERS?
VARIABLES?
RESERVED WORDS?
DISK? (disk version) or
READY CASSETTE

The disk version will prompt FILESPEC? If the program is not a Basic program (or if it is saved in ASCII, which is not wanted) the message CAN'T READ. TRY AGAIN? will be displayed. A negative answer will return to DOS, otherwise a positive answer will restart the prompts.

A no response to the prompt DISK? will bring up the cassette prompt. The cassette Basic program will load only at 500 baud under its own routine so you can't generate a table from a 1500 or 250 baud tape. Also, a dropped bit and subsequent bad load will make itself known *only* after you have started to generate the table (and particularly under Reserved Words). You will really know when disk reserved words show up in your tape only programs. Readjusting the volume will usually take care of this problem. The flashing asterisks are still there during tape loads.

If you have opted for a printer output the results will start immediately to both screen and printer. The output looks like a duplicate of the screen, even to the ending prompt END OF LIST - RESTART? This leads me to believe that a function similar to DUAL (Model III) has been invoked for this option. By the way, if your printer is not ready (or not there) the program will hang up until a printer is made available.

If you did not opt for printer output, you will be prompted to PRESS ENTER WHEN READY. The screen will scroll until everything has been displayed. The SHIFT@ combination will stop scrolling. Pressing ENTER will then terminate the run, while pressing any other key will continue the scrolling. By holding the SHIFT and alternately toggling the @ key and another key, you can control the display very nicely.

At the end of the program you will be prompted END OF LIST - RESTART? An N will drop to the DOS or Level II (depending on which version you run) while a Yes will bring up the prompt SAME PROGRAM? If you enter a Y here, the first four prompts will again be given; otherwise, the program will restart.

Many disk owners who have NEWDOS will want to know if this new utility outperforms the REF function. See Figure 1 for a comparison of both utilities.

Figure 2 is a sample program, while Figures 3 through 5 are outputs of XREF and NEWDOS REF.

The output is neat in appearance and uniform, though the leading zeroes are confusing until you get used to them. The printer output contains the ending prompt.

The Model III DOS has a "REF" utility, which allows reference to one item at a time, be it reserved word, line number or variable. However, it functions as an INSTRING function on all but reserved words, producing somewhat misleading results. The XREF utility functions as described equally on the the disk Model III.

Looks like Radio Shack has scored again with another fine utility. The biggest objections are lack of getting just one variable or line number at a time; not being co-resident with Basic for editing purposes; and no leading zero suppression in the table. The advantages are: prompt driven; neat appearance; both tape and disk compatible; large program capability; reserved word table; separation of variable type and line numbers only.

The advantages appearing to outweigh the faults would make this the preferred utility. It is an easy to use, versatile and powerful program.

Figure 1

Descriptions	Radio Shack's XREF	NEWDOS80 REF
4K Level I (Mod I or III tape)	No	No
16K Level II Mod I/III 500 baud tape	Yes	No
16K Mod III 1500 or Level I 250 baud tape	No	No
Disk, any size memory, Mod I	Yes	Yes*
Disk, any size memory, Mod III	Yes	Unknown
Full size Basic program to limit of memory	Yes	No*
Co-resident with Basic	No	Yes*
Screen output	Yes	Yes
Screen pause in scan	Yes	Yes
Printer output	Yes	Yes
Line numbers only	Yes	No
All real numbers and line numbers	No	Yes
All variables by full name	Yes	No
All variable types by separate rows	Yes	No
All variables, including fielded types	Yes	Yes
Dimensioned variables	Yes	Yes
Defined (as in DEFDBL, etc.) variables	Yes	Yes
All reserved words	Yes	No
Single numbers, variables	No	Yes
Multiple occurance - count given	No	Yes
Multiple occurance - separate reference	Yes	No
Prompt driven program	Yes	No
Requires parameters to execute	No	Yes

*REF function can cause OM? (out of memory) error with large programs while XREF makes use of normal program overhead and stack space to handle large programs including some slightly larger than would run in a given memory size. This is particularly true with disk Basic programs where Basic is not required for XREF to execute.

Figure 2

Sample, unusable program with bugs (many). Doesn't really do anything either.

```
1 DEFDBL A-B
2 DIM X(20)
10 FOR ALPHA = 100 TO 10 STEP -3
20 IF AL=98 THEN 300
30 FOR B!=1 TO 3
40 ON B! GOSUB 50, 60, 70
45 GOTO80
50 PRINT"THIS";:RETURN
60 READD$
63 PRINT" "D$" ";
65 RETURN
70 PRINT" ILLUSTRATES ALPHA"
75 RETURN
79 LSET C$=MKI$(B!)
80 PRINT:PRINT
81 NEXTAL
90 DATA SAMPLE, PROGRAM, ILLUSTRATES, ALPHA, 1, 3, 4, 28
100 DATA TYPE, DATA
110 STOP
120 IF B!=3 THEN B!=B!+1
130 C=SIN(B!)*COS(1.2)[4/3*SQR(ABS(ALPHA))
140 PRINTUSING" ##";B!
150 PRINT TAB(10)"THE"TAB(B!)"END"
152 OPEN"R",1,"SAMPLE/TXT"
154 FIELD1, 2 AS C$, 2 AS K$, 2 ASC$, 3 ASCO$
156 CLOSE
160 END
166 IF ERR/2+1 = 23 AND ERL=60 THEN STOP ELSE RESUME100
173 IF A=3 OR B=2 THEN 20 ELSE 30
```

Figure 3

Output from XREF left and REF right. Note the LINE NUMBERS under XREF return only line numbers. This is not true of NEWDOS which returns all numbers whether constants, line numbers, buffer numbers or whatever.

```
LINE     REFERENCES
00020    00173
00030    00173
00050    00040
00060    00040
00070    00040
00080    00045
00100    00166
00300    00020
END OF LIST - RESTART?
```

```
  1   30 120 152 154 166
  2   154/3 166 173
  3   10 30 120 130 154 173
  4   130
 10   10 150
 20   2 173
 23   166
 30   173
 50   40
 60   40 166
 70   40
 80   45
 98   20
100   10 166
300   20
```

Figure 4

Variable references under XREF (above) and REF (below). Note the XREF returns all variables as well as the word AS in the fielding statement. Also note the O$ from line 154. The program was tricked into a typical syntax error by not spacing between AS and CO$, creating the reserved word ASC and variable O$.

Note the ALPHA reference. While Basic will recognize only the AL portion, the utility recognized the entire word. Microsoft's BASCOM would also make the distinction. A nice feature not normally needed.

```
VARIABLE REFERENCES
A        00001 00173
AL       00020 00081
ALPHA    00010 00130
AS       00154 00154
B        00001 00173
B!       00030 00040 00079 00120 00120 00120 00130 00140 00150
C        00130
C$       00079 00154
D$       00060 00063
K$       00154
O$       00154
X        00002
END OF LIST - RESTART?
```

```
  A   1 173
  AL  10 20 81 130
  B   1 30/! 40/! 79/! 120/!3 130/! 140/! 150/! 173
  C   79/$ 130 154/$
  D   60/$ 63/$
  K   154/$
  O   154/$
  X   2(
```

Figure 5

Reserved words output from XREF returns all reserved words found in the program including mathematical and logical operators. Altogether quite impressive.

```
VERB      REFERENCES
END       00160
FOR       00010 00030
NEXT      00081
DATA      00090 00100
DIM       00002
READ      00060
GOTO      00045
IF        00020 00120 00166 00173
GOSUB     00040
RETURN    00050 00065 00075
STOP      00110 00166
ELSE      00166 00173
DEFDBL    00001
RESUME    00166
ON        00040
OPEN      00152
FIELD     00154
CLOSE     00156
LSET      00079
PRINT     00050 00063 00070 00080 00080 00140 00150
TAB(      00150 00150
TO        00010 00030
USING     00140
ERL       00166
ERR       00166
THEN      00020 00120 00166 00173
STEP      00010
+         00120 00166
-         00001 00010
*         00130 00130
/         00130 00166
[         00130
AND       00166
OR        00173
=         00010 00020 00030 00079 00120 00120 00130 00166 00166
          00173 00173
ABS       00130
SQR       00130
COS       00130
SIN       00130
MKI$      00079
ASC       00154 00154
END OF LIST - RESTART?
```

Memory Size Reset

For the TRS-80 Model I

James F Williams
Rocky Mount, North Carolina

How many times have you gotten your Basic program loaded or written, and realized that you needed a printer driver, or a monitor, or a renumber routine for which you failed to reserve memory? The normal procedure is to CSAVE your Basic program, type SYSTEM *ENTER*/0 *ENTER*, set the appropriate memory size, load the machine language program, then reload your Basic program.

MEMORY will eliminate some of those time consuming cassette operations, and even allow Decimal/Hex conversions to boot. There are many ways to accomplish a change of memory size without destroying Basic. Most methods involve calculating low and high order address bytes and poking them into memory. This method is prone to calculating and keying errors. Dennis Kitsz sent me a marvelous utility "KEEP03" which allows you to reset memory size and much more, but is must itself be loaded into reserved memory first. MEMORY is loaded only when you need it, and cannot conflict with anything you have in memory, because it and its stack load and operate completely within the Basic input buffer, an area that must be kept free for Basic. This means of course, that when you return to Basic, the MEMORY program will be destroyed, but most of the operations it does are "one shot" affairs anyway. If you need it again, it only takes seconds to load.

To use MEMORY, BREAK your Basic program and type SYSTEM *ENTER* MEMORY *ENTER*. After loading is complete, MEMORY initializes itself and the current MEMORY SIZE is displayed in Hex and Decimal. It should be noted that the actual value stored in 40B1H is two less than the value keyed in and displayed. For that reason the value displayed for a 16K machine with no memory reserved will be 8001H (32769), two bytes higher than the highest possible address. This is done so that the beginning of the reserved area is two bytes higher than the highest byte used by Basic.

The command string is displayed on line 2. The one letter commands are activated without the *ENTER* key.

B RETURN TO (B)ASIC

Pressing this key will return you to Basic command mode (Ready prompt). If you use this exit, and do not reset Memory Size, Basic variables will remain intact.

S (S)YSTEM PROMPT

This will probably be the most frequently used exit. From the SYSTEM prompt, you may load the machine language program for which you reset MEMORY SIZE.

R (R)UN BASIC PROGRAM

If you need to execute your Basic program immediately, there is no reason to exit and type RUN *ENTER*. Just press R. This command (like RUN) will clear all Basic variables.

