Timekeeping For The VIC-20

The Resource Magazine For CBM/PET • Apple • Atari • OSI

Creating A Simple Word Processor

COMPUTE!

$2.50
February, 1982
Issue 20
Vol. 4, No. 2
63379

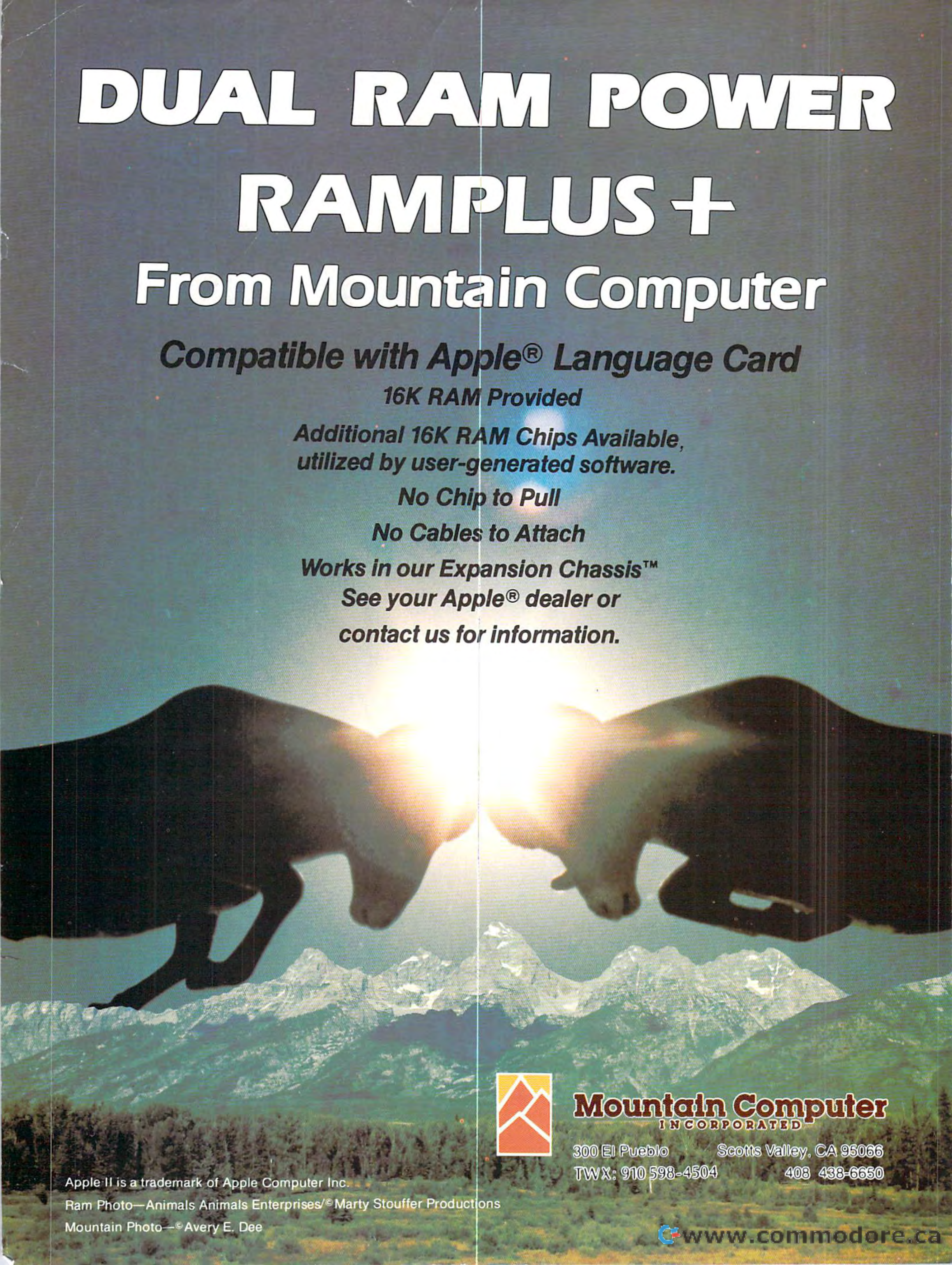
The Journal For Progressive Computing™

Programming Your Home Insurance Inventory

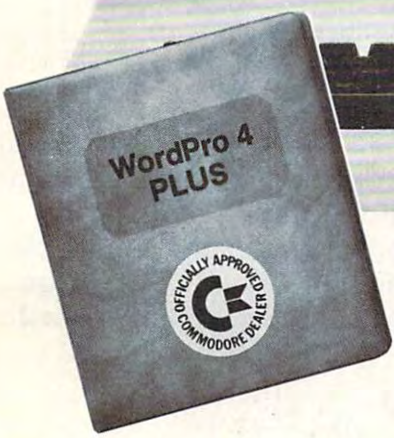
Named GOSUB With Variable Passing: Apple

Atari Player-Missile Graphics Made Easy

A User Defined Character Editor For PET/CBM


DUAL RAM POWER
RAMPLUS+
From Mountain Computer
Compatible with Apple® Language Card
16K RAM Provided
Additional 16K RAM Chips Available, utilized by user-generated software.
No Chip to Pull
No Cables to Attach
Works in our Expansion Chassis™
See your Apple® dealer or contact us for information.
Mountain Computer
INCORPORATED
300 El Pueblo
Scotts Valley, CA 95066
TWX: 910 598-4504
408 438-6650
Apple II is a trademark of Apple Computer Inc.
Ram Photo—Animals Animals Enterprises/© Marty Stouffer Productions
Mountain Photo—© Avery E. Dee
www.commodore.ca

IF YOU'RE WAITING FOR THE PRICE OF WORD PROCESSORS TO FALL WITHIN REASON,



Everyone expected it would happen sooner or later . . . with WordPro PLUS™ it already has! Now all the marvelous benefits of expensive and advanced word processing systems are available on Commodore computers, America's largest selling computer line. WordPro PLUS, when combined with the new 80 column CBM 8032, creates a word processing system comparable to virtually any other top quality word processor available—but at savings of thousands of dollars!

New, low cost computer technology is now available at a fraction of what you would expect to pay. This technology allowed Commodore to introduce the new and revolutionary CBM 8032 Computer.

WordPro PLUS turns this new CBM 8032 Computer into a sophisticated, time saving word processing tool. With WordPro PLUS, documents are displayed on the computer's screen. Editing and last minute revisions are simple and easy. No more lengthy re-typing sessions. Letters and documents are easily re-called from memory storage for editing or printing with final drafts printed perfectly at over five hundred words per minute!

Our nationwide team of professional dealers will show you how your office will benefit by using WordPro PLUS. At a price far less than you realize.

Invest in your office's future . . .
Invest in WordPro PLUS . . .
Call us today for the name of the WordPro PLUS dealer nearest you.

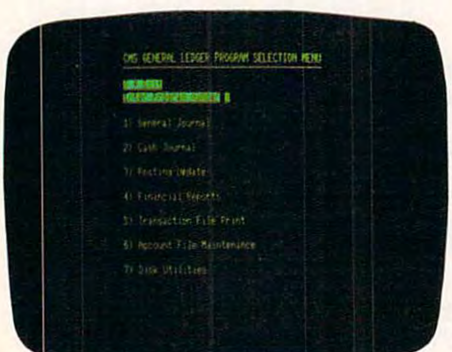
Professional Software Inc.
166 Crescent Road
Needham, MA 02194
(617) 444-5224
TELEX: 95 1579

TM WordPro is a Registered Trademark of Professional Software, Inc. WordPro was written by Steve Punter.

CMS Softwa
General Accou

For the Commodore 8000

Now there is an interactive General Accounting System designed especially for the first time user. The user is led through each function by highlighted prompts explaining the required input at each point. And all input requests are prompted with complete verification. Plus, a step-by-step instruction manual guides the user through the program. The user prompts and the detailed manual make it virtually impossible for the first time user to get lost or to accidentally crash.

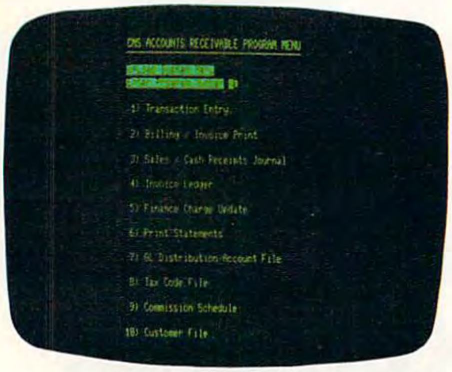


General Ledger

Cash Disbursements and Cash Receipts Journal as well as a General Journal. Maintains account balances for Present Month, Quarter to Date and Year to Date. Accepts postings from Accounts Receivable, Accounts Payable, Payroll and other programs. Also provides user customized financial statements. Up to 1000 accounts and up to 99 departments.

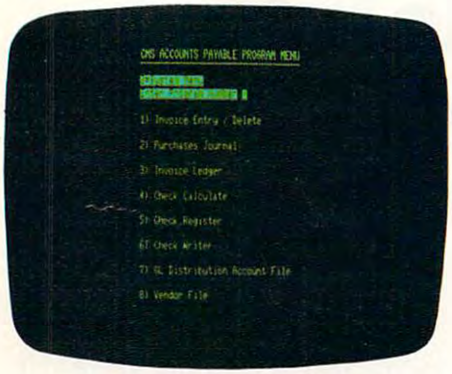
re Systems
nting System II

Series Computer System



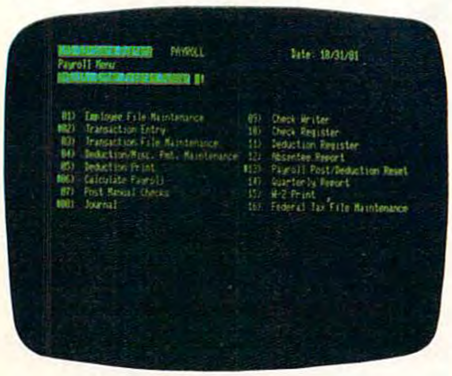
Accounts Receivable

Invoices and Monthly Statements as well as Credit and Debit Memos. Accomodates full or partial payments. User sets finance charge rate and period. User also sets invoice aging reports and aging breaks. Invoices may be distributed among nine different General Ledger accounts with automatic updating to the General Ledger. During invoice data entry the invoice is displayed on the screen and typed exactly as if in a typewriter. File capacity is 2000 minus number of customers multiplied by 1.4. Five hundred customers allows room for 2100 invoices.



Accounts Payable

Accounts Payable checks with full voucher detail. Detailed check register. Automatic application of Credit Memos. User sets invoice aging reports and aging breaks. Invoices may be distributed among nine different General Ledger accounts with automatic updating to the General Ledger. File capacity is 2000 minus the number of vendors multiplied by two.



Payroll

Payroll checks with full deductions and pay detail. Pays regular, overtime, holiday and piece work hours. Accommodates Weekly, Bi-Weekly, Semi-Monthly and Monthly employees. Up to eight miscellaneous deductions or payments per employee. Prints Payroll Journal, Payroll Check Register and an Absentee Report as well as 941 information and W2 forms. Maintains Monthly, Quarterly and Yearly totals for up to 350 employees. Automatic General Ledger updating.

See your nearest Commodore dealer for a demonstration.

CMS Software Systems

2204 Camp David Mesquite, TX. 75149 214 285-3581

TASC™. The Applesoft* Compiler. It turns your Apple into a power tool.

Step up to speed. TASC, the Applesoft Compiler, converts a standard Applesoft BASIC program into super-fast machine code. By increasing program execution speed up to 20 times, Microsoft gives you a power tool for Applesoft BASIC programming.

Highest capacity available. TASC will compile and run larger programs than any other Applesoft Compiler. As a disk-based system, it doesn't require the simultaneous presence of compiler and program in memory. The memory you save allows you to compile significantly bigger programs.

Power without bulk. Code expansion of up to 100% severely restricts other compilers. TASC's special code compression schemes typically limit code expansion to only 25%. You'll really appreciate that with complex programs or programs that utilize Apple's hi-res graphic pages.

More BASIC power. TASC's powerful new commands increase Applesoft BASIC programming capability. Chain with COMMON allows compiled programs to share variables, so a main menu supports several programs in a single runtime environment.

TASC's True Integer Arithmetic and Integer FOR...NEXT capabilities maximize the execution speed of compiled programs. TASC's near total compatibility with Applesoft speeds compilation of existing programs with little or no modification.

What about mistakes? You perfect your programs interactively with Applesoft. If something does slip by, TASC recovers from errors discovered in compilation and traps all runtime errors. It even permits graceful interruptions during compilation.

See for yourself. Ask for a demonstration of TASC at your Microsoft dealer. Discover the software package that turns your Apple into a power tool.

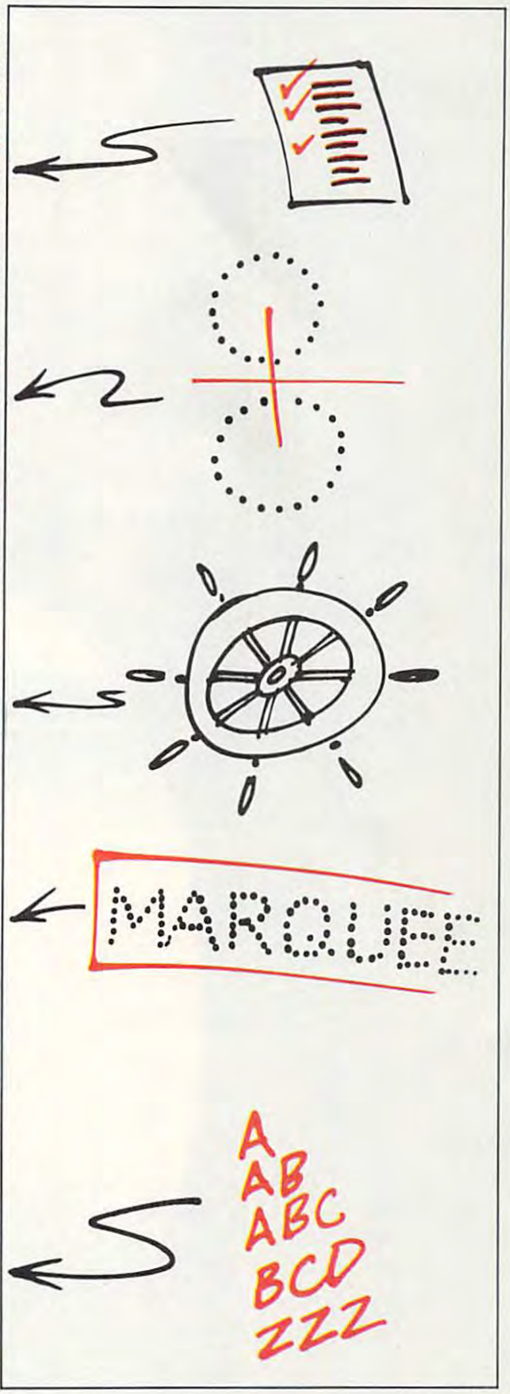
MICROSOFT CONSUMER PRODUCTS

A Division of Microsoft Inc.
10700 Northup Way • Bellevue, WA 98004

*Applesoft is a trademark of Apple Computer, Inc.