M RESET (M)EMORY SIZE BOUNDARY

After pressing "M", a right arrow will appear. This means that the program is expecting Hex or Decimal input. Hex input is designated by following the Hex numbers with an "H". Corrections may be made with the backspace key, and entry is terminated with the *ENTER* key. The maximum Hex range is FFFFH and the maximum Decimal range is 65529. After *ENTER* is pressed, you will return to the original prompt with the new MEMORY SIZE on the first line. This command will clear all Basic variables.

C CONVERT

This command allows you to perform any number of Hex-Dec or Dec-Hex conversions. Just type in one number and the conversion is displayed next to it. The original prompt may be re-entered by hitting *ENTER* alone.

There are several interesting programming techniques used in MEMORY that may be used separately in other programs.

IMMEDIATE EXECUTE (LINES 240-250, 310-320)

By loading a JP (entry point) into location 41E2H-41E4H, no manual initialization will be necessary. After the tape is read, execution will begin immediately at the address overlayed by a separate ORG. If you plan to be able to return to Basic and use the SYSTEM command, you must load a RET back into location 41E2H.

MEMORY RESET (LINES 550-590)

This routine could be used with any machine language program to protect itself in high memory. Simply load ADDRESS-2 of the beginning of the machine language program to be protected in HL, execute the given code, and return to Basic with a JP 6CCH.

EXECUTE BASIC PROGRAMS FROM MACHINE PROGRAMS (LINES 880-910)

This routine was discovered by Robert Shiels, a North Carolina State Computer Science major. This would be useful if you were loading Basic and machine code as a block of memory and wanted to execute it from the machine language program. However, to make that work, you must set a few Basic pointers as well.

CONVERT A DECIMAL STRING TO A VALUE IN DE (LINE 770)

Point HL to the beginning of an unsigned decimal string ending in a 0 byte, then CALL 1E4FH. On exit DE will contain the value of the string in the range 0-65529. John Hind was the first to show me that one.

CONVERT VALUE IN HL TO UNSIGNED DECIMAL STRING AND DISPLAY AT CURRENT CURSOR POSITION (LINE 1150)

Load HL with value and CALL 0FAFH. See "Binax Kibuff"by John Blair in the September 1980 issue of *80-Microcomputing* for more information.

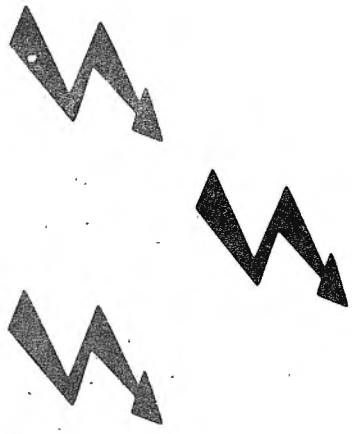
In the same issue David Cornell wrote an excellent article and program "Free Space", that uses the input buffer to contain his program. He describes the limits of the SYSTEM stack and gives a great deal of other very useful information, as well as a super program. I believe many "one shot" type utilities could use this method very effectively, if they can be written to be contained in that limited space. ●

```
00100 ;MEMORY SIZE RESET BY JAMES F. WILLIAMS
00110 ;
00120 ;   THIS PROGRAM EXECUTES IMMEDIATELY AFTER LOADING
00130 ;WITHOUT INITIALIZATION, AND DISPLAYS THE CURRENT
00140 ;MEMORY SIZE IN HEX AND DECIMAL.
00150 ;
00160 ;                COMMANDS
00170 ;
00180 ;   B   RETURN TO (B)ASIC
00190 ;   S   JUMP TO (S)YSTEM COMMAND
00200 ;   R   (R)UN BASIC PROGRAM
00210 ;   M   RESET (M)EMORY SIZE BOUNDARY (HEX OR DEC)
00220 ;   C   (C)ONVERT HEX-DEC, DEC-HEX. USE H AFTER HEX NO.
00225 ;          USE <ENTER> ALONE TO EXIT LOOP.
00230 ;
00240         ORG     41E2H           ;SET FOR AUTO EXECUTE
00250         JP      ENTRY
00260         ORG     41E6H           ;BOTTOM OF INPUT BUFFER
00265 ;CONVERT ROUTINE
00270 CLOOP   CALL    INPUT           ;HEX-DEC INPUT
00280         LD      A,H
00290         OR      L               ;HL=0?
00300         JR      NZ,CLOOP        ;NO, LOOP BACK
00305 ;COMMAND LOOP
00310 ENTRY   LD      HL,41E2H        ;RESTORE SYSTEM CMD
00320         LD      (HL),201
00330         CALL    1C9H            ;CLEAR SCREEN
00340         LD      HL,(40B1H)      ;MEM SIZE TO HL
00350         INC     HL              ;ADD 2
00360         INC     HL
00370         CALL    DISPLY          ;DISPLAY HEX
```

```
00380           CALL    DECDPY          ;DISPLAY DECIMAL
00390           LD      HL,PROMPT       ;CMD PROMPT
00400           CALL    28A7H           ;DISPLAY IT ON SCREEN
00410           CALL    49H             ;WAIT 'TIL KEY PRESSED
00420           CP      'B'             ;B?
00430           JP      Z,6CCH          ;YES, TO BASIC
00440           CP      'R'             ;R?
00450           JR      Z,RUN           ;YES, TO RUN ROUTINE
00460           CP      'S'             ;S?
00470           JP      Z,2B2H          ;YES, TO SYSTEM CMD
00480           CP      'C'             ;C?
00490           JR      Z,CLOOP         ;YES, TO CONVERT
00500           CP      'M'             ;M?
00510           JR      N2,ENTRY        ;NO, TO CMD LOOP
00515 ;MEMORY RESET ROUTINE
00520           CALL    INPUT           ;HEX-DEC INPUT
00530           DEC     HL              ;SUBTRACT 2
00540           DEC     HL
00550           LD      (40B1H),HL      ;SET BOUNDARY
00560           LD      DE,0FFCEH       ;CLEAR 50
00570           ADD     HL,DE
00580           LD      (40A0H),HL      ;SET STRING AREA PTR
00590           CALL    1E7AH           ;BASIC CLEAR FIXES PTRS
00600           JR      ENTRY
00605 ;INPUT ROUTINE
00610 INPUT     LD      A,94
00620           CALL    33H             ;PRINT RIGHT ARROW
00630           LD      HL,(4020H)      ;SAVE CURRENT CURSOR POS
00640 SCAN      CALL    49H             ;STROBE UNTIL KEY PRESSED
00650           CP      0DH             ;CR?
00660           JR      Z,EVAL          ;YES, EVALUATE RESULT
00670           CALL    33H             ;NO, PRINT CHARACTER
00680           JR      SCAN            ;LOOP 'TIL CR
00690 EVAL      LD      DE,(4020H)      ;CURSOR POS IN DE
00700           DEC     DE              ;BACK UP TO LAST CHAR
00710           LD      A,(DE)          ;PUT IT IN A
00720           CP      'H'             ;H?
00730           JR      Z,HEX           ;YES, HEX ENTRY
00740           INC     DE              ;NO, NEXT SCREEN LOC
00750           XOR     A
00760           LD      (DE),A          ;PUT TERMINATOR ON SCREEN
00770           CALL    1E4FH           ;VALUE OF DEC STR IN DE
00780           EX      DE,HL           ;VAL IN HL
00790           CALL    DISPLY          ;DISPLAY HEX VALUE
00800           JR      CR
00810 DECDPY    CALL    DECML
00820 CR        LD      A,13
00830           CALL    33H             ;PRINT CR TO SCREEN
00840           RET
00850           ORG     4288H           ;MAKE ROOM FOR STACK
00855 ;RUN SUBROUTINE
00860 RUN       CALL    1E7AH           ;BASIC CLEAR ROUTINE
00870           LD      HL,(40A4H)      ;START OF BASIC PROGRAM
00880           DEC     HL
00890           PUSH    HL
```

```
00900           POP     DE
00910           JP      1D1EH           ;TO ROM
00915 ;CONVERT HEX STRING TO VALUE IN HL
00920 HEX       LD      DE,0            ;CLEAR DE
00930           EX      DE,HL           ;0 TO HL
00940           DEC     DE              ;FOR LOOP
00950 NEXT      INC     DE              ;CHAR ON SCREEN
00960           LD      A,(DE)          ;IN A
00970           CP      'H'             ;H?
00980           JR      Z,DECDPY        ;YES, DISPLAY DEC
00990           CP      '0'             ;LESS THAN ASCII 0?
01000           JR      C,NEXT          ;YES, GET NEXT CHAR
01010           ADD     HL,HL           ;NO, SHIFT HL LEFT 4
01020           ADD     HL,HL
01030           ADD     HL,HL
01040           ADD     HL,HL
01050           CP      'A'             ;LETTER?
01060           JR      NC,LTR          ;YES
01070           ADD     A,7             ;NO, ADJUST
01080 LTR       SUB     55              ;ADJUST TO REAL VAL
01090           OR      L               ;MERGE A AND L
01100           LD      L,A             ;BACK TO L
01110           JR      NEXT            ;NEXT CHAR
01115 ;CONVERT VALUE TO DECIMAL STRING
01120 DECML     LD      A,20H
01130           CALL    33H             ;PRINT SPACE TO SCREEN
01140           PUSH    HL              ;SAVE VALUE
01150           CALL    0FAFH           ;DECIMAL STR TO SCREEN
01160           POP     HL              ;RESTORE VALUE
01170           RET
01175 ;VALUE IN HL TO HEX STRING
01180 DISPLY    LD      C,4             ;FOR 4 DIGITS
01190           PUSH    HL              ;SAVE VALUE
01200           LD      A,20H
01210           CALL    33H             ;PRINT SPACE
01220 DIGIT     LD      B,4             ;BIT COUNT
01230           XOR     A
01240 SHFT      ADD     HL,HL           ;SHIFT HL LEFT 4
01250           RLA                     ;SAVE CARRY IN A
01260           DJNZ    SHFT            ;LOOP 4 TIMES
01270           ADD     A,48            ;ADJUST TO ASCII 0-9
01280           CP      58              ;0-9?
01290           JR      C,PRINT         ;YES
01300           ADD     A,7             ;NO, ADJUST TO A-F
01310 PRINT     CALL    33H             ;DISPLAY DIGIT
01320           DEC     C               ;4 DISPLAYED?
01330           JR      NZ,DIGIT        ;NO, DO IT AGAIN
01340           LD      A,'H'           ;YES
01350           CALL    33H             ;DISPLAY 'H'
01360           POP     HL              ;RESTORE VAL
01370           RET
01380 PROMPT    DEFM    'BSRMC'         ;COMMAND STRING TO SCREEN
01390           DEFB    0
01400           END     ENTRY
```

Software Review
Ball Turret Gunner
Instant Software Inc
Peterborough, NH 03458
$9.95



Ball Turret Gunner is a one player game in which you try to destroy enemy fighters in a race against the clock.

The program, written in Basic, puts one in control of a Ball Turret in deep space trying to destroy incoming alien gnat-class fighters. The game is straightforward, giving one a few commands ranging from moving in two dimensions to firing the stratolaser. One's fight to save the good guys' territory is often challenging, depending upon which sector the turret is located in. Players have the choice of difficulty they want to play. The sector is determined from this information.

When I played, the game continued to amaze me. Its graphic representation of each sector was enough to leave me befuddled as to how the author managed to map a full universe in memory because the turret is able to rotate a full 360 degrees horizontally and 90 degrees vertically. Movement may be diagonal and can be stopped altogether by hitting the space bar.

One outstanding feature of the graphics is that the movement is very smooth. Chalk another one up to good programming. The game is played in real time, this being an advantage because the display is constantly shown and not constantly redrawn.

The display is unique in the way it shows the approaching fighters. At a great distance, they appear as just moving stars, but as they come closer they become little T.I.E. fighters that grow as they near. If the gnats (that is what they are called in this game) get too close to the turret, they crash into the unfortunate gunner. Apparently, gnats are kamikazes in space.

The program documentation is very thorough. In fact, you can't help but understand how to play the game. The people who wrote the manual should be commended.

Finally, the Ball Turret Gunner game is a very good 'shoot-em-up' program. While input is through INKEY$ routines, the idea is not new and it does not take a great deal of intellecutal prowess to play, the program does work without errors.

Pat Perez

Cyborg Wars
Stratagem Cybernetics
286 Corbin Place
Brooklyn, NY 11235
$15 incl postage & handling

On a small asteroid somewhere in our galaxy, four nations of Androids have been engaged for several eons in constant warfare over the asteroid's limited resources... So begins the game of Cyborg Wars, Stratagem Cybernetics' first entry into the computer software market. Up to four players rule over a nation in an effort to bring peace (and one government) to the small asteroid.

The game is similar to the old Hammurabi and King games of a few years ago, but is a lot more stylized, and complex. In Cyborg Wars, a ruler has a country full of Android subjects, all falling into one of four categories. They are Breeders, Farmers, Workers and Soldiers. Breeders are the Androids that make more Androids. Farmers can be made into anything material on the asteroid's surface. Workers make guns and tanks and things for the fourth classification soldiers: they fight the battles.

On each turn of play, each ruler is given one new Android for every 20 Breeders he has. Androids can be converted in a limited manner. Breeders can be changed to anything, Framers can be made into anything except Breeders, Workers can be converted to Soldiers, but Soldiers must stay Soldiers.

Soldiers may be sent out as spies to neighboring countries to try to recover information about the enemy. Beware though, that spies can be captured and killed, and there goes another precious soldier that is hard to replace.

Once the conversions have been made and all the spies have returned with their information, one final decision must be made - to attack or not to attack the neighboring country. That may sound unfriendly, but the game is not won until a player has conquered the asteroid, or thirty-five turns go by without any outbreaks of war.

When attacking, the ruler has absolutely no control over strategy. That is all controlled by Android generals, so the best attack strategy is to find a country that is grossly outmatched.

The game always has four players, even if three of them are the computer. The game can be played for weeks in the solo mode, and it won't be much different when playing with three other humans.

This program is a good choice for a multi-player strategy game, offering an extended playing time over most other multi-player games, sometimes lasting up to two or more hours. When playing with more than one person, make sure there is something aside from playing the game to do, because all the information should be kept secret. The program does ask for a password at the beginning of each player's turn. This comes in very handy when playing with people who want to win at all costs.

If you are in the market for a multiple player game, this one is a good choice. It comes with an instruction booklet and a pad of special graph paper designed to help keep a record of everything. Have fun..

Pat Perez

Software Review
Air Traffic Controller
Creative Computing Software
PO Box 789-M
Morristown, NJ 07960
CS-3006
$9.95?

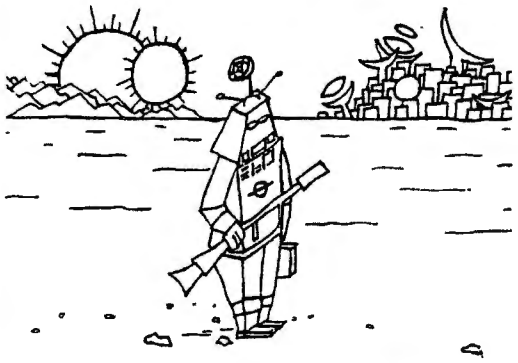
Air Traffic Controller is unusual in that a microcomputer software package seldom gets an entire full-page advertisement to itself.

This one deserves it. I had been apprehensive about buying ATC because I'm an instrument-rated pilot, and simulations often seem contrived to those familiar with the subject. This is not the case with ATC.

The simulation stays close to reality. There are some compromises, of course. Most of these tend to simplify the game. For example, every aircraft in your area is under your control; all aircraft have altitude reporting equipment; you only have to deal with two speeds of planes (240 knot jets and 120 knot propellor planes) and no helicopters; and both speeds of aircraft have the same turning radius. All of the aircraft follow your directions, and never get mixed up; there are no emergencies; and the planes climb and descend much faster than real planes.

The game is not easy to win, even with those simplifications. The skill level is set by how many minutes you give yourself to get all 26 planes to their destinations. The documentation for the program claims that no one has yet successfully completed the

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

Circle 60

Introducing . . .

OMNITERM

The Ultimate TRS-80™ Terminal Package . . . . At Last!

OMNITERM - The Ultimate TRS-80™ Terminal Package is now available after nearly 2 years in the making. Created as a result of dissatisfaction with existing terminal programs, **OMNITERM** is the **Most Powerful** and **Flexible** microcomputer intelligent terminal program at any price - **Bar None.** Now the **TRS-80™** can communicate with and transfer files to almost any computer system **Without** writing special software **OMNITERM** solves the problem of computer systems that send 32, 40, or 80 column lines to your 64 column **TRS-80™** by **Reformatting** your screen for easy reading. **OMNITERM** can compensate for incompatibilities with 7 different translation tables, one to and from each device. This capability even allows **OMNITERM** to use codes such as **EBCDIC**, and translate them to the **TRS-80™'s ASCII**. With **OMNITERM** you can even **Review** text that has **Scrolled Off** the **Screen!** At any time you can examine and change any of **OMNITERM's** settings, because **OMNITERM** gives you a **Full Status Display** of all functions. It works with all **ROM's** and **DOS's** since it uses only officially documented calls

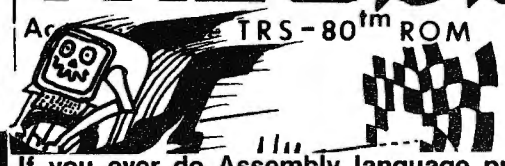
- **Spooled Printer**
- Single Key **Auto Signon**
- **Prompted** File Output
- True **Break** Key
- **X/Y Cursor Control**
- **Xon/Xoff** File Control
- **Repeat** Key
- **Lower Case** Support.
- Can **Echo** characters
- Sends **Special Characters** not on the **TRS-80** keyboard
- Control-G causes "**Beep**" sound and **Graphic Bell** to flash on the screen.
- Accepts all standard **VIDEOTEX™** control codes
- Can configure the **UART** for baud rates from 50-19,200 baud.
- Can send text **Slowly** for computers that work at "typist" speed.
- Keeps continuous count of **Parity, Framing,** and **Overrun** errors.
- **Saves** a special file with proper settings for each different use.
- Includes: 60 page manual, **Text Editor, Hex Conversion Utilities.**

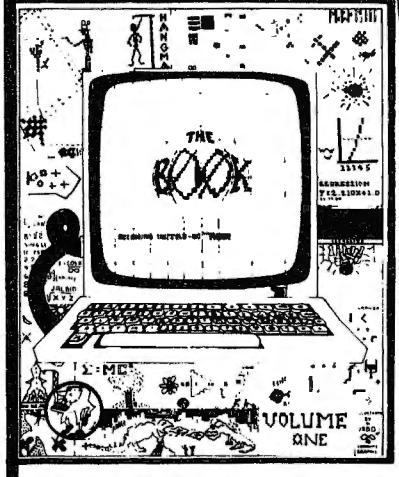
Everyone who sees **OMNITERM** is **Amazed** by its **Power** and **Ease of Use!** Requires **TRS-80** with 32K memory, 1 disk. $95 (plus shiping if C.O.D.). Call for shipment within 24 hours. Manual alone $15, refunded with purchase. Order by Visa, M/C or C.O.D. Personal checks delay 2-3 weeks. Mass. residents add 5% sales tax. Dealer Inquiries Invited.

lindbergh Systems

49C Beechmont Street, Worcester, MA 01609 (617) 799-2217

THE BØØK





If you ever do Assembly language programming, or you just want to know more about your TRS-80 ROM, "THE BOOK"s are for you.

*TRS-80 is a trademark of Tandy Corp.

Each volume is priced at: $14.95 + $1.50 S&H = $16.45 ($17.05 in VA) Overseas add $2.50 for air shipment

Insiders Software Consultants, Inc.
P.O. Box 2441, Dept. U 2
Springfield, VA 22152

☐ **Please send me Volume I of THE BOOK**

☐ **Please send me Volume II of THE BOOK**

NAME _____

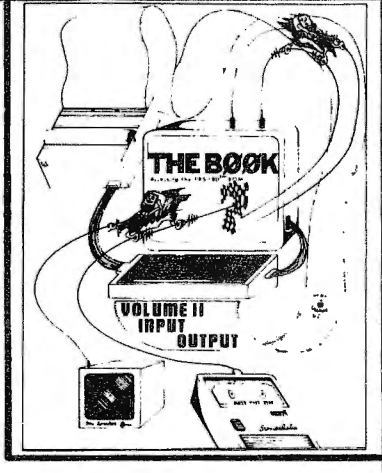
ADDRESS _____

CITY, STATE _____ ZIP CODE _____

Check payable to Insiders Software Consultants, Inc.
MASTER CHARGE MC Bank Code
VISA Exp. Date _____ Card Number _____

Signature _____

Dealer Inquiries Invited



Volume I: The most comprehensive book yet on the math routines in Level II ROM, Models I & III. Includes a fully commented listing from 0708H to 1607H and an incredibly complete map of the ROM and reserved areas of RAM.