Table of Contents

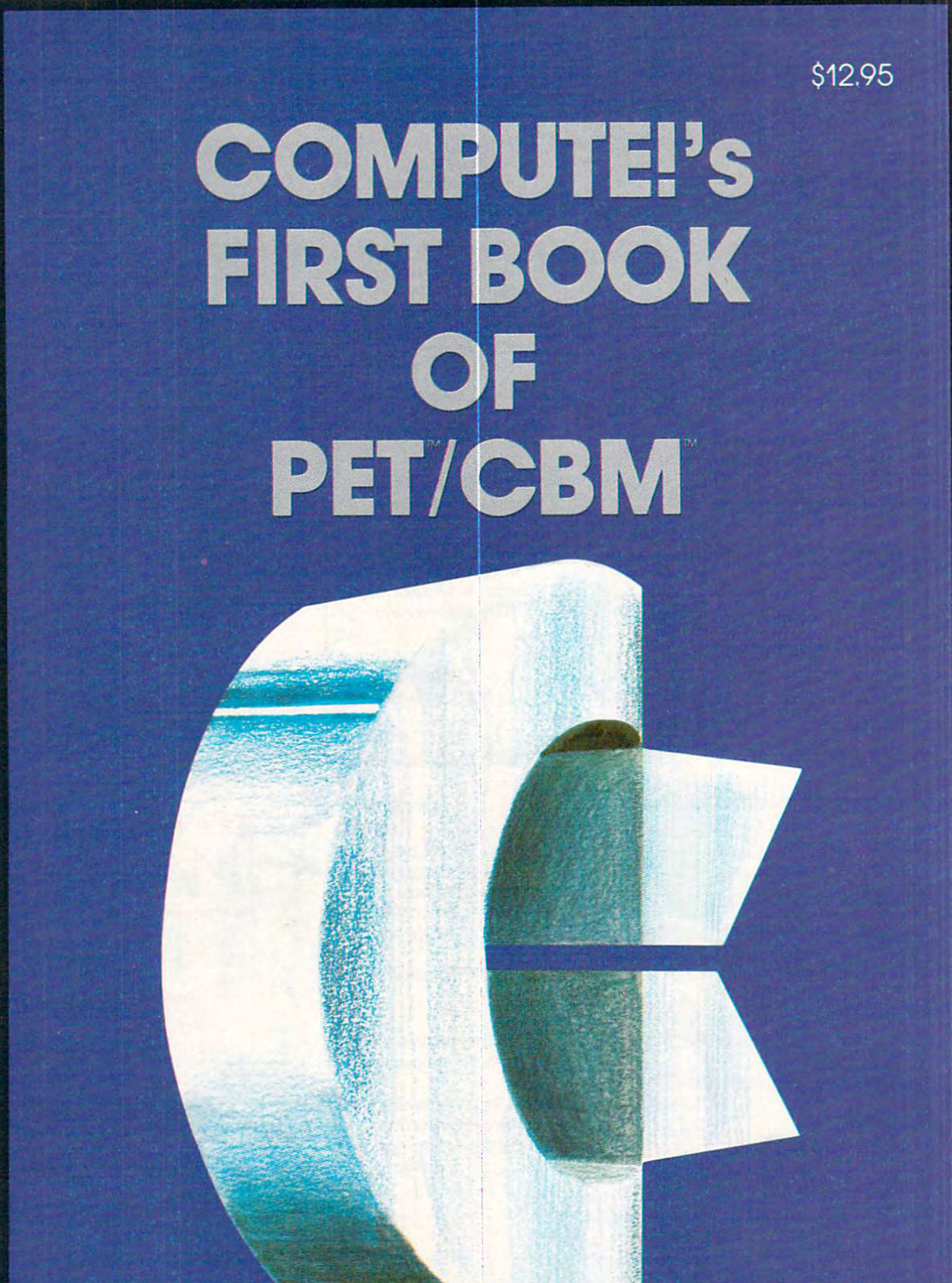
February 1982, Vol. 4, No. 2

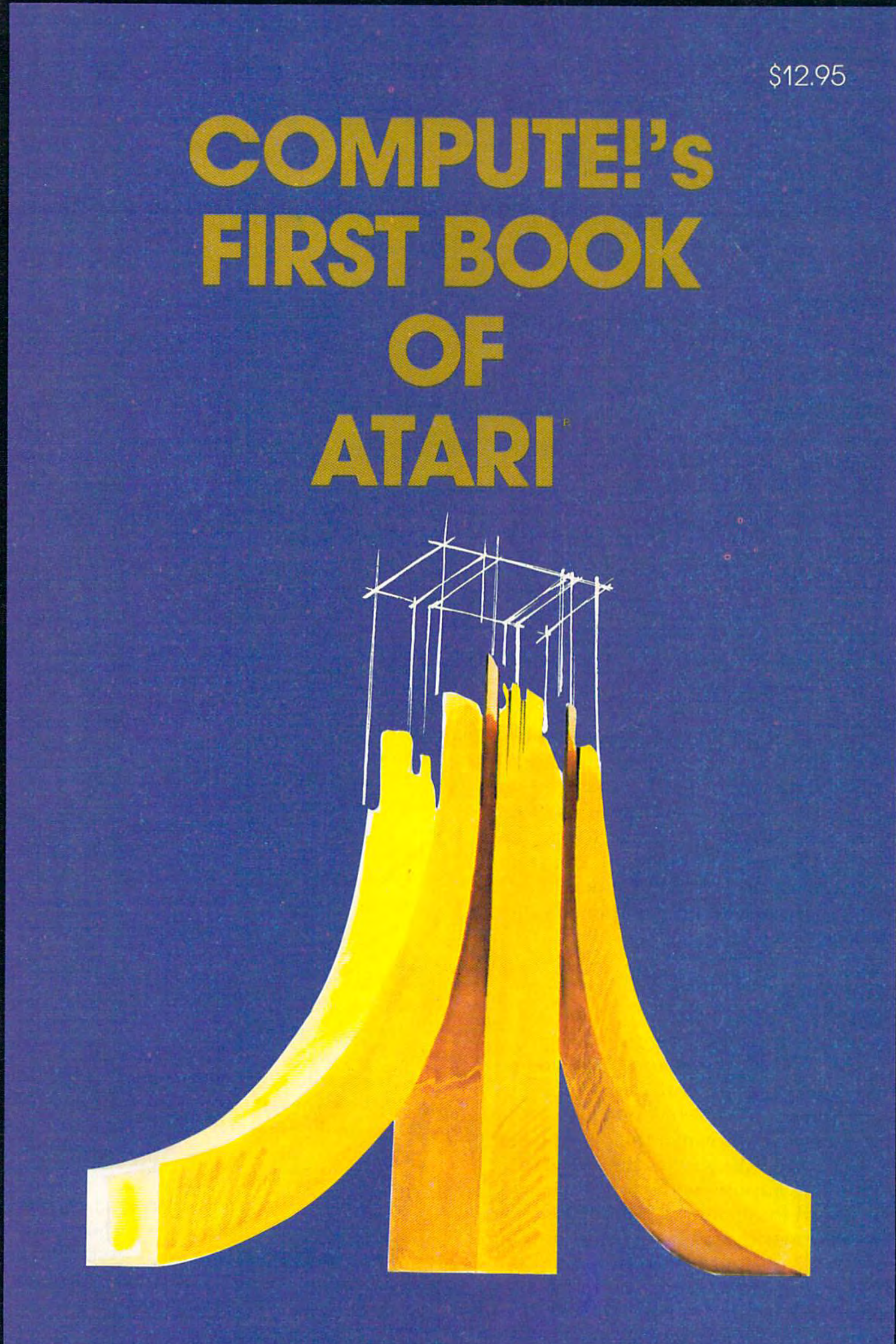


COMPUTE! The Journal for Progressive Computing (USPS: 537250) is published 12 times each year by Small System Services, Inc., P.O. Box 5406, Greensboro, NC 27403 USA. Phone: (919) 275-9809. Editorial Offices are located at 625 Fulton Street, Greensboro, NC 27403.

Domestic Subscriptions: 12 issues, $20.00. Send subscription orders or change of address (P.O. Form 3579) to Circulation Dept., **COMPUTE!** Magazine, 515 Abbott Drive, Broomall, PA 19008. Controlled circulation postage paid at Greensboro, NC 27403 and additional mailing offices. ISSN 0194-357X.

A Small System Services, Inc. Publication
COMPUTE!


$12.95
COMPUTE!'s
FIRST BOOK
OF
PET/CBM



$12.95
COMPUTE!'s
FIRST BOOK
OF
ATARI

The Editor's notes...

Robert C. Lock
Publisher/Editor

A Major Format Change

Now that we've completed the merger of *Home and Educational COMPUTING!* and *Recreational Computing* into **COMPUTE!**, we're concentrating on revamping the organization of the magazine to better serve you. Beginning in March, **COMPUTE!** will have two distinct sections rather than the six it has now. The first section will be called "Home and Educational COMPUTING!," containing applications, tutorials, columns, and reviews. The second section of the magazine will become "The Journal," carrying a mix of articles for intermediate and advanced users.

We'll continue "New Products," *and* continue to provide the same excellent resource and applications articles. As we move into the new year with continuing explosive growth, we're sure you "old timers" will find the new format easier to use, and you beginners to the world of personal computing will find it much more convenient. Remember, this starts in March and, as always, we'd appreciate your feedback and comments.

The Hardware Wars: Late-breaking and Major News

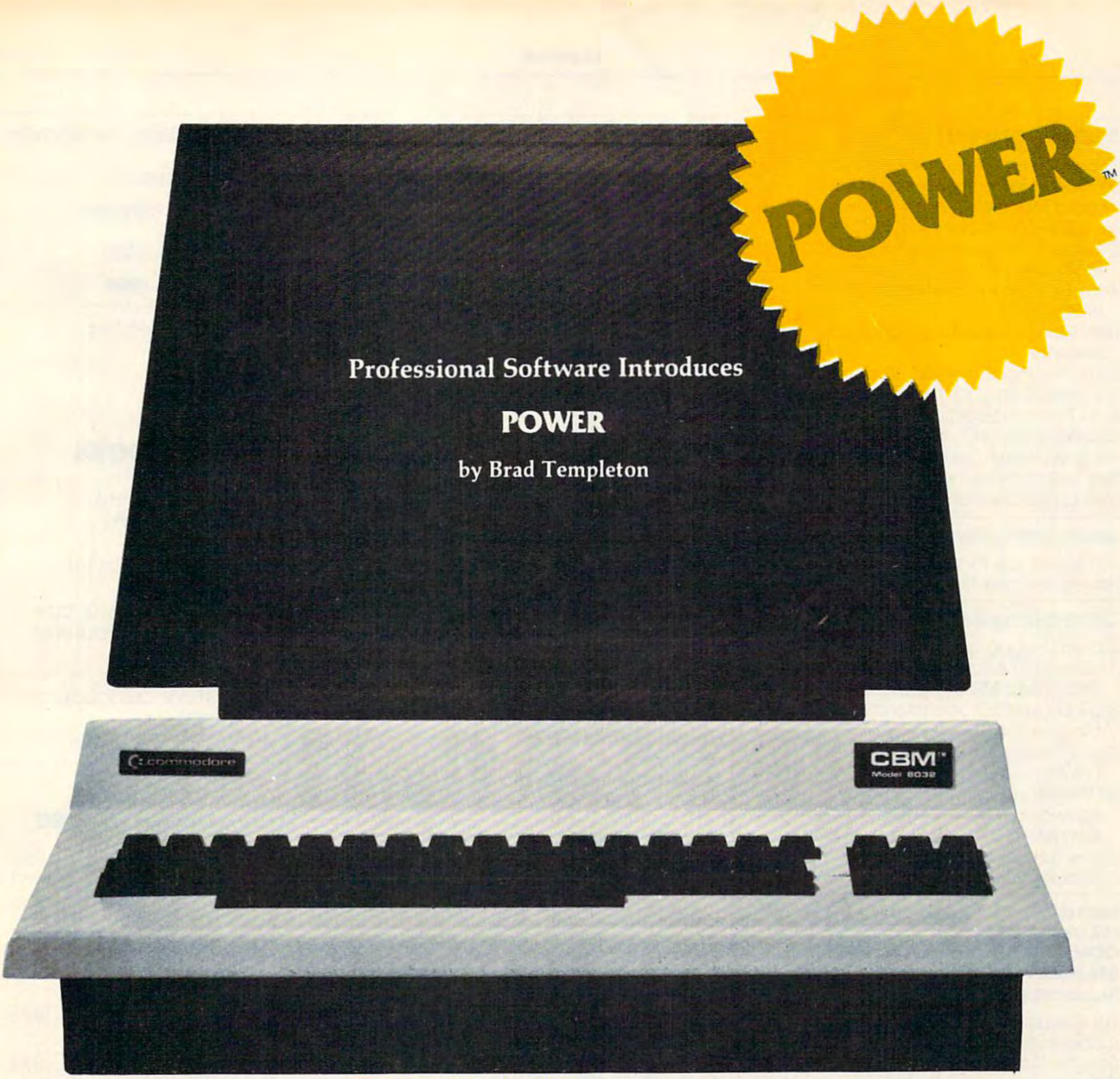
Atari, Inc. has just slashed the suggested retail price of the Atari 800 system from $1,080.00 to $899.00. Commodore is currently introducing two new machines that promise to be quite competitive in the personal market. Shown at the Consumer Electronics Show in Las Vegas the first week in January: a "game" computer with plug-in cartridges and a flat keyboard for around $150.00. And you can add a BASIC cartridge to learn programming. On the "high" end, as it were, and also from Commodore: a 64K color, graphics computer (also for TV connection) said to retail for less than $600.00. And that's with the 64K of memory. Look's like 1982 will surely be an interesting year!.

And As The Industry Grows

In recent editorials, we've commented on software protection, copyrights, the right of back-up, and more. We welcome your thoughts on these and other areas of interest to the personal computer consumer. A letter from a subscriber raised another question that we haven't considered and will put on the 1982 "comments coming" list. I'll raise some of his points here and solicit your input.

> ...I just resisted purchasing an expensive piece of computer software for which the warranty reads in part:
>
> > All...computer programs are distributed on an "as is" basis without warranty of any kind. The entire risk as to the quality and performance of such programs is with the purchaser. Should the programs prove defective...the purchaser and not the manufacturer...assumes the entire cost of all necessary servicing or repair. (The company) shall have no liability or responsibility to a purchaser.
>
> This is not mere legal jargon. It's the embodiment of a business philosophy which seriously harms all of us... To software companies I say: Accept responsibility for your products. Get the bugs out *before* you sell them. Don't try to sell a program debugged by your customers as a "revised" or "improved" product at additional cost.
>
> To software consumers I say: If possible, avoid products for which there is no warranty. Don't buy on faith. Complain loudly to software companies which provide no warranty...

Our reader makes a series of interesting points. While I'm no lawyer, I would wonder if the portions of the warranty shown above are realistic in enforceability. Would some of you lawyer/readers care to join this discussion with the rest of us? I'll look forward to your comments. ©



ADD POWER™ TO YOUR COMMODORE COMPUTER

$89.95

POWER produces a dramatic improvement in the ease of editing BASIC on Commodore's computers. POWER is a programmer's utility package (in a 4K ROM) that contains a series of new commands and utilities which are added to the Screen Editor and the BASIC Interpreter. Designed for the CBM BASIC user, POWER contains special editing, programming, and software debugging tools not found in any other microcomputer BASIC. POWER is easy to use and is sold complete with a full operator's manual written by Jim Butterfield.