Volume II: Everything you want to know about the video, keyboard, cassette, and printer driver routines. Learn how to write your own! Remarkably detailed listings illustrate well-commented source code. Complement Volume I, now.

Circle 61

simulation at the highest skill level (16 minutes). I believe it. In fact, it is not hard to blow the game at the lowest skill level (99 minutes). I find 60 minutes to be quite challenging, even though the documentation recommends 65 minutes as being easy enough for novice players.

Your simulated RADAR screen covers an area 15 by 25 miles. Aircraft in your area show up as blips, indicated by a letter which is the plane's call sign. Beside the letter blip for each plane is a single digit which tells you the plane's altitude in thousands of feet. Your area includes two airports, two radio-navigation beacons (nav-aids, for short), and five airways. Planes may be arriving from either direction on any of the airways, or may be taking off from one of the airports. The planes may be leaving on an airway or may be landing at one of the airports.

Your basic set of commands lets you instruct a plane what altitude you want it at, and lets you turn a plane either left or right by 45, 90, 135 or 180 degrees. Additional commands let you instruct a plane to do something special when it encounters a nav-aid -- turn westbound toward one of the airports, turn northwestbound toward the other airport, or enter a holding pattern. There is also a command to cancel previous turning and holding commands, and one to clear a plane to land.

ATC arrived eight days after I ordered it by phone. The production of the cassette is top quality. ATC is recorded in SYSTEM format on a quality cassette with a professional label. It comes in a real cassette box, which is further sealed in a plastic wrap. I had no trouble loading the program on my 16K Level II TRS-80, but then, I have never had the problems with cassettes that other TRS-80 owners are always complaining about. The documentation is packaged inside the cassette box, all neatly folded up.

The instructions seem formidable at first reading. There are only a few commands -- that isn't the problem. It is the amount of detail that promises to overload your brain and turn you into a cabbage: approach headings, overflight routings, fuel limitations, separation requirements, aircraft performance parameters (speed, turn radius, climb and descent rates), and the symbols used on the display.

While you're trying to absorb all of this detail, you will probably miss a few vital points. The ones that I missed that cost me the first few games were that 1) all inbound planes will go into a holding pattern when they reach a nav-aid, unless you have specifically instructed them not to; 2) airplanes cannot be descending the last mile to the airport; and 3) the instruction entry area on the screen will clear itself every few seconds, so that just because you hit ENTER and the command went away, this does not mean that the aircraft received your instruction -- you have to look for the

(Continued on page 120)

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 gived with progresively 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.

MOD-1 16K L-II $19.95

MULTIPLY

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-12,000 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.

$14.95

D & M SOFTWARE
1510 SOUTH 97TH
TACOMA, WASHINGTON 98444

Circle 64

CONVERT YOUR TRS-80 MODEL-I INTO A DEVELOPMENT SYSTEM

Now you can develop Z-80 based, *stand-alone* devices such as games, robots, instruments and peripheral controllers, by using your TRS-80 as a development system. The DEVELOPMATE plugs into the expansion connector of your TRS-80 and adds **PROM PROGRAMMING** and **IN-CIRCUIT-EMULATION** capabilities to your system (with or without expansion interface).

Complete instructions and sample schematics are included to help you design your own simple stand-alone microcomputer systems. THESE SYSTEMS CAN BE AS SIMPLE AS FOUR ICs - one TTL circuit for clock and reset, a Z-80, an EPROM, and one peripheral interface chip.

When the In-Circuit-Emulation cable is plugged into the Z-80 socket of your stand-alone system, the system becomes a part of your TRS-80. You can use the full power of your editor/assembler's debug and trace programs to check out both the *hardware and the software*. Simple test loops can be used to check out the hardware, then the system program can be run to debug the logic of your stand-alone device.

Since the program is kept in TRS-80 RAM, changes can be made quickly and easily. When your stand-alone device works as desired, you use the Developmate's PROM PROGRAMMER to copy the program into a PROM. With this PROM, and a Z-80 in place of the emulation cable, your stand-alone device will work by itself.

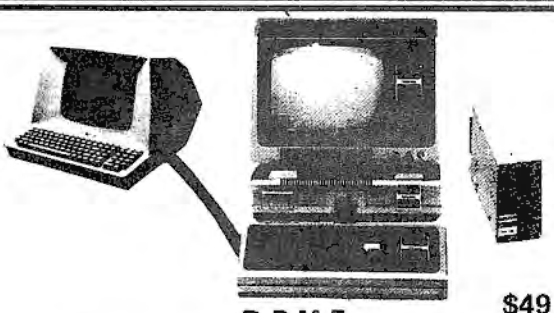
The DEVELOPMATE is extremely compact. Both the PROM programmer and the In-Circuit-Emulator are in one small plastic box only 3.2" x 5.4". A line-plug mounted power supply is included. The PROM programmer has a personality module which defines the voltages and connections of the PROM so that future devices can be accommodated. However, the system comes with a universal personality module which handles 2758, 2508(8K), 2716, 2516(16K), 2532(32K) as well as the new electrically alterable 2816 and 48016(16K EEPROMs).

The COMPLETE DEVELOPMATE 81 with software, power supply, emulation cable, TRS-80 cable and universal personality module is ONLY $329!
The PROM PROGRAMMER is available *separately* for ONLY $239.

ORION INSTRUMENTS
172 Otis Avenue Dept B Woodside CA 94062
(415) 851-1172
Master Charge and Visa phone orders accepted
California residents please add 6% sales tax

Circle 65

TWO USER MULTI-TASKING



$89

TSHARE VI.3
A SPLIT PERSONALITY FOR YOUR TRS – 80 MODEL I

- Interrupt driven executive patches to parent operating system NEWDOS 2.1 or TRSDOS 2.3.
- Allows two active users to share a single TRS-80.
- Execute BASIC or MACHINE LANGUAGE (above 7700 Hex).
- Joint access of disk files.
- Second user ties to HOST via modem, terminal, or 2nd TRS-80.
- Options for Expansion Interface, RS-232, TRS232, or HUH interface.
- SIMPLEX MODE for non serial-port users. Requires only a printer to act as second "screen". Jobs share the keyboard under user control.
- CONFIGURE UTILITY partitions memory usage and selects I/O.

$49

DBM5
MULTI – FEATURE DATA MANAGER

- Up to 20 user defined fields.
- Files extend across multiple diskettes.
- Supports up to four drives.
- Automatic single drive disk mount requests.
- Four data types and computational fields.
- Flexible REPORT GENERATOR.
- Fast assembly language sorts.
- MULTI-FIELD sorts and searches.
- Keyed access on any sorted field.
- Indexed relations between files.

TRS-80 tm Radio Shack/Tandy Corp.
NEWDOS tm Apparat, Inc.
TRS232 tm Small System Software
HUH tm California Computer Systems.

TO ORDER

ComSoft
203 ARDEN STREET
GLENDALE, CA 91203
(213) 649-0369

Circle 66

ROGER response.

A couple of the commands interact in a way that I, a pilot, did not anticipate. In particular, if you instruct a plane to turn 90 degrees and then immediately clear the same plane for the approach to the airport, the plane stops its turn. This is a shock if the reason that you were turning the plane was to line it up with the airport. This isn't consistent with real life, in that if you instruct a real pilot of a plane to make a similar turn and clear him to land (which is different than cleared for the approach), he will continue the turn as originally instructed, and then land.

The game is unforgiving; one mistake and you are through. If you let planes get too close, or make one use too much fuel, or if a plane leaving your area misses its departure fix or leaves at the wrong altitude, the game ends immediately. There is one mistake that you can get away with. If you clear a plane to land at the wrong airport, it will fly to the airport and then reject the landing. Then you can send him to the correct airport.

The game clock is quite accurate on my TRS-80. A useful feature is that you can speed up the clock by banging away on the space bar. This lets you get past the occasional periods when you have no planes to handle.

If there is a weak point, it is in the documentation. Not in substance, but in professionalism. The documentation contains sentences like: *The command AO causes the aircraft to descend to 0 altitude, but is also causes the aircraft to ignore all further instructions.* What is needed here is a little bit of proofreading. That shouldn't be hard for the distributors to do, since they publish a national magazine.

Errors in the advertising for ATC also seem to stem from poor proofreading, rather than an attempt to mislead. The full-page ad calls the clock "inimitable", which it is not. I suspect that they meant "inimical", or perhaps "indomitable". That same ad gives the price as $9.95, when the other ads and the catalog list it at $7.95 except for the Apple II version. And another ad incorrectly says that you must handle 27 planes, rather than 26.

There is also an annoyance in that when you're done playing ATC you have to turn your TRS-80 off and back on to do anything else. Pressing RESET seems to work at first, but Basic will not run correctly.

Air Traffic Controller is available on TRS-80 Level II 16K cassette.

Douglas L Pardee

Review
Revised Level II Manual
Radio Shack, $5.95

Radio Shack's Second Edition of the *Level II Reference Manual* is already in the hands of most purchasers of the TRS-80, but those who bought their machines before about May, 1980, have the old manual, recognizable by its staple binding. The new manual has a lay-open spiral binding, but the changes are more than binding-deep.

In spite of being only a revision, the manual has been worked over thoroughly and thoughtfully, with many small changes to clarify explanations and improve the layout for easier reading and reference. The extent of this work is indicated by there being 28 more pages in the main body of this edition than in the first.

It is in the Appendices, however, where the most noticeable changes and additions have been made. Appendix A, the short summary of Level II, now has page references with every item to direct the user back into the body of the manual for more detail. Of similar usefulness is an extensive index, which it lacked previously. A highly legible table of every decimal number to 255 with its binary, octal and hex equivalent is provided. Much new information about control and character codes is offered, including a nice chart of the graphics characters.

For the modest price of $5.95 this manual is worth replacing that dog-eared old version - and if this provides a pretext for rereading it carefully, some surprising jewels of new knowledge might turn up.

Edward M Roberts

Software Review
Keyplus Integrated Utility Package
SJW Inc
$14.95 cassette
$19.95 diskette

Some time ago I saw a new product announcement for the Keyplus Integrated Utility Package. The announcement stated that Keyplus provided auto repeat, keyboard debounce, Basic shorthand, direct graphic entry, lowercase video, restoration of lost Basic programs, lowercase without shift, single key stroke user definable strings, and more. All this for only $14.95!

Frankly, I was sceptical. It seemed to me that many of these features were mutually exclusive. And let's face it, $14.95 for all that? No way!

Out of curiosity I called the distributor, SJW Inc., and discovered that Keyplus uses SHIFT-CLEAR to enter command mode. Once in command mode specific utilities may be enabled or disabled by typing a key.

They also told me that the program loads in just 20 seconds and that there is a disk version with additional features. I sent them a check for $19.95 and less than two weeks later received the disk version by first class mail.

Disk Keyplus not only contained all the features of the non-disk version, but also contained extremely powerful disk based utilities. For example, Disk Keyplus contains an auto input routine that allows a user defined string to be generated as if it were being typed in from the keyboard. This string may be automatically input at power up or by hitting just two keys. The possibilities are endless. On one of my TRSDOS disks I use this routine to load Basic, run a number crunching program, load from disk the present status of the calculations, and finally store the results back to disk. All this is done from power up without requiring further action on my part.

The direct graphic entry routine allows graphics to be typed directly from the keyboard. Basic programs containing super fast graphics can be written using this utility.

Basic shorthand allows Basic keywords such as GOSUB, NEXT, RETURN, etc., to be typed in one keystroke. This saves considerable time.

The restoration of lost Basic programs routine does just that. Don't ask how, but it works.

In fact, everything works! This is an incredible program; one I can strongly recommend. Keyplus and Disk Keyplus are available from SJW Inc, PO Box 438, Huntingdon Valley, PA 19006.

James Joachim

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.

Circle 67

* Radio Shack is a division of Tandy Corp and not affiliated with Unique Software

---

ebg &ASSOC

EMMANUEL B. GARCIA, JR.
& ASSOCIATES
203 N WABASH
CHICAGO, ILLINOIS 60601
PHONE (312) 782-9750

NOW IN ITS THIRD YEAR OF SPECIALIZING IN PRODUCTS RELATED TO THE TRS-80*

- Dealer of quality software and hardware—we sell **only** the best, proven items.
- Sponsor of CHICATRUG—monthly learning sessions for TRS-80 users.
- Publisher of "CHICATRUG News," monthly publication for TRS-80 users.

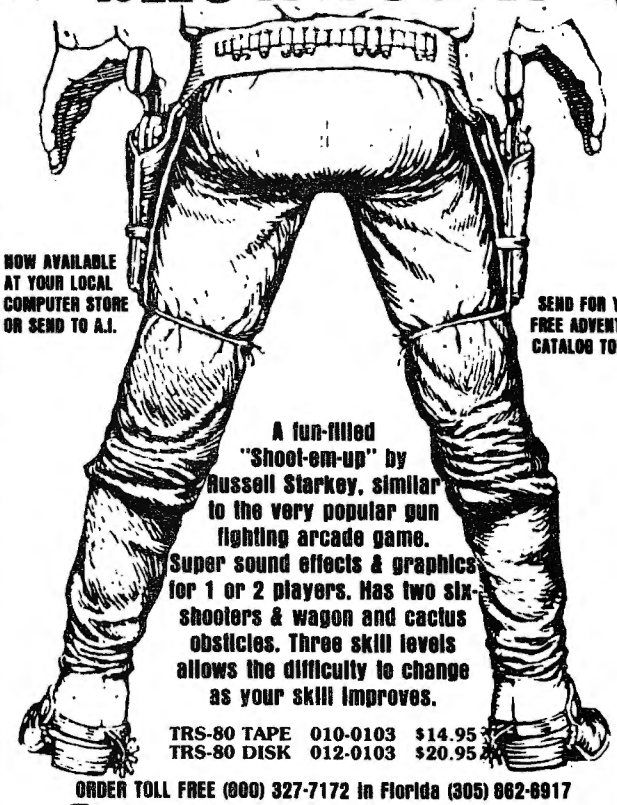
■ ■ Send for sample issue ■ ■

- Sponsor of the FORUM-80 of Chicago, a 24-hour computerized bulletin board.
- Author of "Pensionmaker," the revolutionary defined benefit pension system running on the TRS-80 Model I where other pension actuaries are using mainframes.
- Repair facilities (90-day warranty on new equipment and 30 days on used).

E.B.G. & Associates is NOT affiliated with Tandy Corporation or Radio Shack.

*TRS-80 is a trademark of Tandy Corporation. Circle 69

NEW **SHOWDOWN** NEW



NOW AVAILABLE AT YOUR LOCAL COMPUTER STORE OR SEND TO A.I.

SEND FOR YOUR FREE ADVENTURE CATALOG TODAY!

A fun-filled "Shoot-em-up" by Russell Starkey, similar to the very popular gun fighting arcade game. Super sound effects & graphics for 1 or 2 players. Has two six-shooters & wagon and cactus obsticles. Three skill levels allows the difficulty to change as your skill improves.

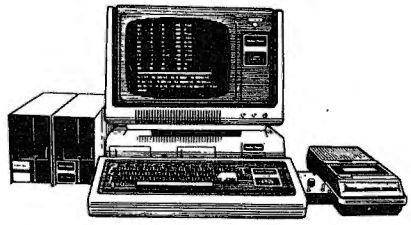
TRS-80 TAPE 010-0103 $14.95
TRS-80 DISK 012-0103 $20.95

ORDER TOLL FREE (800) 327-7172 in Florida (305) 862-8917

Adventure INTERNATIONAL P.O.Box 3435 – Longwood, FL 32750

Circle 70

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.

Circle 71

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_____

notes

PEEK and POKE for TRSDOS 2.0 (Model II)

The patches for implementing PEEK and POKE on the Model II were sent to us from Belgium. Many thanks to Daniel Lesenne, who sent us the information.

Be warned that **all eight patches *must be entered*** for this modification to work. They replace the OCT$ and NAME statements of Basic, so if you need these two commands, **do not** make the following changes!

1. Make a fresh backup copy of TRSDOS 2.0 or 2.0A.

2. Build a DO file (see pages 2/13-2/15 of the TRSDOS section of the Model II owner's manual) which reads as follows:

```
PATCH BASIC A=67F3,   F=AFCD8761,  C=CDDD3CD5
PATCH BASIC A=67F7,   F=C5CD7166,  C=E72CCDEA
PATCH BASIC A=67FB,   F=E741E753,  C=3CD112C9
PATCH BASIC A=67FF,   F=E3011E00,  C=CD5D447E
PATCH BASIC A=6803,   F=09444D,    C=C3FB3A
PATCH BASIC A=28FB,   F=CE414D,    D04F4B
PATCH BASIC A=2A05,   F=CF435424,  C=D045454B
PATCH BASIC A=5ADB,   F=CD8A4E,    C=C3FF67
```

When you are sure you have no errors in your file, execute the DO file. "PATCH IS DONE" should appear eight times. If you see **any** error message, make note of which patch caused it. You will have to enter that particular patch by hand from TRSDOS READY.

At this point, PEEK and POKE should be a permanent part of Basic on this disk.

PEEK and POKE will now have the same syntax as Model I/III:

PEEK(memory location)
POKE memory location, number

Number is any integer in the range 0-255. Memory location is any integer up 32767. Locations above 32767 must be converted using the formula:

-1*(65536-desired address)

Using PEEK and POKE will open up the world of string and line packing, saving variables between programs, and so forth.

Have you tried the short program in the color computer operation manual? If you are using your own color television, it may be well to try it. If yours is like ours, the red will be dark purple and the orange will be red. Run the program, it puts color bars on the screen and tells you which color should be where. Then adjust the television controls until the colors are right. You will be surprised at the difference in your displays.

Line Printer VI and Graphics

Line Printer VI users need to be aware that they have a medium-weight printer. Continuous printing for extended periods of time can cause the head to overheat and a fuse to blow. This is particularly true using **graphics!** It seems graphics characters put a heavier load on the head (they draw more current) when all pins are fired in rapid succession. Radio Shack is advising that after printing a full 15" print line of graphics, the printer needs to rest for 1 minute.

Color Computer Note

If you own a basic or extended basic color computer you may have noted that the statement: FORI=ATOB, will give you a syntax error. We have found that saying A+0 will get rid of the error. That's not nice though, and checking with Microsoft we found that a space is required after the A. This brings up an interesting point: the color basic is slightly different than what we have been used to in other TRS-80's. There are several places where a space is required to prevent syntax errors.

Point(X,Y) Function for Alphanumerics

Robert Hood, of Bremerton, Washington, sent us the following:

In the Model I Level II Basic Manual the POINT(X,Y) function is generally used for determining whether a specific graphics block is on or off. This function does not apply to alphanumeric characters. However, there is a simple method to make the same test for these characters and even to determine if a specific character is at a specific screen location. This is done by peeking the Video memory.

To determine the desired PEEK location add 15360 to the PRINT@ location. PEEK(location) will return the ASCII code for the character. A blank space will be 32. Since the SET(X,Y) function places the graphics code number in the video memory this method will also detect those codes as well.

One example of a use for the procedure is shown in the following subroutine:

```
4998 REM THIS SUBROUTINE LIMITS THE SCREEN DISPL
  AY TO 15 LINES OF TEXT WITHOUT THE USE OF LINE
  COUNTERS.
4999 REM IT MAY BE USEFUL IN THE DISPLAY OF A FILE
  WHERE THE NUMBER OF LINES TO BE DISPLAYED IS
  UNKNOWN.
5000 FOR SL=16526 TO 16270
5010 IF PEEK(SL) <> 32 THEN 5040
5020 NEXT SL
5030 RETURN
5040 SL=16271
5050 PRINT@977,"HIT ANY KEY TO CONTINUE
  DISPLAY";
5060 ST$=INKEY$ : IF ST$=""THEN 5060
5070 CLS : RETURN
```

Byte Miser Software

ALL PROGRAMS by JAMES F. WILLIAMS
Reviews in 80-U.S. and CREATIVE COMPUTING
LOOK! MODEL III SOFTWARE!

ASPTCH 3.0--> Loads behind Radio Shack's EDITOR ASSEMBLER 1.2 and adds many features while taking very little memory from the original text buffer. Reserve memory, dump object code directly to memory and execute, (NEW!) verify source dumps and use <BREAK> to exit unwanted (L)oads, return to BASIC with EDTASM/ASPTCH protected in high memory, combine EDTASM/ASPTCH for single load, and MORE! For MODEL I (L2) and MODEL III, 16K and up. Only $17.