POWER's special keyboard 'instant action' features and additional commands make up for, and go beyond the limitations of CBM BASIC. The added features include auto line numbering, tracing, single stepping through programs, line renumbering, and definition of keys as BASIC keywords. POWER even includes new "stick-on" keycap labels. The cursor movement keys are enhanced by the addition of auto-repeat and text searching functions are added to help ease program modification. Cursor UP and cursor DOWN produce **previous** and next lines of source code. COMPLETE BASIC program listings in memory can be displayed on the screen and scrolled in either direction. POWER is a must for every serious CBM user.

Call us today, for the name of the Professional Software dealer nearest you.

Professional Software Inc.
166 Crescent Road
Needham, MA 02194
Tel: (617) 444-5224 Telex #951579

TM POWER is a Registered Trademark of Professional Software, Inc.

Staff of COMPUTE!

Robert C. Lock, Publisher/Editor
Kathleen Martinek, Managing Editor
Richard Mansfield, Assistant Editor
Alice S. Wolfe, Publisher's Assistant
Georgia Papadopoulos, Art Director/ Production Manager
Harry Blair, Director, Advertising Sales/ Promotion
Terry Cash, Typesetting/Production Assistant
De Potter, Production Assistant
Dai Rees, Production Assistant
Kate Taylor, Production Assistant
Charles Brannon, Editorial Assistant
Sonja Whitesell, Dealer Coordinator
Bonnie Valentino, Accounting Coordinator
Fran Lyons, Circulation Coordinator

Associate Editors

Jim Butterfield, Toronto, Canada
Harvey Herman, Greensboro, NC

Contributing Editors

Marvin DeJong, Dept. of Mathematics -Physics, The School of the Ozarks Pt. Lookout, MD 65726
Eric Rehnke, 1067 Jadestone Lane, Corona, CA 91720
David Thronburg, P.O. Box 1317, Los Altos, CA 94022
Bill Wilkinson, Optimized Systems Software, 10379-C Lansdale, Ave. Cupertino, CA 95014
Gene Zumchak, 1700 Niagara St., Buffalo, NY 14207

Subscription Information (12 Issue Year): COMPUTE! Circulation Dept. 515 Abbott Drive Broomall, PA 19008 USA

U.S. $20.00
Canada $25.00 (U.S. funds)
Europe: Surface Subscription, $25.00 (U.S. funds)

TOLL FREE Subscription Order Line
800-345-8112
In PA 800-662-2444

Canadian Retail Dealers should contact:
Micron Distributing
409 Queen Street West
Toronto, Ontario M5V 2A5
416-361-0690

Advertising Sales

If you're in **Oklahoma, Texas or the Western States, we're now represented by Phoebe Thompson and Associates.** Give them a call for space reservations, contract/insertion information or questions. You can reach them through the following offices:

Northern California, Pacific Northwest, Rocky Mountain States, Texas, Oklahoma
Phoebe Thompson and Associates
101 Church Street, Suite 13
Los Gatos, CA 95030
408-354-5553
PHOEBE THOMPSON

Southern California, Arizona, New Mexico, Nevada
Phoebe Thompson and Associates
P.O. Box 700-462
Redondo Beach, CA 90277
JOANN SULLIVAN
213-378-0143

If you're in the East, we're now represented by **The Gittelman Company**. You can reach them through the following offices:

New England, New York State:
The Gittelman Company
Statler Office Building
Suite 518
20 Providence Street
Boston, MA 02116
617-451-0822
JOAN DONAHUE

The Gittelman Company
Summit Office Centre
7622 Summit Avenue
Fort Washington, PA 19034
215-646-5700
DOUG JOHNSON

New York City Metro Area, Mid-Atlantic and Southeastern States:
Local Numbers:
New York 212-567-6717
Atlanta 404-523-1252

If you're in the **Midwest, we're now represented by GB & Associates**. You can reach them through the office of:

GB & Associates
P.O. Box 335
Libertyville, IL 60048
312-362-1821
GORDON BENSON

COMPUTE! Home Office Contacts

Advertising Sales Manager, Harry Blair
Advertising/Production Coordinator, Alice S. Wolfe

Address all advertising materials to:
625 Fulton Street
Greensboro, NC 27403 USA

Mailing address: COMPUTE!
Post Office Box 5406
Greensboro, NC 27403 USA

Telephone: 919-275-9809

Small System Services, Inc. publishes:

COMPUTE!
The Journal For Progressive Computing

COMPUTE! Books

Robert C. Lock, President
W. Jerry Day, Corporate Comptroller
Kathleen Martinek, Assistant To The President
Ellen Day, Accounting Manager
Carol Holmquist Lock, Research Assistant

Corporate offices are located at 625 Fulton Street, Greensboro, NC 27403 USA

COMPUTE! Subscription Rates

US	$20 (one yr.)
	$36 (two yrs.)
	$54 (three yrs.)
Canada and Foreign Surface Mail	$25
Air	
Europe, US Possessions	$38
Middle East, Central America and North Africa	$48
South America, South Africa, Far East and Australia	$88

Coming In The March Issue of COMPUTE!:

Home Energy Workbook– Third In The Series

Atari Disk Data Structure

PET/CBM Disk Checkout

And Much, Much More...

Ask The Readers

Robert Lock, Richard Mansfield, And Readers

Please address any questions or answers to: Ask The Readers, **COMPUTE!** *Magazine, P.O. Box 5406, Greensboro, NC 27403.*

Answers

"This is in response to Jerry Stern's question in the August 1981 issue, concerning the use of a keypad controller with the Atari 800. All of Atari's controllers are usable with all of their computers, video game included. And the keypad controller works quite well for repetetive numerical data. However, it is not limited to use with numbers. The computer's response to the keypad controller is defined by the program you write or load into it. Therefore you can use it to output any one of twelve symbols or execute any of twelve commands or any combination of both the above.

There is a program on page H-14 of the Atari Basic Reference Manual *that will get you started with using the controller. The* Operating System Manual *goes into detail on how it works.*

Point of Interest: In [**COMPUTE!** #14] *his excellent article, 'Atari Tape Techniques,' Richard Kruse mentions that the use of LIST "C"/ENTER "C" can reduce the size of your BASIC program, but that the reason for this is undocumented. The documentation is on pages 2 and 3 of the* Basic Reference Manual, *under the section titled 'Variable Name Limit.'"* Roberta L. Mevis

"I enjoyed the article by William Taylor in **COMPUTE!** *#17. Apparently, the Stringy Floppy was interfaced using the regular cassette SAVE and LOAD routines. Your readers might be interested to know that a full ROM-based operating system and plug-in hardware board exist for mating the Stringy Floppy with the OSI C1P. It operates at 7200 BAUD, is available at power-up, and resides in the otherwise-unused memory locations from $E800-$EFFF. Information on the availability of this device may be obtained by writing: MSB Electronics, Barre-Montpelier Road, Barre, Vermont 05641."*
Philip K. Hooper

[On PET/CBM disks] "The ID contained in a track and sector is written once only – when the disk is NEW/HEADER-ed or created with a BACKUP/DUPLICATE. It can never be written again; disk writes don't touch this special area.

If you wanted to write a non-standard ID into a track-sector header, you'd have to do it by generating your own NEW formatting routine. This is not an easy trick, since it involves downloading a program into the inner processor of the disk. It would almost certainly involve destroying all information written on at least one track of the disk." Jim Butterfield

"In response to Mr. Keplinger's commentary on Computer Assisted Instruction – Worth The Effort?, I say a big positive yes. Going on four years without a promotion in the Air Force, I purchased the Atari 400 computer. I programmed all of my study material multiple choice questions and answers that I could find and had the computer drill me day after day. When it came to test for promotion, it seemed that I knew all the answers. A month later, I was notified of my promotion. Computer assisted instruction really works." Bob Holsti

"Re: Question from John Fry about files in OS65D 3.0 **COMPUTE!** #18.

It seems that although the program example is complete, there was no mention of the creation of a buffer. The program on my C4P MF was completely erased as I knew it would be when I typed it in to confirm my suspicions. Since OSI uses the beginning of the workspace for file buffers, the file is brought on top of the program or portions thereof when the open occurs.

There are many inordinate constructions in OS65D, one of which is the placement of the buffers. I have successfully edited the DOS to place the buffers at the top of memory. The advantage is that I can now easily write programs with sequential and/or random files and need not use the awkward CHANGE utility to create the buffers before I write the program. (This was an enormous help to me as virtually 80% to 90% of the programs I write use random files.)" Ross C. Votaw

Questions

I was delighted with the idea of 'The Unwedge – Tape Append and Renumber' by David Hook in the Sept. '81 issue, p. 103, but ran into problems when I tried it out:
1. The formula in line 26: QV$ = MID$ (STR(4 + 2 (QV = 1)),2). [What can the* QV = 1 *mean?]*
2. [What about the DATA statement in line 116?]..."
John Sweeney

Thanks for the kind words, John. The program was completely rechecked and, indeed, the final number in that DATA line should be H259, not

NASIR
GEBELLI
PRESENTS:
PATHFINDER
PATHFINDER
Never before released, a super arcade-adventure game for Atari . . . alone in a vast transparent maze inhabited by radioactive monsters and nuclear waste, you must blast the waste and free the maze of mutant monsters. By Solitaire Group, requires 24K Atari with disk drive.
MATCH
RACER
MATCH RACER
Brand new for the Atari, this one or two player game tests your driving skills on a race track filled with treacherous obstacles. See how far you can get before your luck and skill run out! By Solitaire Group, requires 16K Atari with disk drive.
ANDROMEDA
ANDROMEDA
Introducing a spectacular, original arcade-adventure for the Atari . . . a battle in inner space . . . you must invade a living organism and destroy its tissues before its antibodies destroy you. By Solitaire Group, requires 16K Atari with disk drive.
HOLLYWOOD
GEBELLI
Gebelli Software Inc.
1771 Tribute Rd., Suite A, Sacramento, CA 95815
1981 Gebelli Software, Inc. Atari is a registered trademark of Atari, Inc.

H25. We make every effort to assure that typos do not get into **COMPUTE!**, and we feel that we have succeeded in eliminating most of the causes. However, publishing 20 to 30 programs each month results in an occasional error. We attempt to announce any corrections (or useful, optional program modifications) the following month in our CAPUTE section in the back of the magazine.

As to the meaning of QV = 1, programmers sometimes choose to use *relational expressions*. Try this in immediate mode: ? 5 = 5. Then try: ? 5 = 2. As you will see, if the proposed equality is true, the "value" of the equality is -1. If false, it's zero.

```
10 INPUT X,Q
20 Y=Y-(X=Q)
30 ?Y:GOTO 10
```

Since subtracting a negative from a positive is, in fact, "addition" – the program above will increase the value of Y whenever X and Q are equal. You could achieve the same result with: 20 IF X = Q THEN Y = Y + 1. Strings can also be used as expressions and evaluated in this way. Likewise, such statements as: IF NOT X THEN PRINT "-1" or IF X THEN PRINT "0" will trigger the THEN action on -1 and zero, respectively.

"I know that there are screendump programs which exist for Atari which will allow the contents of the screen to be put on a printer. What I need is something similar to this but allowing the screen to be saved to tape (or in DATA statements written by the program itself) so that that screen could later be recreated easily. I am currently trying to write a graphics type of Adventure game which uses a redefined character set and requires numerous POSITION and PRINT commands to use those special characters to draw some fairly complicated floorplans. I am nearly at the tearing-out-my-hair point from trying to code these floorplans. It would be much simpler if I could draw the room, using the edit and cursor control functions, and then save screen to be used in the playing of the game itself. Does such a pair of utilities exist or is anyone currently working on one?" Michael A. Ivins

---

"I am an electronics instructor at a technical school and I am interested in programming my own PROMs. We have the KIM-1 at our school and I understand that they can be adapted for this purpose. [Please advise where I can obtain] a schematic and/or instructions."
Mark Iskovitz

"In **COMPUTE!** *#16 you had an article for Applesoft on loading tape. Well, we have an Atari 800, and my husband is having trouble loading and unloading tape, he keeps losing his program... Could you publish something in regards to it?"* Mrs. W. Phipps

Here are some suggestions and precautions: 1. Use the more expensive, better quality tapes. 2. Remember to issue an LPRINT before any CSAVE. 3. Always have two copies (in case one goes bad). 4. Make sure that files are saved with the aid of the digital counter. Don't overlap. 5. When loading, try several times with the tape positioned slightly differently each time (via FFwrd). 6. Clean the tape heads and the rubber wheel that pulls the tape through with cleaning solution and demagnetize the heads with a demagnetizer (both items available at electronics stores). 7. Don't leave the Play or Play and Record buttons down for any long periods. 8. Experiment with alternative ways to SAVE programs: CSAVE/CLOAD, SAVE"C"/LOAD"C", or LIST"C"/ENTER"C". 9. If, after all this, you're still experiencing problems – chances are your heads are misaligned or something else is electronically wrong. Take the recorder in for professional adjustment.

"As owner of a CBM 8032, with 4040 disk drive and an Epson MX-80, I have been reading current and back issues of **COMPUTE!** *ever since my computer dealer introduced me to the magazine several months ago. While I have found much in the magazine of interest, that interest has bordered at times on desperation arising out of statements such as 'This program will not work with the new ROM or with the 80 column screen.'*

How about a program that will rewrite any other program from old or new ROM 40-column to 80-column format, including assembly-language programs? While you are at it, why not a program that will translate from CP/M programs, especially the hundreds of TRS-80 programs, to 'PET BASIC'. I am sure that many of your readers would 'rise up and call you blessed.'"
Dr. Harold Peters

Your suggestion is an excellent one, and we would welcome programs which stand between various machines and translate and harmonize. Unfortunately, writing a program which *emulates* another computer is not an easy task. Some work has been done in this direction, though. In **COMPUTE!** #6 is "Feed Your PET Some Applesoft." Going in the other direction, in **COMPUTE!** #8 is "Load PET Programs Into The Apple II." An extraordinary program which snaps the 80 column CBM screen into 40 columns appears in **COMPUTE!** #12: "Running 40 Column Programs On A CBM 8032." Also, for hand-translations, see **COMPUTE!** #16, "PET, Apple, Atari: On Speaking Terms." Time permitting, we translate individual programs, often presenting multiple versions of programs appearing in the PET Gazette. In addition, we print several programs each month (in the Applications section at the front of the magazine) with versions for both Atari and Microsoft BASICs. ©

Computers And Society

David D. Thornburg
Innovision
P.O. Box 1317, Los Altos, CA 94022

On Piracy ...

As the mist cleared we could see our goal before us. High in the foothills overlooking the San Francisco Bay, the opening to the cave was unmarked – except for the power lines which snaked their way into the cavern through a crack in the rock.

"You are on your own now," my guide said as he scurried down the hill. At last I had found the home of the famed software pirate, Long John Silicon.

As I entered the cave, I asked myself why an editor of a prestigious magazine would risk his life in pursuit of a story, but the recent lawsuit preventing Long John Silicon from selling his home video copy of Tooth Fairy was too exciting to ignore.

With a great heave, I opened the door and found Long John sitting at a keyboard, ready for our interview.

DT: Long John, you have a reputation as a vicious software pirate. Tell us – is it deserved?

LJS: Aye matey! I am the meanest software pirate to ply the 57 keys. Once I see a game I like, it is only a matter of time (usually months) before the game is up and running on the computer of your choice.

DT: Wait a minute. I'm not sure I understand what you mean. I thought software pirates just made carbon copies of other people's software.