ORGAN--> Play your TRS 80 keyboard in real time as if it were a 2 manual electronic organ. Display shows 2 manual organ console with TRS 80 keyboard characters overlayed. Asterisks appear on screen as keys are pressed. Two volume settings and 11 tone quality settings. Play single note melodies and chords! Record performances directly to cassette, or listen in real time by connecting aux cable to amp. For Model I, L2, 4K and up, and MODEL III, 16K and up--> only $12. Model I, Disk version--> $15.

DATORG--> Unique file keeping system designed especially for tape users (Disk and Stringy versions also available). Use for indexing magazine articles, budget info (check amt., date, no., expense code, description, etc.), mailing list, collections index (LP, coin, etc.) and much, much more! Extremely memory efficient with variable length fields and user definable delimiters (including special tab characters for left justified fields). Control program is in BASIC for easy user modification, and machine language module for high speed sort, search, write, verify, read, and merge. Advanced editing features and optional printer output. For Model I, L2, 16K and up Tape and Stringy Floppy> $20. Disk (32K and 48K)> $25.

CPYALL--> Make back-up copies of almost any L2 format tape (BASIC, SYSTEM, EDTASM source, data, and many programs with special loaders). Model I, L2, 4K and up, and MODEL III (with selectable baud rate), 16K and up. Can you believe only $7?

Send check or money order to BYTE MISER SOFTWARE, 729 W. Haven Blvd., Rocky Mt., NC 27801.

Circle 77

NEW EDUCATIONAL PROGRAMS for the TRS-80*

You can now order reading programs for

READING COMPREHENSION- ECONOMICS- PHYSICS- MATH-AUTO MECHANICS- HISTORY

2 FULL LENGTH PROGRAMS only $9.90 (one cassette)
16 PROGRAM COURSE only $77.00 (8 cassettes)

Order by checking the desired programs:

*Reading Comprehension	*Auto Mechanics	
☐ Cd 1 Cd 2 $9.90	☐ Ka 1 Ka 2 $9.90	☐ Physics 1 Physics 2 $9.90
☐ Cd 3 Cd 4 $9.90	☐ Ka 3 Ka 4 $9.90	☐ Economics 1 Economics 2 $9.90
		☐ History 9 History 10 $9.90

*12 more Cd's available *12 more Ka's available

☐ 2 Demo programs and Catalogue $9.90

Utilize with any Model I, Level I or Level II, 4K and 16K or any Model III. Displays lower case if you have it.

*TRS-80 IS A TM OF TANDY CORP.

I have enclosed a total of _____. $9.90 for each 2 program cassette ordered plus $.50 per cassette for shipping/handling to:

DORSETT Educational Systems, Inc.
P.O. Box 1226, Norman, Ok. 73070

Circle 78

Name _____

Address _____

City _____ State _____ Zip _____

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.

Circle 37

NEW EDITION

SFINKS 1.81

The only microcumputer chess program available which examines all legal moves! We challenge any microcomputer chess game to solve number 1000 from Reinfeld's 1001 Checkmates, much less beat SFINKS's time of 15 seconds. **Buy the best, SFINKS 1.81!**

TRS-80 32K Model I cassette or disk only $29.95

Original version of SFINKS is now available on Model III cassette only $29.95

WILLIAM A. FINK
1300 S.E. 3rd AVENUE
DEERFIELD BEACH, FL 33441

the superior chess game

Circle 79

BUSINESS USERS!

SAVE TIME & MONEY BY CONTROLLING YOUR LONG DISTANCE TELEPHONE COSTS

LONG DISTANCE ANALYZER will

- Sort your billed calls
- Identify each party called (from your known list) or print locations of unrecognized numbers
- Total the cost of calls to each number, to all numbers, and to all unrecognized numbers
- Analyze calls and tolls by area code, state, and WATS zone
- Print an alphabetical directory of recognized numbers
- Save verification time
- Create cost-consciousness
- Facilitate cost accounting and client billing
- Isolate personal calls and billing errors
- Reveal inefficient use patterns
- Help you perform WATS feasibility studies

Requires Model I, II, or III TRS-80*, Level II, 16K, and 1 disk. Printer is optional. No connection to phone system.

Model	Cassette	TRSDOS* disk
I	$95	$135
II	N/A	$155
III	$95	$135

Brochure on request.
Send model number and check or money order to:

Golden Braid Software
P. O. Box 2934
Sarasota, Florida 33578

*TRS-80 and TRSDOS are trademarks of Tandy Corporation.

Circle 80

TRS-80 LEVEL II

●**PLANET LOCATOR** - As the name implies, this program will compute the location and display of the positions of the planets. It also provides important statistical data. This is a valuable tool for the ASTRONOMY BUFF (16K) .....$ 9.95

●**LEAVE ONE** - This game places you against the computer in a match (taking turns and using game rules) to pick up objects and leave the last object for your opponent. The game itself can be played without a computer, but there is a trick to winning. (4K) .....$ 6.95

●**CHECK STORAGE AND YEAR END SUMMARY** - This two-program system handles the capture, sorting, summarizing and various other important functions needed to maintain you data. Graphics have been incoporated to simplify keying in the checks. The second program is the year end summary report. This is very useful at tax time or just as a expenditure analysis report. (16K) $11.95

●**AMORTIZATION** - This handy little program will calculate a six-column display as follows:
COL. 1 - Payment Number
COL. 2 - Monthly Payment Amount
COL. 3 - Portion Applied To Principal
COL. 4 - Portion Applied To Interest
COL. 5 - Remaining Principal Balance
COL. 6 - Principal Paid Back To Date
Totals at the end show total interest and total amount paid. (4K) .....$ 6.95

●**MESSAGE CODER-DECODER** - Have any written material you wish to keep private. Select one of 250 coding methods which will scramble and change your message so that only the computer can decode it back to its original form. (4K) .....$ 6.95

●**CONTRACT PROGRAMMING:** For as little as $20.00. Send a written summary of desired system. You will be contacted as to price.

HOW TO ORDER

Send Check or money order to **ARLINGTON MICRO-SOFT, INC.**
4800 Crestmont Court Arlington, TX 76017
Name _____
Street _____
City/State _____
Zip _____ Phone (___)_____
Texas Residents add 5% Sales Tax Allow 4 to 6 weeks for delivery

Circle 44

MODEL I or III SOFTWARE

intraSORT *machine code* becomes part of **YOUR BASIC PROGRAM.**

- CLOAD CSAVE LOAD SAVE **TAPE or DISK**
- **NO MACHINE LANGUAGE KNOWLEDGE REQUIRED**
- NO **SYSTEM or MEMORY SIZE** hassles
- **SORT** *any* type and combination of fields with each field ascending or descending
- **SORT** multi-dimension arrays
- **VERY FAST** and compact
- Complete USERS GUIDE
- Includes full feature mailing list/label program demonstrating all **SORT** features

••• COMPLETE PACKAGE ONLY $19.95
(requires Level II or Model III Basic, 16K)

BASIC ARCADE KIT *(Level I only)*

Hours of action fun for adults and children are provided by a single ready to run program written entirely in Level I BASIC. Designed for the novice programmer, twelve easy to use, fully documented subroutines and three demo games; TAG, PHASERS, and SKETCH make up the program. Subroutines provide: three keyboard commands which alter graphics motion without halting it; keyed input without disruption of display; two players with separate skill levels & scoring; obstacle deflection; random motion; and cassette storage of graphic displays. Write your own games or just ENJOY!

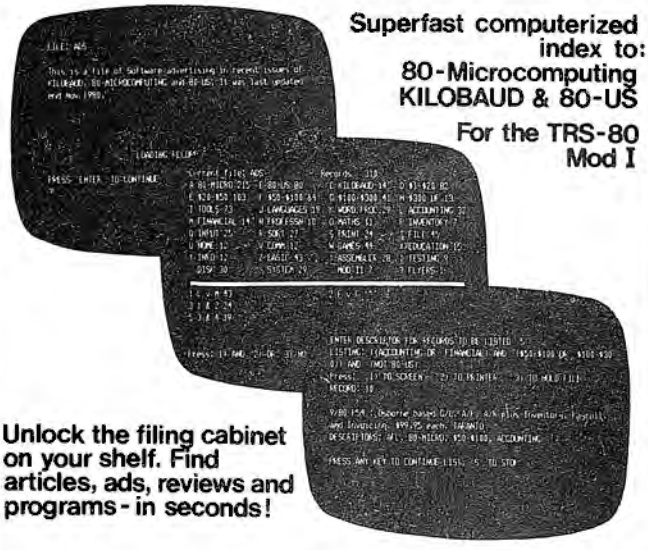
includes 20 page guide ONLY $9.95

Original Software by:

technique COMPUTER SOFTWARE
2720 S. Oak St. • Port Angeles, WA 98362

Circle 72

THE 80-INDEX



Superfast computerized index to: 80-Microcomputing KILOBAUD & 80-US

For the TRS-80 Mod I

Unlock the filing cabinet on your shelf. Find articles, ads, reviews and programs - in seconds!

Index & Programs & Manual US$29 CASH, VISA, OR MASTERCARD
Manual available separately US $8

Requires 48K TRS-80* Mod I with 2 Drives & TRSDOS

*TRS-80 is a registered trade mark of Radio Shack

HEXAGON SYSTEMS
P.O. Box 397 Stn. A
Vancouver, B.C. Canada V6C 2N2
Telephone (604) 682-7646

Circle 73

AT LAST!

Mass production prices for high quality software. Buy direct and save 50%. Also available for CPM and HDOS.

DATA BASE MANAGER Mod I & III $69, $149 (48K) Mod-II $199
Maintain a data base and produce reports, all without user programming. Define file parameters and report formats on-line. Key random access, fast multi-key sort, field arithmetics, audit log, label. No time-consuming overlays. 500 happy users in one year. Mod-II and 48K versions have over 50 enhancements, including 40 fields maximum. "IDM-M2 is great!" - 80-US

A/R Mod-I $69 Mod-II $149 Mod-III $69
Handles invoices, statements, aging, sales analysis, credit checking, forms input, and order entry. Unlike other accounts receivable programs, ours can be used by doctors, store managers, etc.

WORD PROCESSOR $49
Centers, justifies, indents, and numbers pages. Mod-I version features upper/lower case without hardware modification! File merge option available.

MAILING LIST Mod I & III $59, $79 (48K) Mod-II $99
The best! Compare and be selective. Includes forms input, 5-digit selection code, zip code extension, sort on any field, and multiple labels. Who else offers a report writer and merges with word processor?