LJS: Copy existing programs? Ha Ha! Oh matey, you must be kidding! My parrot wouldn't do something that easy. No, what I do is the true pirate's craft. I slink around the arcades looking for new games. When I first saw Tooth Fairy I knew that riches were at hand.

DT: Once you find a game you like, how do you go about copying it?

LJS: First, I spent many pieces of eight playing Tooth Fairy, gaining mastery in every aspect of the game. In the space of a few weeks I was playing the game in my sleep. Next I created a story board for the game.

DT: Excuse me, Long John, I'm not too versed in the pirate's craft. Would you tell our readers what a story board is?

LJS: Of course. A story board is a visual map showing the play of the game. It includes pictures of the screen and so on.

DT: That sounds like a lot of work to go through before writing any of the program.

LJS: Of course it is. Who said piracy is easy work? In any event, once the story board is finished the real work begins. One just doesn't sit down and copy a game without worrying about display resolution, color, machine speed, game controller options – ah, the stories I could tell...

DT: Yes, well I am sure our readers would be fascinated, but tell us more about the game. Is it an exact copy of the arcade version?

LJS: The same? How insulting! I've half a mind to slit you from your index register to your stack! No pirate would miss the chance to improve on a game. To start with, I spent about as much time copying Tooth Fairy as its creator's spent designing it in the first place. Why shouldn't I improve the game.

DT: Oh, I agree with that; but why is this piracy then. After all, people who write love stories aren't being sued by Shakespeare's estate. From what I can see, you might have created a new game.

LJS: No! A thousand times, no. If my version of Tooth Fairy was new, I wouldn't have been sued for infringement by Ajax Computer Company would I?

DT: I guess not. Say, your copy of Tooth Fairy runs on the Ajax computer doesn't it?

LJS: Yes, in fact it has helped sell their computers. Most of the local computer stores used to use my game to show off the Ajax's power.

DT: Then why are they suing you? They didn't write the original software.

LJS: Yes, that's true; but they did buy the home video rights to the game, so I guess I infringed on their copyright, even though they didn't write any of the original program.

DT: Well, I'll bet that their version of Tooth Fairy is a real knockout. Now that your program is illegal, I assume I can buy theirs.

LJS: Oh no! First of all, they have to go through all the work I did to get the game to run on their computer. Their version is at least six months away.

DT: What a shame! Why didn't they just license your copy?

LJS: License me! A pirate! Shiver me disks. Why would they do that? Of course I asked for a license, but they insisted on having the job done over.

DT: I'm not trying to downplay the devious immorality of your crime, but I am still having a hard time thinking of your work as piracy. Tell me, are you countersuing?

LJS: Aye, a countersuit is in progress, but I don't have the legal resources of a company like Giant Toys, Inc.

DT: Oh yes, Giant Toys sells a hand-held version of Tooth Fairy. Is Ajax suing them too?

LJS: Not yet. Ajax only has a few attorneys, and they can't sue everyone at once. I'm flattered that they picked me first, even if I have to give up the battle when my money runs out. After all, even pirates live in fear of their lawyers.

DT: What a shame. You mean that you might lose because you can't afford the fight?

LJS: Yes. After all, if Giant was sued, the case might drag on for years. My suit will probably be mercifully short. My days as a pirate are nearing an end.

DT: So you have given up on piracy forever then?

LJS: Not exactly. Just last night, for example, I used my video cassette recorder to make a copy of a movie that was broadcast past my bedtime. ©

The Beginner's Page

Translating Equations

Richard Mansfield
Assistant Editor

Computers are excellent teachers. They have infinite patience; provide instant pass-fail corrections of your efforts; permit you to work at your own speed on topics of your choice; and they don't (as yet) become sarcastic when you blunder.

Many people, myself included, decided long ago that math was not their forté. This decision is usually made at age fifteen or thereabout and follows a series of mishaps in the educational system. Algebra is often the final blow.

Computers cure this math phobia rather quickly. The machine does all of the tiresome calculations for you. You are free to float above and observe relationships, discover patterns, even construct visual analogs where you can watch the numbers transform on the screen.

Algebraic Equations, BASIC Assignments

After you get over the initial surprise that, in BASIC, A = A + 1 makes perfect sense – you will find that the meaning of *variable* becomes quite clear. A variable is simply a "name" written on a "box." You might have a box in your house marked "BILLS." Each month you pay all the bills and the box is empty (BILLS = 0). Then, when each bill comes in, you put it in the box (BILLS = BILLS + 1). This is not an algebraic equation, it is an *assignment* of a certain number (BILLS + 1) to the variable BILLS.

In algebra, an equation is expected to balance: whatever is on the left side of the equals sign is presumed to be equal to the right side. In BASIC, the variable on the left side is *being defined* by whatever is on the right side. In earlier versions of BASIC, you had to type: LET BILLS = BILLS + 1 to show that you were *assigning* a new value to BILLS, not stating an equality. One other thing: computers allow you to use meaningful, easily recognized variable names such as BILLS, or INTEREST, or DOLLARS. This, too, can be an advantage over the traditional single-letter variable names of algebra.

In any case, much useful math becomes clear after you work a while on your computer. For example, let's put this on our computer (to see how easy it really is):

$$F = D(1 + I/C)^{C*Y}$$

or

$$\mathrm{FINALAMOUNT} = \mathrm{DOLLARS}(1 + \mathrm{INTEREST}/\mathrm{COMPOUNDING})^{\mathrm{COMPOUNDING} * \mathrm{YEARS}}$$

This formula will let you know how much money you'll end up with after making an investment. It can also tell you how much your house will be worth if it is going up in value a certain amount each year or show the effects of inflation. It's a handy formula, but to the "non-mathematical" it looks forbidding. On the computer, it's a snap. Just use INPUT statements to ask for each of the variables and then, (in line 100), duplicate the formula using BASIC symbols:

```
10 PRINT "WHAT IS THE ORIGINAL AMOUNT IN
     VESTED";
20 INPUT DOLLARS
30 PRINT "HOW MANY YEARS BEFORE YOU CASH
     IN THE INVESTMENT";
40 INPUT YEARS
50 PRINT "WHAT IS THE ANNUAL INTEREST RA
     TE";
60 INPUT ITEREST
70 PRINT "HOW MANY TIMES PER YEAR IS IT
     COMPOUNDED";
80 INPUT COMPOUNDING
90 ITEREST = ITEREST/COMPOUNDING/100:REM
     MAKE ITEREST INTO A DECIMAL FRA
     CTION
100 FINALAMOUNT = DOLLARS * (1 + ITEREST)
     ↑ (COMPOUNDING * YEARS)
110 PRINT "AT THE END OF "; YEARS; "YEARS
     YOU WILL HAVE $"; FINALAMOUNT
```

Notice that we spell it *iterest* to avoid using one of BASIC's special, reserved words *INT*. It is also necessary to enclose *compounding multiplied by years* in parentheses to show that this is to be calculated before the other part is raised to a power. The order in which calculations are performed is, of course, quite important and you should familiarize yourself with what your computer's manual instructs on this subject. When in doubt, use parentheses – they will always cause whatever is within them to be figured first.

The Universal Rounding Engine

Programs can often be refined, customized, and made to perform new functions with surprisingly little effort. This same program could include a function to round off the *finalamount* to the nearest penny by adding this line:

```
105 FINALAMOUNT = INT(FINALAMOUNT * 100
     + .5)/100
```

What would this Universal Rounding Engine do if you changed the two 100's to 1000's...or 10's? We can also easily adjust the program to predict how much your house will be worth in ten years, given a rise in value of, say, six percent per year. The math stays the same, all we need to do is change the *prompts* (the questions the computer asks). Line 10 should read: "HOW MUCH IS YOUR HOUSE

WORTH NOW?" Line 30: "HOW MANY YEARS DO YOU WANT TO PROJECT?" Line 50: "HOW MUCH IS IT INCREASING IN VALUE EACH YEAR?" Line 70: COMPOUNDING = 1. In line 110, change "YOU WILL HAVE" to "YOUR HOUSE WILL BE WORTH."

To work with inflation projections, make the following replacements:

```
10  "WHAT IS THE COST OF THE ITEM TODAY";
50  "WHAT IS THE ANNUAL INFLATION RATE";
70  COMPOUNDING=1
110 [change YOU WILL HAVE to: IT WILL COST]
```

When creating such useful variations to simple programs, you are, at the same time, learning new things about mathematical relationships. It's fun and therefore painless. As an experiment with the Inflation version of this program, try adding: 5 THISYEAR = 1982 so the computer will know what year it is. Then, using the information gathered in line 30, have the computer give its answer (in line 110) in the form: BY THE YEAR 1985 IT WILL COST $(whatever).

All of this is worlds away from that algebra class where some of us mistakenly decided that mathematics, when it wasn't impossibly obscure, was tedious. By pushing and shaping programs, you can see and feel numbers, their interactions, their beauty. ©

Guest Commentary

Hidden Costs Of Computer Technology

Craig Brod
President, Technostress International, Inc.

During the 1970's most banks computerized their operations. At one California bank, a team was assigned to develop a program to pay savers their interest automatically on the first of each month. The task was completed to everyone's apparent satisfaction and most members of the team were reassigned to new projects. The day before the first automatic payments were to be disbursed, due to a fluke – a favored customer being handed his computerized check a day early – it was discovered that the bank had overpaid everybody as much as double the interest due them. At 6:00 P.M. the team's remaining analyst was called in. The project manager came. His manager came. The vice president came. An estimated eight to ten million dollars in bank funds were on the line, to be disbursed when the doors opened for business the next morning. Could the analyst find the flaw in his team's program? Could he develop an algorithm to withdraw the appropriate amount of overpayment from each of the savers? This analyst was a prime candidate for, if not the victim of, *technostress*.

Exactitude, Repeatability, Detail

Computer technology has become a fact of organizational life and has brought with it new values and new costs for those within the organization. Exactitude, repeatability, and close attention to detail are the hallmarks of everyday operations. Computer technology promotes formal relations between people, their machines, and their environment.

The new technology is qualitatively different from the old. Compared to a computer information system, a telephone or xerox machine were simple communication devices whose use required a minimum of quiet and concentration and whose users had a great deal of latitude. The computer, on the other hand, requires a specific response time from the user (turned *operator*).

The recording of information and the retrieval of data within the language of the program both place constraints on the operator, who has become machine-dependent and works in a captive environment. Control over sound, lighting, and work flow is important for maximum concentration and effective management of data. While assembly-line work, or even typing, are sometimes grueling types of work requiring attention to detail, the demands on the computer operator are unlike those heretofore known in the workplace.

> **"Computer technology often reverses the relationship between age, experience, and competence at work."**

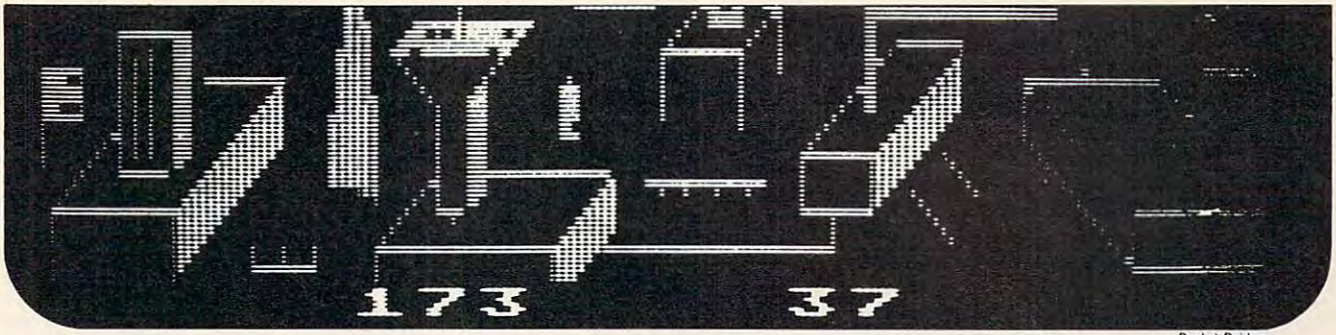
Captive environments and machine-dependent people are indications of new forms of organizational life, and one result is technostress: the condition resulting from the inability of a person or organization to cope with the demands created by the operation and maintenance of computer technology. It occurs where necessary technological stress (such as response to work changes) is translated into unnecessary human strain. There are examples of technostress at all levels.

Age, Experience And Competence: A Reversed Relationship

Computer technology often reverses the relationship between age, experience, and competence at work. Unlike managers of the past who passed tips on to new employees on how to "kick the ditto machine" to make it work, their years of experience have often merely accumulated outdated knowledge in today's managers. And they are usually at a disadvantage to young recruits who command a great deal of recent technical knowhow.

Today's project manager has no reliable way to measure productivity. The manager functions as a go-between, talking to the system user – say the department of a bank that wishes interest payments computerized – and then schedules it, deciding whether it should take six people three months, four people a year, or whatever.

To most programmers, such schedules are a joke: "You could throw darts at a board and do as well," is often heard. One told recently of a project scheduled for three months and which came in at two and a half, earning the group praise; yet it could have been done in three weeks and, that the group dallied, constituted a mini-revolt. No matter; as long as the projects come in ahead of arbitrary schedules, managers are happy. Able to measure only results and not understanding the work well enough to gauge productivity, many of today's managers who have not come up through the



Rocket Raiders

Our High Quality Software Is More Than A Stroke Of Genius... It's A Work Of Art.

□ **PM EDITOR:** by Dennis Zander (Atari, 16K)
Create your own fast action graphics game for the Atari 400 or 800 using its player missile graphics features. By using player data stored as strings, players can be moved or changed (for animation) at machine language speed. All this is done with string variables (POS(Y)=SHIP4). This program is designed to permit creation of up to 4 players on the screen, store them as string data and then immediately try them out in the demo game included in the program. Instructions for use in your own game are included. PM EDITOR was used to create the animated characters in **ARTWORX** RINGS OF THE EMPIRE and ENCOUNTER AT QUESTAR IV.
PRICE .... **$29.95** cassette **$33.95** diskette

□ **ROCKET RAIDERS** by Richard Petersen (Atari 24K)
Defend your asteroid base against pulsar bombs, rockets, lasers, and the dreaded "stealth saucer" as aliens attempt to penetrate your protective force field. Precise target sighting allows you to fire at the enemy using magnetic impulse missiles to help protect your colony and its vital structures.
PRICE .... **$19.95** cassette **$23.95** diskette

□ **INTRUDER ALERT!** by Dennis Zander (Atari, 16K)
This is a fast paced action game in which you must escape from the "Dreadstar" with the secret plans. The droids are after you and you must find and enter your ship in order to escape. If you fail, the rebellion is doomed. **PRICE** .... **$16.95** cassette **$20.95** diskette

□ **THE RINGS OF THE EMPIRE:** by Dennis Zander (Atari 16K)
The Empire has developed a series of battle stations protected by one or more rings of energy. You must destroy these weapons by attacking them in your Y-wing fighter armed with Zydon torpedoes. Each time you blast through the rings and destroy the station, the Empire develops a new station with more protective rings.
PRICE .... **$16.95** cassette **$20.95** diskette

□ **FOREST FIRE!:** by Richard Petersen (Atari,24K)
Using excellent color graphics, your Atari is turned into a fire scanner to help you direct operations to contain a forest fire. You must compensate for changes in wind, weather and terrain. Not protecting valuable property can result in startling penalties. Life-like variables make FOREST FIRE a very suspenseful and challenging simulation. **PRICE** .... **$16.95** cassette **$20.95** diskette

□ **PILOT:** by Michael Piro (Atari, 16K)
Pilot your small airplane to a successful landing using both joysticks to control throttle and attack angle. PILOT produces a true perspective rendition of the runway, which is constantly changing. Select from two levels of pilot proficiency.
PRICE .... **$16.95** cassette **$20.95** diskette

□ **ALPHA FIGHTER:** by Douglas McFarland (Atari, 16K)
Consisting of two different programs, ALPHA FIGHTER requires you to destroy the alien starships. As you become more successful, the games get harder and harder. **PRICE** .... **$14.95** cassette **$18.95** diskette

□ **GIANT SLALOM:** by Dennis Zander (Atari, 16K)
Bring the Winter Olympics to your computer anytime of the year! Use the joystick to guide your skier's path down a giant slalom course consisting of open and closed gates. Choose from three levels of difficulty. Take practice runs or compete against from two to eight additional skiers.
PRICE .... **$15.95** cassette **$19.95** diskette

□ **HODGE PODGE:** by Marsha Meredith (Apple 48K, Applesoft or Integer BASIC)
This captivating program is a marvelous learning device for children from 18 months to 6 years. HODGE PODGE consists of many cartoons, animations and songs which appear when any key on the computer is depressed. A must for any family containing young children and an Apple.
PRICE .... **$19.95** diskette

□ **STUD POKER:** by Jerry White (Atari, 16K)
This is the classic gambler's card game. You will find the computer to be a worthy opponent who occasionally bluffs but never cheats! STUD POKER employs all of the Atari's sound, color and graphics capabilities.
PRICE .... **$14.95** Cassette **$18.95** diskette

TYPE-'N-TALK*

ARTWORX is offering the fantastic TYPE-'N-TALK™ from Vortrax*. This easy-to-use unit connects to your computer's serial port. Text is automatically translated into electronic speech enabling the TYPE-'N-TALK™ hobbyist to use and enjoy it immediately.
PRICE **$329.00**

The following **ARTWORX** programs are available for TYPE-'N-TALK:

STUD POKER (Atari,24K) .... **$16.95** cassette **$20.95** diskette
TEACHER'S PET (Atari,24K; North Star) **$16.95/ $20.95**
BRIDGE 2.0 (Atari, 24K; North Star) .. **$19.95 / $23.95**
NOMINOES JIGSAW (Atari, 24K) ..... **$17.95 / $21.95**

Please specify "TNT" version when ordering programs.

□ **CRANSTON MANOR ADVENTURE:** by Larry Ledden (Atari, North Star and CP/M)
You must enter mysterious Cranston Manor and attempt to collect its many treasures. This extemely challenging program will provide you with many hours (days?) of adventure. The program may be interrupted at will and your status saved onto the diskette.
PRICE .... **$21.95** diskette

□ **BLOCKADE:** by Edward Schneider (Atari, 16K)
Every games library needs Blockade program, and this is one of the best. Choose from three levels of difficulty and play against another person or by yourself against the clock.
PRICE .... **$14.95** cassette **$18.95** diskette

□ **TEACHER'S PET:** by Arthur Walsh (Atari, Apple, TRS-80, PET, North Star and CP/M (MBASIC) systems).
This is an introduction to computers as well as a learning tool for the young computerist (ages 3-7). The program provides counting practice, letter-word recognition and three levels of math skills.
PRICE .... **$14.95** cassette **$18.95** diskette

□ **FORM LETTER SYSTEM:** (Atari, North Star and Apple)
This is the ideal program for creating personalized form letters! FLS employs a simple-to-use text editor for producing fully justified letters. Addresses are stored in a separate file and are automatically inserted into your form letter along with a personalized salutation. Both letter files and address files are compatible with **ARTWORX** MAIL LIST 3.0 and TEXT EDITOR programs.
PRICE .... **$39.95** diskette

□ **TEXT EDITOR:** (Atari and North Star)
This program is very "user friendly" yet employs all essential features needed for serious text editing with minimal memory requirements. Features include common sense operation, two different justification techniques, automatic line centering and straightforward text merging and manipulation. TEXT EDITOR files are compatible with **ARTWORX** FORM LETTER SYSTEM.
PRICE .... **$39.95** diskette

□ **MAIL LIST 3.0:** (Atari, Apple and North Star)
The very popular MAIL LIST 2.2 has now been upgraded. Version 3.0 offers enhanced editing capabilities to complement the many other features which have made this program so popular. MAIL LIST is unique in its ability to store a maximum number of addresses on one diskette (typically between 1200 and 2500 names!). Entries can be retrieved by name, keyword(s) or by zip codes. They can be written to a printer or to another file for complete file management. The program produces 1, 2 or 3-up address labels and will sort by zip code (5 or 9 digits) or alphabetically (by last name). Files are easily merged and MAIL LIST will even find and delete duplicate entries! The address files created with MAIL LIST are completely compatible with **ARTWORX** FORM LETTER SYSTEM.
PRICE .... **$49.95** diskette

□ **THE VAULTS OF ZURICH:** by Felix and Ted Herlihy (Atari, 24K, PET)
Zurich is the banking capital of the world. The rich and powerful deposit their wealth in its famed impregnable vaults. But you, as a master thief, have dared to undertake the boldest heist of the century. You will journey down a maze of corridors and vaults, eluding the most sophisticated security system in the world. Your goal is to reach the Chairman's Chamber to steal the most treasured possession of all: THE OPEC OIL DEEDS!
PRICE .... **$21.95** cassette **$25.95** diskette

□ **BRIDGE 2.0** by Arthur Walsh (Atari (24K), Apple TRS-80, PET, North Star and CP/M (MBASIC) systems)
Rated #1 by Creative Computing, BRIDGE 2.0 is the only program that allows you to both bid for the contract and play out the hand (on defense or offense!). Interesting hands may be replayed using the "duplicate" bridge feature. This is certainly an ideal way to finally learn to play bridge or to get into a game when no other (human) players are available.
PRICE .... **$17.95** cassette **$21.95** diskette

□ **ENCOUNTER AT QUESTAR IV:** by Douglas McFarland (Atari, 24K)
As helmsman of Rikar starship, you must defend Questar Sector IV from the dreaded Zentarians. Using your plasma beam, hyperspace engines and wits to avoid Zentarian mines and death phasers, you struggle to stay alive. This BASIC/Assembly level program has super sound, full player missile graphics and real time action.
PRICE .... **$23.95** cassette **$27.95** diskette

□ **THE NOMINOES JIGSAW PUZZLE:**
by C. Minns/B. Brownlee (Atari, 24K, TRS-80, and Apple)
We quote . . ."A brainteaser supreme. . . the concept of NOMINOES JIGSAW is brillant. . . this video jigsaw game is so clever and completely original that only the most hardhearted puzzle hater could fail to be charmed."—ELECTRONIC GAMES MAGAZINE.
PRICE **$17.95** cassette (also available for TRS-80 color computer) **$21.95** diskette.

Artworx

Highest Quality Software*, Guaranteed.

ARTWORX SOFTWARE COMPANY
150 North Main Street Fairport, NY 14450 (716) 425-2833

Call ARTWORX toll-free number to order direct:

master charge

800-828-6573 In New York, Alaska, Hawaii call: (716)-425-2833

VISA

All orders are processed and shipped within 48 hours.

Shipping and handling charges:
Within North America: Add$2.00
Outside North America: Add 10% (Air Mail)
New York State residents add 7% sales tax.

Quantity Discounts:
Deduct 10% when ordering 3 or more programs

Write for FREE Catalogue listing more information about these and other quality ARTWORX programs.

Ask for ARTWORX at your local computer store.

*ATARI, APPLE, TRS-80, PET, NORTH STAR, CP/M, and TYPE'N TALK are registed tradenames and/or trademarks.

COLLEGE BOARD SAT* PREP SERIES

TRS-80, APPLE, PET, OSI, ATARI, CP/M, PDP-11

Each program confronts the user with a virtually limitless series of questions and answers. Each is based on past exams and presents material on the same level of difficulty and in the same form used in the S.A.T. Scoring is provided in accordance with the formula used by College Boards.

S.A.T., P.S.A.T., N.M.S.Q.T. — Educator Edition set includes 25 programs covering Vocabulary, Word Relationships, Reading Comprehension, Sentence Completion, and Mathematics. **Independent tests of S.A.T. series performance show a mean total increase of 70 points in students' scores.** Price **$229.95**

GRADUATE RECORD EXAM Series — Educator Edition includes 28 programs covering Vocabulary, Word Relationships, Reading Comprehension, Sentence Completion, Mathematics, Analytical Reasoning and Logical Diagrams. Price **$289.95**

COMPETENCY PROFICIENCY EXAM PREP SERIES

This comprehensive set of programs consists of simulated exam modules, a thorough diagnostic package, and a complete set of instructional programs. It is designed to teach concepts and operations, provide drill and practice and assess achievement levels through pre and post testing. The Competency Exam Preparation Series provides a structured, sequential, curriculum encompassing mathematical, reading and writing instruction.

This program is designed for individual student use or use in a classroom setting. Programs provide optional printer capability covering worksheet generation and performance monitoring. C.E.P.S. are available in three software formats.

Special editions available for California Proficiency Assessment Test and New York Regents Competency Tests. Call for Prices.

M.I.T. Logo for Apple $179.95

Odyssey In Time

This spectacular adventure game adds a new dimension of excitement and complexity to **Time Traveler.**

Odyssey In Time includes all the challenges of **Time Traveler** plus 10 additional eras. Each game is different and may be interrupted and saved at any point for later play. **$39.95**

Time Traveler

The best of the adventure games. Confronts the player with complex decision situations and the demand for real time action. Using the Time Machine, players face a challenging series of historical environments. To succeed you must build alliances and struggle with the ruling power. Each game is unique. **$24.95**

☆NEW☆ ☆MICRO-DEUTSCH☆

Micro-Deutsch set includes 24 grammar lessons, covering all material of an introductory German course. Four test units also included. Grammar lessons use substitution transformation drills, item ordering, translations and verb drills. Drill vocabulary based on frequency lists. Suitable for use with any high school or college textbook. Extensively field tested at SUNY Stony Brook. Available for Apple II and PET/CBM. (PET version includes a special foreign language character chip.) Also available soon: MICRO-FRANCAIS, MICRO-ESPAÑOL, MICRO-IVRIT, MICRO-YIDDISH, MICRO-CHINESE, MICRO-JAPANESE. **$179.95**

Isaac Newton + F.G. Newton

Perhaps the most fascinating and valuable educational game ever devised - **ISAAC NEWTON** challenges the players (1-4) to assemble evidence and discern the underlying "Laws of Nature" that have produced this evidence. **ISAAC NEWTON** is an inductive game that allows players to intervene actively by proposing experiments to determine if new data conform to the "Laws of Nature" in question. Players may set the level of difficulty from simple to fiendishly complex.

Full Graphics Newton. This version of Isaac Newton presents all data in graphic form. Because data is graphic rather than symbolic, this game is suitable for very young children. Players may select difficulty levels challenging to the most skilled adults. **$49.95**

☆ NEW ☆ ☆Pythagoras and The Dragon☆

Mathematics in a fantasy game context. Based on **The Sword of Zedek, Pythagoras and The Dragon** introduces Pythagoras as a mentor to the player. When called on for aid, Pythagoras poses math questions, and depending on the speed and accuracy of the player response, confers secret information. With Pythagoras as an ally, the quest to overthrow Ra, The Master of Evil, assumes a new dimension of complexity. Depending on the level chosen, problems range from arithmetic through plane geometry. 32K **$39.95**

Free Bonus with purchase of $300.00 or more: Applesoft Tutor Series

Krell Software Corp. has no official ties with the College Entrance Examination Board or the Educational Testing Service. Krell is, however, a supplier of products to the E.T.S.

PROGRAMS AVAILABLE FOR TRS-80, APPLE II, PET & ATARI

N.Y.S. residents add sales tax.

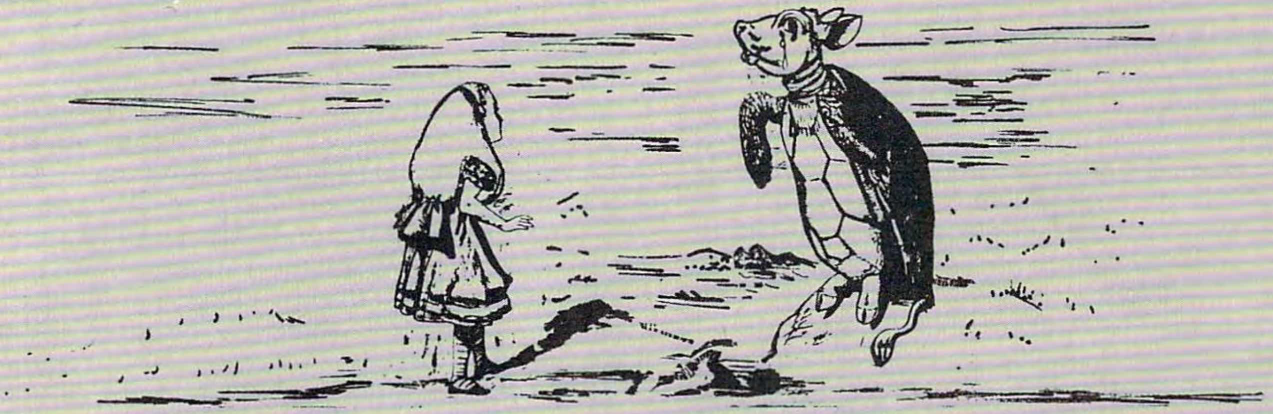
All programs require 16K ● TRS-80 programs require LEVEL II BASIC ● APPLE programs require Applesoft BASIC

KRELL SOFTWARE CORP.

"The State of the Art in Educational Computing"

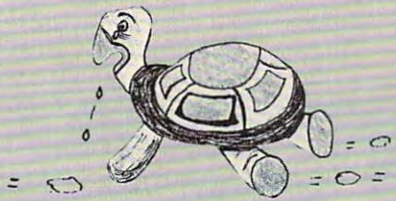
21 Millbrook Drive, Stony Brook, N.Y. 11790 (516) 751-5139

Mock Turtle



is crying. Why? Because he has learned that a leading microcomputer manufacturer is planning to market an imitation of **M.I.T.'s Logo** for Apple II*. He thinks that Alice and the world's children deserve the genuine article. Namely,

M.I.T. Logo for Apple II*



Krell's **M.I.T. Logo** for Apple II* is a copyrighted product of the Massachusetts Institute of Technology. Logo was developed under a grant furnished by the National Science Foundation. Krell's **M.I.T. Logo** for the Apple* includes many special features such as Beaver Graphics and Krell Instant Logo Tutor Package. The entire package is fully documented for teachers and students. Requires 64K **$179.95**

RELATED BOOKS

Apple Logo by Harold Abelson (McGraw Hill): indispensable **14.95**

Mindstorms by Seymour Papert (Basic Books) **12.95**

Computer Connections by Jean N. Nazzaro, Editor (ERIC) **16.65**

Memory expansion boards to 64K for Apple II*. **130.00**

FREE BONUS with purchase of $300.00 or more:
Applesoft Tutor Series

PROGRAMS AVAILABLE FOR TRS-80, APPLE II, PET & ATARI

N.Y.S. residents add sales tax.

* Trademark of Apple Corp.

KRELL SOFTWARE CORP.