INVENTORY Mod I & III $89, $109 (48K) Mod-II $149
Fast key random access. Reports include order info, performance summary, EOQ and user-specified reports. Many people have converted to our system! "Next to impossible to damage the file."

GL, A/R, A/P, PAYROLL Mod-II $129 each
Integrated accounting package. 100+ page manual. As opposed to Osborne's slow binary search and 64 column screen, we use fast ISAM and 80 columns. Dual disk and TRSDOS required.

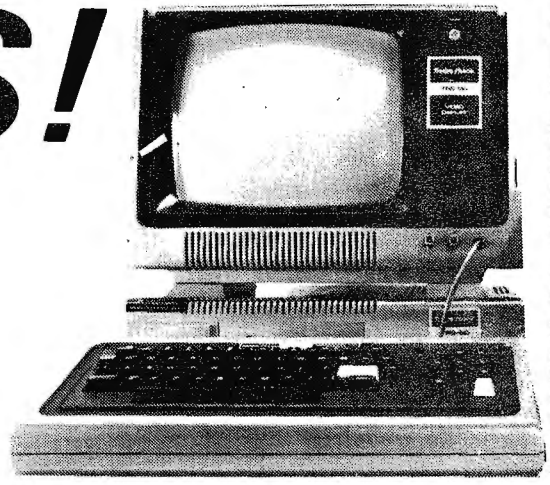
L216 $59
A cassette package of 10 business programs for Level II 16 K systems. Includes word processor and data base manager. Poker game $19

Most programs are on-line, interactive, random-access, bug-free, documented, and delivered on disks. Mod-I programs require 32K TRSDOS. We're #1 in business software—don't let our low price fool you! Ask for our free 20-page catalog if you're still not convinced. Compiled versions are available.

MICRO ARCHITECT, INC.
96 Dothan St., Arlington, MA 02174

Circle 76

LDOS!



- **The Ultimate operating system for the Model I TRS-80**
- **Upward compatible with TRSDOS**
- **Mix drives and densities, 5", 8" or hard disks, all at the same time!**
- **Complete with a superb 250 page manual and continuing support.**
- **Toll-Free hot line for assistance.**
- **Available NOW!**

TRS-80 Model I LDOS, with documentation: $139.00

from

80-U.S. Software
3838 South Warner Street
Tacoma, Washington 98409
(206) 475-2219

MasterCard/Visa welcome

TRS-80 is a trademark of the Tandy Corp

Circle 59

UNCLASSIFIED ADS

$2.50 per half inch and $2.50 each additional half inch per insertion. Ads must be typewritten or printed clearly, and must be accompanied with payment in advance. No telephone orders of "bill-me's" will be accepted. Unclassified ads are always set in the type on this page, with the first line in bold caps. Use the ads on this page as a guide to figure your size and payment. Send to: 80-U.S. Journal, 3838 South Warner Street, Tacoma, Washington 98409

TRS-80 EQUIPMENT! DISCOUNTED TO YOU Centronics 737(same as Line Printer IV) $775. Okidata Microline 80 40/80/132 column plus graphics. Handles single sheet/roll/fanfold paper $504 (cable for either $25). 16K RAM set/8 $32, Mini Flippy Kits (punch and use both sides of your diskettes) $7.50. 100 =/- programs on disks $39. Novation Modems, acoustic $145, direct $159. Diskettes X10 BASF $27, 3M $32, Maxell $36, Dysan $42. All equipment factory warranted. Write for the complete list. Specify Model I or II. Richard Smith, 3078 Baronscourt Way, San Jose, CA 95132 (408) 258-5202(071)

WANTED: "SCRABBLE TYPE"CROSS-word game for the TRS-80, any model. Write to Margaret, PO Box 7112, Tacoma, WA 98407 (071)

DEALERS WANTED. GREAT DEMAND FOR school soft-ware with educators, students & families. AA Educational Services, 7211 South 180th St., Kent, WA 98031 (071)

MAKE YOUR LIST ILLEGIBLE CONCEAL your access codes. $5.00 cash. Make the bums look like bums. AA Educational Services, 7211 South 180th, Kent, WA 98031 (071)

CAN YOU DEMONSTRATE L2 16K TO schools, students, families in your area? Write for dealer discount schedule. AA Educational Services, 7211 S 180th, Kent, WA 98031 (071)

---

5 FOR 1 PROGRAM EXCHANGE, SEND 1unreserved program that runs plus $5.00. Receive 5 programs in exchange. 5 for 1 Program Exchange. 25824 124th S E, Kent, WA 98031 (071)

FOR SALE - THE MICROCONNECTION almost new direct connect modem for the TRS-80 Model I complete with all instructions and programs. $150.00. Call Roger evenings (201) 886-2447 (051)

GRADEBOOK/ROLLBOOK- CONVENIENT and useful for the teacher Up to 44 scores in 3 catagories for each of 44 students. No real limit on number of classes. Instant progress reports. Clear & easy displays of scores Teacher-passworded. Requires Mod I or III, disk. Printer supported. $30 for disk or large stamped envelope for instructions Hal Broenkow, 8511 226 SW., Edmonds, WA 98020

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)

WORD PROCESSOR, PERSONAL FORM letter printer. Adds addresses, inserts 1st name from address into letter body. Global word search, replace. Caps-lock, overstrike, move words between lines. Delete, replace, insert, move, center, right-justify lines. Basic-like line edit, with A,C,D,H,I,K,L,S,X commands. T/D save. System? (16K, WP only) $10 on tape, $12 on disk. D D Hinrichs, 2116 S E 377th, Washougal, WA 98671 (051)

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)

GL, A/R, A/P, PAYROLL FOR TRSDOS 1.2 Not Osborn's. Fast ISAM, double entry accounting, full screen and integrated. MAI 96 Dothan St., Arlington, MA 02174 (091)

ROMAN CONQUEST FOR 16K LEVEL II Caesar orders you to seize cities, sell slaves, draft centurions and colonize. Can you cope with fierce barbarians, plagues, rebellions and the will of Caesar? $9.95 (cassette) Anthony Targonski, 106 Fifth St., Cambridge, MA 02141

PROFORMA CASH-BUDGET PROGRAM: Computes a forecasted cash-flow shecdule for up to 12 periods in advance. LEASE-PURCHASE program: Compares lease and purchase costs, taking into account investment tax credit and income tax, and computing relative interest rates. PROCUREMENT PROGRAM: calculates the optimal amount to purchase of a commodity in each of next subsequent periods, in times of varying prices over time. COLLEGE ENROLLMENT PROJECTION PROGRAM: Forecasts Community College enrollment in advance based on the population of the area from which students are drawn and the enrollment for the preceding seven years. Package of 13 business analysis programs including ratio analysis, economic order quantity, leverage, cost of capital, IRR and NPV, depreciation schedule, capital budgeting, risk-adjusted discount rate, etc. For the TRS-80 Model I or III. Output on video or line printer. Requires 2 disk drives and 32K capacity. For brochure write to Management Systems Software, 5200 Brittany Drive #1006, St Petersburg, FL 33715 (813) 864-4347 (051)

DISK USERS: DOUBLE YOUR MEMORY space. We have developed a method of using the flip-side of your 5¼" or 8" diskettes. For $5. receive full instructions. Microcomputer Services, PO Box 263, Clinton, MD 20735(051)

CHEAPTALK FOR THE TRS-80 - SOFTWARE package lets your TRS-80 16K Level II reproduce speech through any small audio amplifier connected to the cassette plug. Digitize your own words with simple hardware mod. $19.95 ppd or send SASE for more details to Alan Saville, PO Box 5190, San Diego, CA 92105 (091)

WIN WITH OUR HANDICAPPING AIDS - Designed exclusively for use with TRS-80, we believe our "Greyhound" and "Harness Racing" software handicapping aids are the most advanced/useful such systems available anywhere. Years of research have gone into the development of the algorithms used. Designed for ease of operation while ensuring a most comprehensive approach. Used correctly, a greater return on your track investments will be yours. Requires minimum 16K Level II Just $24.95 (per system) for cassette for $28.95 for 5¼" floppy disk (add $1 shipping; CA residents add 6½% sales tax); save $1 with purchase of both systems; dealer discounts available. Send check or MO to WelMur of California Inc., PO Box M-11C, Fremont, CA 94537 **?Soon availabel for the new hand held computers. (051)

TRS-80 LEVEL II/III PROGRAMS ON cassette with listing and documentation original programs, two computer copies each, money back guarantee. Subroutine structure with remarks makes modification easy. Discount $1 per pgm on 2nd or more pgms if separate tapes not required. Send SASE for more infor on a particular program or send cost via check or MO to: Practical Programs, 1104 Aspen Drive, Toms River, NJ 08753 - **GW01P** - Miss America Computer Notepad (4K) $4. A handy program to use your TRS-80 as a scratch pad Stores up to 10 lines of 254 characters each for 10 codes. Use codes to add new lines of data or retrieve all stored data. **GW02PB** - Tape Address File Handler (8K) $7. Uses two cassettes to store and update an unlimited number (limited by size of cassette/s) of addresses. Four modes of operation - initiate, update, add & retrieve any combination of modes possible in any tape pass. Fields include first and last name, address, city, state, zip, phone, retrieval code and remarks Requires El, printer and second recorder. **GW03PB** - Mail Order Gener (8K) $8 A really neat easy running program for mail ordering. Uses the TRS-80 to generate and print a catalog order. Then manipulates it per the following menu: Add, change, delete, print, change addressee, totals, review order, save to tape, load from tape, check space, sort, form a new order or end. Optional sub accounting allows allocation of line item costs with subtotals per account Optional constant Cat. No. prefix/suffix makes order entry easier. for 32/40/80 column printer. (051)

CALCUL80 - USE YOUR TRS-80 AS A FULL function calculator with versatile memory. Hook up printer and printout is automatic Screen display always shows memory contents. As easy to use as standard calculator, but with bonus of alpha-numeric entry and output. Formatted or floating decimal. User oriented, with full, clear documentation Perfect for taxes, budgets, checkbooks. Level II, 4K or more. Cassette $12 95 from Microlex, 19224 Castlebay Lane, Northridge, CA 91326. California residents add 6% tax. (051)

EXATRON STRINGY FLOPPY SOFTWARE and patches: 1) ESF version of ENHBAS, enhances Basic from Cornsoft Group - complete package $49 95, supplied on wafer 2) Basic Toolkit (c) Bottom Shelf, patch only. 3) Microchess 1.5 (c) Peter Jennings, patch only. 4) Pinball (c) Acorn Software, patch only. 5) Basketball (c) Acorn Software, patch only. New patches will be added on a regular basis. Call or write for patch prices and additions to this list. Multi Media Systems, PO Box 41084, Indianapolis, IN 46241 (317) 839-6520 (091)