"The State of the Art in Educational Computing"

21 Millbrook Drive • Stony Brook, New York 11790

(516) 751-5139

technical ranks lack the respect of their workers.

A Struggle Between Monotony And Perfectionism

To operate a computer is to live with stress, even for relatively low-level operators. Consider the operator whose job it is to process claims and transfer data from one source to another. The machine will have "peaks" and "valleys" – a job requiring thirty minutes at 7:00 A.M. may take several hours at mid-afternoon; and, every so often, due to overload, the whole system *crashes*.

These fluctuations fragment the worker's planning process, his ability to structure his workday. However, the machine makes no mistakes and turns out a uniform product. Given the repetitive nature of his task, the worker struggles between monotony and perfectionism. It is the machine that gets credit for a job well done, and there is no human feedback intrinsic to the system.

Analysts, those who write the programs, are familiar with the dreaded 2:00 A.M. phone call: "It blew up." And the challenge will be to fix it before 6:00 in the morning when perhaps thousands of other workers must depend on it, or, in the case above of the automatic interest payments, eight million dollars may ride on it. It is no surprise that many of them eat Maalox like candy.

An ace analyst is one who has few peers and earns little praise, due to the fact that so few understand what goes into his work. Knowledge builds with the number of systems upon which he has worked. This can have a snowball effect within the organization, always with increasing numbers of people asking "How does this program work?" about increasing numbers of systems. For some, it is a gradual process of becoming identified with the machine – more and more information demanded – with the effect that large chunks of self, time, and energy are drained. Tyrannized by their own expertise, the pressure ceases only when they change jobs. Then they are no longer responsible for every system upon which they have ever worked.

All of these problems are felt, but generally go unstated, surfacing as negative behavior – sabotage, absenteeism, last-minute sick calls, frequent job changes – or as direct problems with productivity. Employees and managers need more technical training, training on how to adapt to new technology, and better work designs.

The power of technology has silenced all but the gallows humor which has grown up around it: "If I hung myself, the machine would just keep plugging away." Technostress manifests itself in a variety of ways. ©

ROCKWELL Microcomputers from Excert, Inc.

THE AIM 65/40
Single Board or Smorgasbord

- A full size terminal style keyboard w/8 special function keys
- A smart, 40 character display with its own microprocessor
- A 40 column printer w/text and graphic output
- Up to 64K of on-board RAM and ROM
- On-board interfaces include RS232, dual audio cassette and 2 user I/O R6522 devices
- Firmware includes interactive monitor and text editor w/options of Assembler, BASIC, FORTH and PL/65

THE AIM 65
Take-Out Order

- A full size terminal style keyboard w/3 special function keys
- A 20 character display
- A 20 column printer w/text and graphic output capability
- Up to 4K RAM and 20K ROM on-board
- On-board interfaces include 20MA TTY, dual audio cassette and 1 user I/O R6522 device
- Firmware includes interactive monitor and text editor w/options of Assembler, BASIC, FORTH, PASCAL, & PL/65

And if the above isn't enough,
Try the RM65 — a product line filled with embellishments including:

32K DRAM Board	ACIA Board	Prototype cards
CRT Controller	IEEE-488Board	Adaptor Buffer Modules
Floppy Disk Controller	CPU/SBC Board	General Purpose I/O Board
PROM Programmer	4-16 Slot Card Cages	PROM/ROM Board

NEW LOWER PRICES AND A CASH DISCOUNT* TO BOOT!

A65/40-16 (16K RAM)	$1225
A65/40-32 (32K RAM)	$1295
A65/40-A (Assembler)	$ 85
A65/40-B (BASIC)	$ 65

A65-1 (1K RAM)	$420
A65-4 (4K RAM)	$445
A65-4B (4K RAM w/BASIC)	$495
A65-PS (PASCAL)	$100
A65-F (FORTH)	$ 65
A65-A (Assembler)	$ 35

Mail Order to:

Educational Computer Division
EXCERT INCORPORATED

- SALES
- SERVICE
- INSTALLATION
- CONSULTING

P.O Box 8600
White Bear Lake, MN 55110
(612) 426-4114

Higher quantities quoted upon request, COD's accepted, shipping will be added. *Deduct 5% cash discount on prepaid orders. Minnesota residents add 5% sales tax. Prices subject to change without notice.

Insurance Inventory

Robert W Baker
Atco, NJ

This program – versions for Microsoft and Atari BASICs – was written to provide an easy means of maintaining an inventory of personal possessions for insurance or other related purposes. Information is stored on cassette tape for later retrieval and easy, compact storage such as in a safety deposit box.

Running the program is quite simple: to create a new data file simply select that mode and answer the questions concerning the item description, make, model, serial number of other identifying markings, date acquired, and original value. Typing RETURN for any question will automatically enter a question mark for that entry. When all questions are entered, the entire entry will be displayed and you will be asked if it is correct before it is actually written in the data file.

Typing "D" (for DONE) for any entry will abort that entire item entry, close the output file, and return to the program command mode. Typing "E" (for ERROR) will abort the entire item entry and restart it with the first question. Be careful when entering new items into the data file, do not use commas "," or colons ":" to separate words within a description, etc. since BASIC thinks you may be entering more than one string. Use dashes or some other graphic character and play it safe. Avoid using quotes as well, for similar reasons.

A Full Update Capability

To read an already created data file, insert the tape and select that program mode. Three items will be displayed at a time, with all information. Hitting any key except "D" (or RUN/STOP) will display the next three entries. Typing "D" will terminate the read mode, close the input file, and return to the program command mode.

Other program modes are provided to copy or edit the data files produced by this program. The edit mode allows copying or deleting individual entries. You can insert new items at any point. Also, a search feature is included to copy all items until a specific item is found. However, with tape data files, two tape drives are required for these functions for obvious reasons.

All program modes provide file and/or drive selection for ease of use. A default file name of INVENTORY DATA will be generated unless you enter a specific filename. If you should have a large number of items to catalog, you may want to use separate data files for each room, for items acquired each year, specific collections, etc. Program use should be self-evident through prompting instructions displayed by the program. At present, the program does not provide a print option since it was designed for storage of large amounts of personal data. It should be rather easy to add a printing feature if you really think it's necessary.

Describing the actual program is rather difficult since portions of the program are used for every mode. The program flow changes depending on the selected mode and various control flags that are set dynamically during program execution. The program was originally developed on an 8K PET, so I tried to maximize memory usage, allowing more room for the data being generated and used.

Take my word for it, the program does work. If you want to avoid typing in the program, send $2 and an SASE and tape, for a copy on cassette tape. [This is for PET/CBM users only.] For anyone with a 2040 disk, I have another version of this program that uses sequential disk data files. This makes the EDIT mode much more useful. If you do send for a copy on tape, be sure to indicate which version you want. Also, please send all requests directly to me.

Robert Baker
15 Windsor Drive
Atco, NJ 08004

Program 1: Microsoft Version

```
80 POKE 59468,12 :PRINT"{CLEAR}     HOUS
     EHOLD INVENTORY PROGRAM" :GOSUB ~
     1340
90 PRINT"DESIRED PROGRAM MODE:
100 PRINT :PRINT" 0 = DONE
110 PRINT" 1 = READ DATA
120 PRINT" 2 = WRITE NEW DATA FILE"
130 PRINT" 3 = COPY DATA FILE @.
140 PRINT" 4 = EDIT DATA FILE @1@ REQ'S 2
      TAPES
150 GOSUB 1340 :PRINT :PRINT"MODE ?";
160 GOSUB 1440 :IF R$="0" THEN END
170 R=VAL(R$) :IF R<1 OR R>4 THEN 160
180 Z=R :ON R GOTO 400,190,390,390
190 GOSUB 1310 :IF Z>2 THEN T=2 :T$="2" :
     GOTO 220
200 PRINT :INPUT"OUTPUT TO TAPE DRIVE# (1
      OR 2)   2{03 LEFT}";T$
210 T=VAL(T$) :IF T<1 OR T>2 THEN 80
220 PRINT :PRINT"PUT OUTPUT TAPE IN DRIVE
      #";T$ :GOSUB 1390
230 IF F$<>"" THEN 260
240 F$="INVENTORY DATA" :IF Z>2 THEN F$=X
     $
250 PRINT :PRINT"DEFAULT FILENAME = ";F$
```

LETTER PERFECT

T.M. LJK

WORD PROCESSING

ATARI 400/800 — APPLE II & II+

EASY TO USE — Letter Perfect is a single load easy to use program. It is a menu driven, character orientated processor with the user in mind. FAST machine language operation, ability to send control codes within the body of the program, mnemonics that make sense, and a full printed page of buffer space for text editing are but a few features. Screen Format allows you to preview printed text. Indented margins are allowed. Data Base Merge with **DATA PERFECT** by LJK, form letters, accounting files and mailing labels only with MAIL MERGE/UTILITY by LJK. **FEATURES** — Proportional/Incremental spacing * Right Justification * File Merging * Block movement * Headers * Footers * Print Multiple Copies * Auto Page Numbering * Scroll forward/backward * Search and Replaces * Full cursor control * Underlining * Boldface * Superscripts * Subscripts * Auto page numbering * Insert character/line * Delete character/line * Centering * Horizontal tabs/changeable * Multifunction format line (line spacing — left margin — page width — lines/page — change fonts — top/bot margin adjust) **MUCH MORE! $149.95**

ATARI VERSION 2.0 #2001

Compatible with Atari DOS. Uses proportional font, right justified with Atari 825/Centronics 737, 739 printers. Uses EPSON MX* Series + Graftrax/italicized font. Can mix type fonts on same page; mix boldface and enhanced font in same line with justification. Can be used with 16K Atari/400.*

"Compared to the price of many other word processors, this package is a steal. It does everything the advertisement claims and more. On top of this the software is very easy to use." **A.N.A.L.O.G. MAGAZINE**

APPLE VERSION 5.0 #1001

DOS 3.3 compatible — Use 40 or 80 column interchangeably (Smarterm — ALS; Videoterm-Videx; Full View 80 — Bit 3 Inc.; Vision 80 — Vista; Sup-R-Term — M&R Ent.) Reconfigurable at any time for different video, printer, or interface. **USE HAYES MICROMODEM II*** LCA necessary if no 80 column board, need at least 24 K of memory. Files saved as either Text or Binary. Shift key modification allowed. Data Base Merge compatible with **DATA PERFECT*** by LJK.

"For $150, Letter Perfect offers the type of software that can provide quality word processing on inexpensive micro-computer systems at a competitive price." **INFOWORLD**

DATA PERFECT T.M. LJK

APPLE & ATARI
DATA BASE MANAGEMENT — **$99.95**

Complete Data Base System. User orientated for easy and fast operation. 100% Assembly language. Easy to use. You may create your own screen mask for your needs. Searches and Sorts allowed, Configurable to use with any of the 80 column boards of Letter Perfect word processing, or use 40 column Apple video. Lower case supported in 40 column video. Utility enables user to convert standard files to Data Perfect format. Complete report generation capability. **Much More!**

EDIT 6502 T.M. LJK

This is a coresident — two pass **ASSEMBLER, DIS-ASSEMBLER, TEXT EDITOR, and MACHINE LANGUAGE MONITOR.** Editing is both character and line oriented. Disassemblies create editable source files with ability to use predefined labels. Complete control with 41 commands, 5 disassembly modes, 24 monitor commands including step, trace, and read/write disk. Twenty pseudo opcodes, allows linked assemblies, software stacking (single and multiple page) plus complete printer control, i.e. paganation, titles and tab setting. User can move source, object and symbol table anywhere in memory. Feel as if you never left the environment of **BASIC**. Use any of the 80 column boards as supported by **LETTER PERFECT**, Lower Case optional with **LCG**.

LJK DISK UTILITY — APPLE $29.95

This menu driven program allows the user to manipulate a variety of different file types. Binary, Text, and Source files may be easily converted into each other. The program may be used with **APPLESOFT*, VISCALC***, and other programs. These program files may be readily adapted for multiple use including editing with **LETTER PERFECT** word processings.

MAIL MERGE/UTILITY — APPLE & ATARI — $29.95

This menu driven program combined with **LETTER PERFECT** allows user to generate form letters and print mailing labels. With the Atari, you may **CONVERT ATARI DOS FILES**, or Visicalc files compatible for editing with **LETTER PERFECT**. Utility creates Data Base files for Letter Perfect.

LOWER CASE CHARACTER GENERATOR — $34.95

```
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFG
HIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
pqrstuvwxyz{|}~
```

Lower Case Character Generator for the Rev. 7, Apple II or II+ computers. When installed, this Eprom will generate lower case characters to the video screen. Lower case characters set has two dot true descenders. Installation instruction included. Manual includes listing of software for full support and complete instructions for shift key modification. Compatible with **LETTER PERFECT**.

LJK™ COMPUTER BASED SOFTWARE — ENTERPRISES Inc

LJK ENTERPRISES INC.
P.O. Box 10827
St. Louis, MO 63129
(314) 846-6124

DEALER INQUIRES INVITED

*Trademarks of: Apple Computer — Atari Computer — Epson America — Hayes Microcomputers — Personal Software — Videx — Bit 3 Inc. — M&R Ent. — Advanced Logic Systems — Vista Computers

```
260 PRINT :PRINT"IS OUTPUT TAPE POSITIONE
    D"; :GOSUB 1360 :IF R$="Y" THEN ~
    340
270 PRINT :INPUT"NUMBER OF FILES TO SKIP ~
      "{03 LEFT}";R$:R=VAL(R$):IF R<
    1 THEN 80
280 GOSUB 1310 :PRINT :PRINT"POSITIONING ~
    OUTPUT TAPE
290 OPEN 2,T,0
300 GOSUB 1290 :IF C=0 THEN 300
310 CLOSE 2 :IF C=2 THEN 580
320 R=R-1 :IF R>0 THEN 290
330 PRINT :PRINT"OUTPUT TAPE IS NOW POSIT
    IONED" :GOSUB 1400 :PRINT
340 OPEN 2,T,1,F$ :IF Z=3 THEN 600
350 IF Z=4 THEN 650
360 GOSUB 940 :IF C>0 THEN GOSUB 1180 :GO
    TO 360
370 CLOSE 2 :PRINT"{CLEAR}{REV}END OF MOD
    E #2{OFF}  DONE WRITING DATA FIL
    E
380 GOSUB 1340 :GOTO 590
390 GOSUB 1310 :T=1 :T$="1" :GOTO 430
400 PRINT"{CLEAR}{REV}MODE #1{OFF}  READ ~
    DATA FILE FROM TAPE" :GOSUB 1340

410 PRINT :INPUT"INPUT FROM TAPE DRIVE# (
    1 OR 2)   1{03 LEFT}";T$ :T=VAL(
    T$)
420 IF T<1 OR T>2 THEN 80
430 PRINT"{DOWN}PUT INPUT TAPE IN DRIVE #
    ";T$ :GOSUB 1390
440 IF F$=""" THEN F$="" :PRINT :PRINT"RE
    ADING NEXT FILE ON TAPE
450 OPEN 1,T,0,F$ :X$=""
460 IF Z>2 THEN 190
470 GOSUB 1220 :IF C>1 THEN 550
480 GOSUB 1140 :IF C>0 THEN 560
490 GOSUB 1220 :IF C>1 THEN 560
500 GOSUB 1150 :IF C>0 THEN 560
510 GOSUB 1220 :IF C>1 THEN 560
520 GOSUB 1150 :IF C>0 THEN 560
530 GOSUB 1350
540 GOSUB 1460 :IF R$<>"D" THEN 470
550 PRINT"{CLEAR}{REV}END OF MODE #1{OFF}
      DONE READING DATA FILE" :PRINT

560 CLOSE 1 :GOSUB 1350
570 IF C=1 THEN PRINT"END OF DATA FILE!
580 IF C>1 THEN PRINT"TAPE READ ERROR ( S
    TATUS ="ST")
590 GOSUB 1400 :GOTO 80
600 I9$="" :GOSUB 1310 :PRINT"{REV}PLEASE
      WAIT{OFF}  ******  COPYING DATA
      FILE!
610 GOSUB 1220 :IF C>1 THEN 860
620 IF Z=4 THEN IF LEFT$(I$,LEN(I9$))=I9$
      THEN GOSUB 1310 :GOTO 660
630 GOSUB 1180 :IF C=1 THEN 860
640 IF Z=3 OR I9$<>"" THEN 610
650 GOSUB 1220 :IF C>1 THEN 860
660 GOSUB 1310 :GOSUB 1150 :GOSUB 1340 :P
    RINT"DESIRED ACTION:" :PRINT
670 PRINT"  1 = COPY THIS ITEM, NO CHANGE

680 PRINT"  2 = DELETE THIS ITEM
690 PRINT"  3 = INSERT ITEMS BEFORE THIS ~
    ONE
700 PRINT"  4 = SEARCH & COPY TILL ITEM F
    OUND" :PRINT
710 PRINT"ACTION ? ";
720 GOSUB 1440 :R=VAL(R$) :IF R<1 OR R>4 ~
    THEN 720
730 PRINT R$
740 PRINT"OK" :I9$="" :ON R GOTO 630,750,
    770,800
750 IF C=1 THEN 860
760 GOTO 650
770 I9$=I$ :W9$=W$ :M9$=M$ :S9$=S$ :D9$=D
    $ :V9$=V$ :C9=C
780 GOSUB 940 :IF C>0 THEN GOSUB 1180 :GO
    TO 780
790 I$=I9$ :W$=W9$ :M$=M9$ :S$=S9$ :D$=D9
    $ :V$=V9$ :C=C9 :GOTO 660
800 GOSUB 1310 :PRINT"ALL ENTRIES WILL BE
      COPIED UNTILL
810 PRINT :PRINT"DESIRED ITEM IS FOUND;
820 PRINT"{02 DOWN}ENTER ITEM TO SEARCH F
    OR:
830 INPUT"     "{03 LEFT}";I9$
840 IF I9$=""" THEN I9$="" :PRINT"{03 DOW
    DOWN}SEARCH ABORTED" :GOTO 660
850 PRINT"{03 DOWN}SEARCHING" :GOTO 620
860 IF Z=3 THEN 570
870 GOSUB 1310 :IF C>1 THEN 580
880 PRINT"END OF INPUT FILE!
890 PRINT :PRINT"DO YOU WANT TO ADD ANY E
    NTRIES TO THE
900 PRINT :PRINT"END OF THE DATA FILE";
910 GOSUB 1360 :IF R$="N" THEN 590
920 GOSUB 940 :IF C>0 THEN GOSUB 1180 :GO
    TO 920
930 GOSUB 1310 :GOTO 590
940 C=0 :PRINT"{CLEAR}ENTER ITEM INFORMAT
    ION:" :PRINT
950 PRINT"D = DONE ENTERING DATA
960 PRINT"E = ERROR, RESTART ENTIRE ITEM
970 PRINT :PRINT"DO NOT USE ',' OR ':' WI
    THIN THE DATA
980 PRINT :PRINT"PRESS {REV}RETURN{OFF} A
    FTER EACH ENTRY
990 GOSUB 1340 :INPUT"{REV}ITEM{OFF}    ?{
    04 LEFT}";I$ :IF I$="E" THEN 940

1000 IF I$="D" THEN RETURN
1010 INPUT"{REV}MAKE{OFF}    ?{03 LEFT}";W$
      :IF W$="E" THEN 940
1020 IF W$="D" THEN RETURN
1030 INPUT"{REV}MODEL{OFF}    ?{03 LEFT}";M
     $ :IF M$="E" THEN 940
1040 IF M$="D" THEN RETURN
1050 INPUT"{REV}SERIAL#/ID{OFF}    ?{03 LEF
     LEFT}";S$ :IF S$="E" THEN 940
1060 IF S$="D THEN RETURN
1070 INPUT"{REV}DATE ACQ'D{OFF} (MONTH/DAY
     /YEAR)    ?{03 LEFT}";D$ :IF D$="
     E" THEN 940
1080 D$=LEFT$(D$,8) :IF D$="D" THEN RETURN

1090 INPUT"{REV}$VALUE{OFF}    ?{03 LEFT}";
     V$ :IF V$="E" THEN 940
1100 IF V$="D" THEN RETURN
```

THE PROGRAM STORE

CALL TOLL FREE 800 424-2738

Save $$ on Popular Programs for Atari

LIMITED TIME OFFER

Now through March 15, take advantage of our price reduction on these popular programs

Crush, Crumble and Chomp!

From Epyx
It's a monster movie, and you are the monster! You can be The Glob, Kraken, Mantra, Mechismo, Arachnis, or Goshilla -- or even design your own "custom" monster (disk version only). This hilarious action game is loaded with graphics and sound as you practice your villany. With 6 monsters, 4 cities, and 5 game objectives, you get a choice of more than 100 possible scenarios. A monster's life is not all carnivorous crunching, though: The combined resources of the police, science, and armed forces are bent on your destruction.

32K disk...$29.95

THRU MARCH 15 ONLY: Pay Just $25.45!

FILE MANAGER 800

From Synapse Software
An extremely powerful and versatile database manager for use in both professional and personal applications. You define the format of the records to be filed and FILE MANAGER 800 gives you full control over sorting, searches, and retrieval. You can store up to 1000 records on each disk and the program will allow up to four drives. Access to any record takes less than 1.5 seconds, and most commands can be entered with self-prompting single keystrokes.

Atari (40K disk)...$94.95

THRU MARCH 15 ONLY: Pay Just $80.70!

PERSONAL FINANCE

From Computari
A comprehensive personal/small business budget organizer that's more than just a checkbook balancer. Works for both check and cash and will allow up to 100 transactions per month. One check can be divided into several categories to avoid the problem usually associated with charge card and department store checks. The report printer works with most popular printers and will give printouts of Tabulations, Check Search, and Check Reconciliation categories. Get a grasp on your finances, order PERSONAL FINANCE now.

32K disk...$49.95

THRU MARCH 15 ONLY: Pay Just $42.45!

Programs for ATARI 400/800 Unless Otherwise Noted

A Warner Communications Company

GHOST HUNTER

From Arcade Plus
The exciting haunted house/maze game where you must get the ghosts before they get you! When you eat the pulsating "energizers," the ghosts start pulsating and become vulnerable to your attack. 51 different game variations keep things interesting. Play alone or in the "head-to-head" mode.

16K tape $29.95 16K disk $34.95

By John Harris from On-Line
If you like arcade games with action, you'll love JAWBREAKER. Guide your chompers through a candystore maze, eating "wifesavers" as you go. You are pursued by variously colored "smilies," determined to knock your teeth out. Eat a jawbreaker and the "smilies" turn to blue "frownies," and you can eat them! It's one of the best examples of Atari graphics and sound we've seen -- you'll love it!

16K disk...$29.95

SAMMY the Sea Serpent

From PDI
A great way to introduce young children (ages 4 and up) to your Atari computer. Speech, music, motion and color hold youngsters attention. As the story is read, the child is asked to use the joystick to move Sammy about. As Sammy's tasks are completed, the storyteller goes on to new ones. Side two has additional games for the child to play with Sammy.

16K tape...$16.95

OUTLAW/HOWITZER

From Atari Prog. Exchange
Two super arcade games in one program! OUTLAW transports you to the old west, where you are engaged in a shootout on the open plains. You can hide behind the cacti that are scattered about, but if you get shot 10 times it's a one-way ticket to Boot Hill, Pardner.

You command a tank in HOWITZER. You're facing an opposing tank across a river and must fire 10 hits to win. Both games can be played against the computer or a friend, and there are several handicap options available.

16K tape...$14.95 24K disk...$17.95

SoftPorn Adventure

By Charles Benton from On-Line
You're in a sleazy bar in the notorious town of "Las Vagueness." With $1000 in your pocket, you set out to fulfill your erotic fantasy of seducing 3 women before you leave town. The casino, with its blackjack & slot machines, is the only source of money. The women and other bizarre pleasures might be found anywhere. NOTE: R-RATED PROGRAM, NOT FOR MINORS.

40K disk...$29.95

ASSEMBLER BOOK

ATARI ASSEMBLER BOOK

By Don & Kurt Inman
While the ATARI ASSEMBLER CARTRIDGE comes with an operating manual, it assumes that you already know assembly language. If you're new to the Atari or its 6502 processor, this book is a must. The Inmans guide you through the rudiments of this fascinating type of programing in clear, easy steps. Includes full listing and description of 6502 mnemonics and addressing modes.

$12.95

Some Common BASIC Programs Atari® Edition

By Poole, Borchers & Cook from Osborne
From "Alphabetizing" to "Withholding Taxes" -- this book contains 75 program listings for solutions to common (and not-so-common) problems. Finance, math, statistics, projections, and household utilities are included. With complete listings, sample runs, and descriptions, you needn't be a programmer to benefit from this book.

200 pages...$14.95

Visit our other stores: Seven Corners Center · Falls Church, VA & W. Bell Plaza · 6600 Security Blvd. · Baltimore, MD

VISA master charge

TO ORDER CALL TOLL FREE 800 424-2738

For information Call (202) 363-9797

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.

THE PROGRAM STORE
4200 Wisconsin Avenue NW, Dept.U02 Box 9609
Washington, D.C. 20016

```
1110 GOSUB 1140 :GOSUB 1340
1120 PRINT"IS THIS ENTRY CORRECT"; :GOSUB ~
     1360 :IF R$="N" THEN 940
1130 C=1 :RETURN
1140 PRINT"{CLEAR}";
1150 PRINT"{REV}ITEM:{OFF} ";I$ :PRINT"{RE
     REV}MAKE:{OFF} ";W$
1160 PRINT"{REV}MODEL:{OFF} ";M$ :PRINT"{R
     REV}SERIAL#/ID:{OFF} ";S$
1170 PRINT"{REV}DATE ACQ'D:{OFF} "D$;TAB(2
     2);"{REV}VALUE:{OFF} $";V$ :PRIN
     T :RETURN
1180 X$=I$ :GOSUB 1200 :X$=W$ :GOSUB 1200 ~
     :X$=M$ :GOSUB 1200
1190 X$=S$ :GOSUB 1200 :X$=D$ :GOSUB 1200 ~
     :X$=V$
1200 PRINT#2,X$
1210 RETURN
1220 GOSUB 1290 :I$=X$ :IF C>0 THEN RETURN

1230 GOSUB 1290 :W$=X$ :IF C>0 THEN RETURN

1240 GOSUB 1290 :M$=X$ :IF C>0 THEN RETURN

1250 GOSUB 1290 :S$=X$ :IF C>0 THEN RETURN

1260 GOSUB 1290 :D$=X$ :IF C>0 THEN RETURN

1270 GOSUB 1290 :V$=X$ :IF C=2 THEN C=1
1280 RETURN
1290 C=0 :INPUT#1,X$ :IF ST>0 THEN C=3 :IF
      ST=64 THEN C=2
1300 RETURN
1310 IF Z=2 THEN PRINT"{CLEAR}{REV}MODE #2
     {OFF}  WRITE NEW DATA FILE ON TA
     PE
1320 IF Z=3 THEN PRINT"{CLEAR}{REV}MODE #3
     {OFF}  COPY DATA FILE, REQ'S 2 T
     APES
1330 IF Z=4 THEN PRINT"{CLEAR}{REV}MODE #4
     {OFF}  EDIT DATA FILE, REQ'S 2 T
     APES
1340 PRINT
1350 PRINT"CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
     CCCCCCCC" :PRINT :RETURN
1360 PRINT" (Y/N) ? ";
1370 GOSUB 1440 :IF R$<>"Y" AND R$<>"N" TH
     EN 1370
1380 PRINT R$ :RETURN
1390 INPUT"{DOWN}FILENAME < {REV}RETURN{OF
     OFF} >    "{03 LEFT}";F$ :RETURN
1400 CLOSE 1 :CLOSE 2
1410 IF Z>2 THEN PRINT :PRINT"DEPRESS {REV
     REV}STOP{OFF} ON BOTH TAPE DRIVE
     S" :GOTO 1430
1420 PRINT :PRINT"DEPRESS {REV}STOP{OFF} O
     N TAPE #"T$
1430 PRINT"{DOWN}HIT ANY KEY WHEN READY TO
      CONTINUE"; :GOTO 1470
1440 GET R$ :IF R$="" THEN 1440
1450 RETURN
1460 PRINT :PRINT"HIT ANY KEY TO CONTINUE,
      D=DONE";
1470 GOSUB 1440 :PRINT :PRINT"OK" :RETURN
```