ATTACK OF THE YETI FOR 16K LEVEL II You're a scientist leading an expedition in search of the Yeti, otherwise known as the "abominable snowman" In the Himalayas, aided by a small band of native Sherpa guides and your mountaineers, you search for the beast, financing your expedition along the way capturing wild animals. Yet, there are avalanches, attacks by the animals and other disasters as well as a certain creature who has a nasty habit of carrying off your men. Search and set your traps carefully! $9.95 (cassette). Anthony Targonski, 106 Fifth St., Cambridge, MA 02141 (051)

L.U.S.T. (A QUASI—EROTIC EXPERIENCE!) From the people who brought you Sin Palace. Send $3. for program on cassette, catalog and Genuine Casino Souvenir. Paul's Electric Computer, PO Box 42831U, Las Vegas, NV 89104 (111)

ASTEROID RUNNER - TRS-80 ACTION game. Maneuver your ship through the array of creeping asteroids, and success. Fully dock your ship on the opposite side of the storm. With sound effects! 16K L2, $4.95 The Electro-Gamesmen's Palace, 108 Carousel Dr., Delmont, PA 15626 (412) 468-8996 (051)

LINE PRINTER II INTERFACE TO KEYBOARD for sale. RS #26-1416. Used 6 months list $59, sale $35 first MO or cert check takes. Paul C Richwine, 5516 Central Dr, Monroe, MI 48161 (051)

NEW ROCHESTER DATA TYPEWRITER interface. Complete with extra parts and paperwork. Special Scripsit included. Sets on any electric typewriter keyboard. No expansion interface required. Many adjustable features. Works great. $525 value for $450 or possible best offer. John Berkey, 1637 Cawdor Dr, Fayetteville, NC 28304 (919) 424-4072 (051)

CASSETTE DUPLICATING - LARGE OR small quantities, quick turnaround, low rates, top quality screw cassettes. Write for rate sheet. EDU-WARE EAST, PO Box 336, Maynard, MA 01754 (XXX)

B-17 TAPE OPERATING SYSTEM. LETS YOU save/load Basic or System programs four times faster. Save/load data arrays 400 times faster! Backup your system tapes with B-17. Complete with operating instructions and 30 day unconditional money back guarantee. $24.95 ppd 1st Class. Also ask for free Resource Brochure containing eye-popping value. We'll try to match any price even if we have to raise ours to do it. ABS Suppliers, PO Box 8297, Ann Arbor, MI 48107 (313) 971-1404 (091)

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 (091)

NEW QUARTERLY NEWSLETTER FOR VisiCalc® users. Disks too. $7.50. Micro-Calc Business Users Group, PO Box 12039, Salem, OR 97309 (012)

EXTENSION CABLES FOR PRINTERS - Centronics or RS 232C(EAI) Interface. $20. +$1 per foot. Please indicate length required (8 ft max for Centronics type) and send your check or MO to: Data Devices International, 2504 Clearwood Dr., Arlington, TX 76014, or call (817) 261-4253 if you have questions. (051)

GREEN SCREENS COST TOO MUCH?? 1/8" green Plexiglas (same as most others) cut to fit TRS-80 Mod I...$5.95, custom sizes up to 12 X 12"...$7.95, add $1 shipping. Satisfaction guaranteed. Data Systems, Box 99, Fern Park, FL 32730 (051)

POCKET COMPUTER (TRS-80) SOFTWARE.. Appointment Calendar: Schedule & Maintain a week of appts. (6 per day). Alarm warns of each appt. Review, add, modify, cancel, save appts with ease. $9.95 - Electronic Checkbook: Maintains checking acct. balance, checks & deposits (including descriptions). Balances acct. with bank statement. $5.95 - Blackjack Helper: Beat the dealer with this aid. Tracks cards of all players and gives you the odds of the next card "busting" your hand. Handles multiple decks too. Lets you know which and how many cards remain in the deck. $7.95 - Additional software available (Mod I & III also). Send SASE for catalogue, Blanton Software Service, 4522 Brair Forrest, San Antonio, TX 78217 (512) 657-0766 (051)

MODEL I & III SOFTWARE...WORD processor: Design, develop, review, edit, justify, move lines, print, save, ets. any type personal or business document. Can also print expanded characters. Powerful, versatile, yet simple to use word processor does it all. (16K tape). $19.95 - Personal/Business Mailing List: List maintained in memory as part of program for fast & simple access and modification. Eliminates need for separate and cumbersome data tape. Prints labels or envelopes in normal or expanded characters (return address too). Assign up to 80 different codes to names. Review/print by name, code or entire list. $6.95 - More programs available. Send SASE for catalogue. Blanton Software Service, 4522 Brair Forrest, San Antonio, TX 78217 (512) 657-0766 (051)

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 (28¢) to MAI, 96 Dothan St., Arlington, MA 02174 (051)

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)

MAKE MONEY WITH YOUR 16K computer!! Computer assisted introductions is the latest in money-making ideas that anyone with a little initiative can easily turn into a full-time business. Send SASE for more info, or get starter package of two programs and all details for a mere $40. (+$5 if you prefer disk)! MAX ENTERPRISES, PO Box 19386, San Diego, CA 92119 (051)

TIME FOR THE SHERIFF FOR 16K Level II. You've been elected temporary sheriff of Honesty, New Mexico, the wildest town this side of the Rio Grande and you must do anything to keep the peace. Patrol the bank, general store, hotel, ranch and sheriff's office but watch out! Bank robbers, gamblers, gunfighters and renegades have other plans! When the Marshal returns he'll judge your record (point score) and decide if you should be reelected, fired or hanged from the nearest tree. Only $9.95 (cassette) Anthony Targonski, 106 Fifth St., Cambridge, MA 02141 (051)

PERSONALIZE YOUR SCRIPSIT™!! PERSCRIP lets you design your OWN word processor. Your name or message up to 35 chars will be displayed upon initializing your word processor. Defaults (all margins, spacing, video width, etc.) will be yours, not arbitrary to be reset constantly. Change your cursor to any character! Rid yourself of "loud", eye-straining graphics (cursor, bottom demarcation line, etc.). Underline (most printers)!! Finally, the most eye-saving SCRIPSIT™ enhancement ever: our famous non-cursor. Try it, and you will never use a conventional cursor again!! Package includes manual, cassette, complete map of graphics characters. Disk SCRIPSIT™ req. Specify SCRIPSIT/LC or /UC. PERSCRIP will be nationally advertised at $15.95. Mention 80-U.S., and your cost is $12.95. Add $1.00 post/handling. Send to: Dave Rose, Rohan Concepts, PO Box 20873, Atlanta, GA 30320 (111)

Alternate Directory
Volume IV Number 3 May/June 1981

IN THIS ISSUE

Features

Reviews

Departments

BACK ISSUE AVAILABILITY

The following back issues of the **80-U.S. Journal** are still available:

May/Jun 79	May/Jun 80
Jul/Aug 79	Nov/Dec 80
Nov/Dec 79	Jan/Feb 81
Jan/Feb 80	Mar/Apr 81

Price per copy is $4.00 ppd. Please allow approximately 3 weeks for delivery.

ADVERTISER & READER SERVICE INDEX

Vista

Model V1000

. . . Looking Out For You.

Eight Inch Floppy Disk Drive Subsystem Model V1000

The V1000, Vista's sophisticated new disk drive subsystem, sets new standards for ease of access and use. Its innovative design permits disk drives to be mounted or removed quickly and easily for system reconfiguration or servicing.

Features:

- Deluxe chassis with internal slide allows easy access.
- Storage capacity from 250 kilobytes to 2.5 megabytes.
- Desk or rack mountable.
- Accomodates both single-sided and double-sided drives.
- Industrial quality cabinet with die cast front bezel.
- Drives pull out for easy service and maintenance.

Prices:

Cabinet with (2) single sided drives w/power supply	$1595.00
Cabinet with (2) double sided drives w/power supply	$2295.00
Cabinet (only)	$ 395.00

Model V200

Vista's Line of High Performance, High Reliability Products also Includes these Advanced Components

Daisy Wheel Printer Model V300

Features:

- 96-character proportional, bi-directional printing
- Interface - Parallel or RS232-C option

Prices:

V300-25cps	$1895.00
V300-45cps	$2195.00

Minifloppy Disk System, Model V200

Features:

- Storage capacity from 200K bytes to 1.2 megabytes
- Compatible with industry standard S100 mainframes.
- System software - Vista CP/M™ VOS Disk Operating System and Basic - E compiler.

Prices:

Starting as low as $695.00
V200-Exidy version $1199.00

Model V300

Vista Computer Company 1317 E. Edinger Avenue • Santa Ana, CA. 92705 • (714) 953-0523

Dealer Inquiries Call Toll Free (800) 854-8017

™ is a registered trademark of Digital Research

Pump Up Your TRS-80 with the ES/F Mass Storage System

▲ Actual Size

Actual Thickness ▼

THESE FACTS SPEAK FOR THEMSELVES!

	CASSETTE	ES/F	MINI-DISK
SPEED (Seconds to load "Blackjack")	56	6 (5' wafer)	6½
CAPACITY (thousands of bytes)	38 (C-20)	64 (75' wafer)	59 (TRSDOS)
RELIABILITY (Designed for digital data?)	NO	YES	YES
SYSTEM COST (First unit plus interface)	$60	$250	$800
MEDIA COST (in quantities of ten)	$3.10 cassette	$3.00 wafer	$3.20 disk

Let's face it. Cassette players were not designed to store digital data and programs. That's why we designed a digital storage system using a continuous tape loop: the Exatron Stringy/Floppy (ES/F) and the Wafer. There's no expensive interface to buy—the ES/F comes ready to pump up your TRS-80.*

Once your TRS-80* is pumped up by our ES/F . . . you won't want to deflate it. We're so sure, that we offer an unconditional 30-day money-back guarantee and a one-year limited warranty. Over 2,000 TRS-80* owners have met the wafer . . . why don't you?

TRS-80*

EXATRON'S STRINGY/FLOPPY...

SPEED, CAPACITY AND RELIABILITY FOR ONLY $249.50

CALL
OUR HOTLINE
(800)-538-8559

IN CALIFORNIA,
CALL (408)-737-7111

exatron, inc.
181 Commercial Street
Sunnyvale, Calif. 94086

*TRS-80 is a registered trademark of Tandy Corp.