Program 2: Atari Version

```
10 REM INSURANCE INVENTORY
20 REM ROBERT W. BAKER ATCO, NJ
30 REM ATARI VERSION (C) 1981
40 REM SMALL SYSTEMS SERVICES, INC.
50 REM
60 OPEN #2,4,0,"K:"
70 DIM T$(1),X$(80),I$(80),W$(80),M$(80)
,S$(80),D$(8),V$(20)
80 GRAPHICS 0:? "    INSURANCE INVENTORY
 PROGRAM  "
90 PRINT "Desired Program mode:"
100 ? :? "0 = Done"
110 ? "1 = Read Data"
120 ? "2 = Write New Data File"
150 GOSUB 1340:? :? "Mode? ";
160 GOSUB 1440:IF R=48 THEN END
170 R=R-48:IF R<1 OR R>2 THEN 160
180 Z=R:ON R GOTO 400,190
190 ? "{CLEAR}| MODE #2 | WRITE NEW DATA
 FILE"
200 TRAP 200:? "{DOWN}Number of files to
 skip";:INPUT R:TRAP 40000:IF R<1 THEN 3
40
210 ? "{DOWN}Put tape in drive, press PL
AY"
220 ? "then press |RETURN|."
230 IF PEEK(764)=255 THEN 230
270 ? "{CLEAR}| MODE #2 | WRITE NEW DATA
 FILE"
280 ? "{DOWN}Positioning output tape..."

290 TRAP 310:OPEN #1,4,0,"C:"
300 GOSUB 1290:IF C=0 THEN 300
310 CLOSE #1:IF C=2 THEN 580
320 R=R-1:IF R>0 THEN 290
330 ? :? "Output tape is now positioned.
{DOWN}":GOSUB 1400:?
340 ? "Press PLAY & RECORD, press |RETUR
N|."
350 OPEN #1,8,0,"C:"
355 FOR I=1 TO 128:PUT #1,32:NEXT I
360 GOSUB 940:IF C>0 THEN GOSUB 1180:GOT
O 360
370 CLOSE #1:? "{CLEAR}|END OF MODE #2|
 Done writing data file"
380 GOSUB 1340:GOTO 590
400 ? "{CLEAR}|MODE #1|  READ DATA FILE
FROM TAPE":GOSUB 1340
430 ? "{DOWN}Put input tape in drive, pr
ess |PLAY|,  then press |RETURN|."
450 TRAP 560:OPEN #1,4,0,"C:":X$="":FOR
I=1 TO 128:GET #1,R:NEXT I:TRAP 40000
460 ? "{DOWN}Reading next file on tape."

470 GOSUB 1220:IF C>1 THEN 550
```

```
480 GOSUB 1140:IF C>0 THEN 560
490 GOSUB 1220:IF C>1 THEN 560
500 GOSUB 1150:IF C>0 THEN 560
510 GOSUB 1220:IF C>1 THEN 560
520 GOSUB 1150:IF C>0 THEN 560
530 GOSUB 1350
540 GOSUB 1460:IF R<>ASC("D") THEN 470
550 ? "{CLEAR}|END OF MODE #1| DONE READ
ING DATA FILE":?
560 CLOSE #1:GOSUB 1350
570 IF C=2 THEN ? "End of data file!":C=
1
580 IF C>1 THEN ? "TAPE READ ERROR-";PEE
K(195)
590 GOSUB 1400:GOTO 80
940 C=0:? "{CLEAR}Enter item information
:":?
950 ? "D = Done entering data"
960 ? "E = Error, restart entire item"
980 ? :? "Press |RETURN| after each entr
y."
990 GOSUB 1340:? "|ITEM|";:INPUT I$:IF I
$="E" THEN 940
1000 IF I$="D" THEN RETURN
1010 ? "|MAKE|";:INPUT W$:IF W$="E" THEN
 940
1020 IF W$="D" THEN RETURN
1030 ? "|MODEL|";:INPUT M$:IF M$="E" THE
N 940
1040 IF M$="D" THEN RETURN
1050 ? "|SERIAL#/ID|";:INPUT S$:IF S$="E
" THEN 940
1060 IF S$="D" THEN RETURN
1070 ? "|DATE ACQ'D| (MONTH/DAY/YEAR) ";
:INPUT D$:IF D$="E" THEN 940
1080 IF D$="D" THEN RETURN
1090 ? "|$VALUE| ";:INPUT V$:IF V$="E" T
HEN 940
1100 IF V$="D" THEN RETURN
1110 GOSUB 1140:GOSUB 1340
1120 ? "Is this entry correct?";:GOSUB 1
360:IF R=ASC("N") THEN 940
1130 C=1:RETURN
1140 ? "{CLEAR}";
1150 ? "|ITEM|: ";I$:? "|MAKE|: ";W$
1160 ? "|MODEL|: ";M$:? "|SERIAL#/ID|: "
;S$
1170 ? "|DATA ACQ'D|: ";D$,"|VALUE|: ";V
$:? :RETURN
1180 X$=I$:GOSUB 1200:X$=W$:GOSUB 1200:X
$=M$:GOSUB 1200
1190 X$=S$:GOSUB 1200:X$=D$:GOSUB 1200:X
$=V$
1200 PRINT #1;X$
1210 RETURN
1220 GOSUB 1290:I$=X$:IF C>0 THEN RETURN
1230 GOSUB 1290:W$=X$:IF C>0 THEN RETURN
1240 GOSUB 1290:M$=X$:IF C>0 THEN RETURN
1250 GOSUB 1290:S$=X$:IF C>0 THEN RETURN
1260 GOSUB 1290:D$=X$:IF C>0 THEN RETURN
1270 GOSUB 1290:V$=X$:IF C=2 THEN C=1:ST
OP
1280 RETURN
1290 TRAP 1300:C=0:INPUT #1,X$:TRAP 4000
0:RETURN
1300 C=3:IF PEEK(195)=136 THEN C=2:RETUR
N
1340 ?
1350 ? "{38 R}":RETURN
1360 ? " (Y/N) ? ";
1370 GOSUB 1440:IF R<>ASC("Y") AND R<>AS
C("N") THEN 1370
1380 ? CHR$(R):RETURN
1400 CLOSE #1:? "Depress |STOP| on tape
drive."
1410 ? :? "Press any key to continue..."
;
1440 GET #2,R:RETURN
1460 ? :? "HIT ANY KEY TO CONTINUE, D=DO
NE"
1470 GOSUB 1440:? "OK":RETURN
```

©

NEECO

WHY BUY FROM THE BEST?

Service... Support...
Software...

commodore

16K B (16K RAM-40 Column) - Lim. Qty	$ 995
32K B (32K RAM-40 Clm.) - Lim. Qty	$1295
4016 (16K RAM 4.0 Basic-40 Clm.)	$ 995
4032 (32K RAM 4.0 Basic-40 Clm.)	$1295
8032 (32K RAM 4.0 Basic-80 Clm.)	$1495
8050 Dual Disk (1 Meg Storage)	$1795
4040 Dual Disk (343K Storage)	$1295
8010 IEEE Modem	$ 280
C2N Cassette Drive	$ 75
CBM - IEEE Interface Cable	$ 40
IEEE - IEEE Interface Cable	$ 50
VIC 20 Home/Personal Computer	$ 295

CALL NEECO FOR ANY OF YOUR COMMODORE COMPUTER NEEDS

NEC SPINWRITER PRINTERS

5530 (Parallel)	$3055
5510 (Serial)	$3055
5520 (KSR-Serial)	$3415
Tractor Option	$ 225

EPSON PRINTERS

MX-80 PRINTER	$ 645
MX-80 FT	$ 745
MX-100	$ 945
MX-70	$ 459
INTERFACE CARDS	
8141 (RS-232)	$ 75
8150 (2K Buffered RS-232)	$ 150
8161 (IEEE 488)	$ 55
8131 (Apple Card)	$ 85
8230 (Apple Card)	$ 25
8220 (TRS-80 Cable)	$ 35

APPLE

16K APPLE II+	$1330
32K APPLE II+	$1430
48K APPLE II+	$1530
APPLE DISK w/3.3 DOS	$ 650
APPLE DRIVE Only	$ 490
APPLE III 128K - In Stock! w/Monitor + Info Analystpak	$4740

DIABLO 630 PRINTER

DIABLO 630 - Serial - RS-232	$2710
Tractor Option	$ 250

AMDEK MONITORS

Video 100 12" B+W	$ 179
Video 300 12" Green	$ 249
Color I 13" Low Res	$ 449
Color II 13" High Res	$ 999

INTERTEC COMPUTERS

64K Superbrain (360 Disk Storage), CP/M™	$3495
64K QD Superbrain (700K Disk Storage), CP/M™	$3995

*CP/M is a registered trademark of Digital Research.

ATARI COMPUTERS

Atari 400 (16K RAM)	$ 399
Atari 800 (32K RAM) - good thru 8/31	$1080
Atari 410 RECORDER	$ 89.95
Atari 810 DISK DRIVE	$ 599.95

NEECO carries all available ATARI Software and Peripherals.

PROFESSIONAL SOFTWARE

WordPro 1 8K	$ 29.95
WordPro 3 (40 Clm.)16K	$ 199.95
WordPro 3+	$ 295
WordPro 4 (80 Clm.) 32K	$ 375
WordPro 4+	$ 450

JUST A SAMPLE OF THE MANY PRODUCTS WE CARRY, CALL US FOR OUR NEW 60-PAGE CATALOG.
WE WILL MATCH SOME ADVERTISED PRICES ON CERTAIN PRODUCTS LISTED UNDER SIMILAR "IN STOCK" CONDITIONS.

NEECO
679 HIGHLAND AVE.
NEEDHAM, MA 02194

(617) 449-1760
Telex: 951021

MON-FRI 9:00 - 5:00
MasterCharge and VISA Accepted

Creating A Simple Word Processor

Steve Gradijan
Carrollton, TX

Editor's Note: With minor adjustments, this simple (but effective) word processor will work on Apple, OSI, – any Microsoft BASIC. Described here for the PET/CBM, the author points out which lines to change for other machines. The program is well documented to permit easy adjustment to a variety of printer, disk/tape, etc. configurations. —RTM

Arnie Lee's LED a Line-oriented Text Editor described in **COMPUTE!** #9 can become a moderately sophisticated word processor with the addition of a few lines. Line 5 enables upper and lower case features of the PET; line 8075 disengages these features and returns the PET to normal mode. Minor modification to line 9050 eliminates the printing of the LED's line oriented identification numbers and allows Commodore printers to print both upper and lower case. The addition of the control character to the string referred to in this line, however, uses one position of the 80 character string limiting the text part of the string to 79 characters, including spaces. Thus, Mr. Lee's 80 character string is shortened in line 10055.

Adding Versatility

Additional program lines give the word processor more versatility. Tab functions or line indentation and simulated line feed are accomplished by lines 145, 146, and 10045. Lines 9025 and 9055 provide the option for double spaced print of copy to the printer.

Lines 235 and 21000 to 21100 set the margins. If the margins are not set at the beginning of a typing session, the program defaults to a 79 character line i.e. no margins. Lines 500, 520, 530, and 570 are modified to accomodate the additional command "s," set margins.

A "bell" is provided to prompt you when only five spaces are left in a line (lines 22000 to 22040). It makes use of the "CB-2 sound" provided at the user port and requires connection of a suitable amplifier/speaker to the PET.

Lines 5 and 8075 are not necessary with the Commodore CBM's. The upper-lower case and bell functions use POKE statements not compatible with other computers. However, all other modifications should be usable with machines using Microsoft BASIC.

Options And Commands

Additional commands now available include "s" which allows setting margins. "n" establishes a 79 character line and no margins; "s" creates five character wide left and right margins and a 69 character wide field; "m" increases the size of the margins to ten characters and the "l" to fifteen characters in width. "o" allows creation of your own margins, both left and right. You are asked to specify the length of both the left and right margins. If you forget to set the margins at the beginning of the program the margins will default leaving you with a 79 character line. Once set, margins may later be lengthened, but never shortened!

The sub-commands "@" and "@" + RETURN provide tab functions and line skipping. The sub-command "@" adds five spaces to the text string and is useful as a tab or an indent. It may be used at any time while in a)ppend, i)ndent, or r)eplace functions. To skip a line of text, type "@" followed by the RETURN key.

The option to print the text either single or double spaced is given after requesting the print function p)rint.

The PET keyboard will behave like a normal typewriter after modifying the LED program. The shift key will provide upper case. All punctuation and designators supplied by the PET character set are available except @, which is used for tab functions. Quotation marks are permitted, but they look a little strange when first seen on the CRT screen of the PET. Ignore the funny appearance of anything that you enclose in quotes and depend upon the "bell" to determine the end of the line. Everything within quotes will appear normal when later listed, printed, or saved!

Delete lines 9070 through 9090 from Arnie Lee's original program and you are ready to type a letter, an order, an article for **COMPUTE!** or whatever.

```
0:LIST
 0 REM LINE EDITOR (C)1980 ABACUS ~
    SOFTWARE
 3 REM   FROM   FEBRUARY 1981 COMP
    UTE
 4 REM   MODIFIED BY STEVE GRADIJA
    N, CARROLLTON, TEXAS
 5 POKE59468,14:REM  ENABLE UPPER ~
    & LOWER CASE
 10 DIMT$(500):REM BUFFER SPACE
 20 L$="":REM  CURENT LINE
 30 LL=1:REM LAST LINE #
```

JINSAM

Data Manager selected by NASA, Kennedy Space Center

SAVE TIME! SAVE MONEY!
JINSAM saves taxpayers
100's of thousands of $

OFFICIALLY APPROVED BY COMMODORE

"Much more powerful than you can imagine!"
Robert Baker
Kilobaud Microcomputing

JINSAM is an integrated system. It makes it easy to use your information to its fullest. No more will hundreds of valuable hours be spent searching or analyzing needed information nor re-entering information for various reports.

JINSAM transforms your desk-top computer into the "state of the art" data processing machine with features and accessories found nowhere, even at 10 times the price. NASA, Kennedy Space Center selected JINSAM 8.0 and saved approximately $95,000 over other software/hardware costs. Riley County, Kansas also selected JINSAM 8.0 and saved approximately $90,000 over other software/hardware costs.

JINSAM is designed for you. It is forgiving. It has help commands for every option, available at the touch of a button. The amount of information you store, its structure and/or your hardware can change but your data won't have to be re-entered. Recovery utilites are included even for catastrophes, security passwords are built in for privacy, simple editing and entry includes auto recall, and deleting records is easy and the space is reclaimed. JINSAM includes TWO FREE accessories for reports and labels. You have unlimited report formats with summing and lined up decimals and the label printer prints up to 5 across – any size and even allows single envelopes or index cards.

JINSAM 1.0 allows fast and easy file handling, manipulation and report generation for any CBM computer with CBM 2040 disk drive. It features a menu for ease, has encrypted passwords, 3 deep sorts, .5 to 3 second recall.

JINSAM 4.0 for CBM 4000 series adds JINSORT, user accessible machine sort of 1000 records in 15 seconds; compaction/expansion of information, automatic list maintainance, unlimited number of fields, unlimited record length and much more.

JINSAM 8.0 for CBM 8000 series has all 4.0 features plus unlimited sort, horizontal format, and search by key or record number.

JINSAM 8.2, NEW FOR '82 expands 8.0 capabilities by adding information search by word, key or record number and machine language print, format and manipulation routines.

- ★ CUSTOM DATA FILES
- ★ CUSTOM REPORTS/LABELS
- ★ KEYED RANDOM ACCESS
- ★ FAST/EASY/MENU DRIVEN
- ★ MULTIPLE SEARCH KEYS
- ★ PRIVACY ACCESS CODES
- ★ WILD CARD SEARCH

JINSAM EXECUTIVE version (soon to be released) is our most powerful professional system for the CBM 8000 and 9000 series. Executive will have 8.2 extended features plus allow multiple users with in-use lockout protection, executive command files, automatic math relations, join, merge or link files, greatly increased record capacity and machine information search by word, as well as by key or record number and many, many more features.

There are currently 7 more interfacing modules – and more under development, including independent interfaces between JINSAM and business packages for your G/L, A/R, A/P needs. We announce the availability of modules and enhancements in JINSAM's quarterly newsletter.

WORDPROPACK - Intelligent interface for WordPro 3, 3+, 4, 4+, creates lists of information from JINSAM files. It allows up to 10 conditions based on each item of information. Produce individualized letters, report cards, special reports, checks, invoices, etc.

MULTI-LABEL - Prints multiple labels per record with up to 2 lines for messages and consecutive numbering. Produce inventory, caution labels, bulk mail labels, etc.

MATHPACK - global calculator/editor +, −, *, /, by another field or constant; null (remove contents) of a field or replace contents of a field with any word, number or phrase. Sum multiple fields in each record or running sum of single field in all records. Extract information or effect permanent change. Replace in the same field or place in a waiting field.

DESCRIPTIVE STATPACK - Determine MEAN, MEDIAN, MODE, STANDARD DEVIATION, VARIANCE, RANGE. Generate HISTOGRAMS from 1 to 25 steps, and produce Z-SCORE reports.

ADVANCED STATPACK - (You must also acquire DESCRIPTIVE STATPACK) Generate one, two or three way CROSSTABS (number of occurance) CHI SQUARE, LINEAR REGRESSION, with graphic representation and prediction, LINEAR CORRELATION and SIMPLE ANALYSIS OF VARIANCE.

CALCPACK - 2 way interface to VisiCalc or any user program. It lets you use VisiCalc for complex manipulation, editing, placing results in JINSAM for sorting, storing or moving data to WordPro as well as giving the ability for exchange with your own applications.

INTERAC - Interface which can read VisiCalc files, WordPro files and almost any sequential files to build JINSAM databases automatically. For example: You could "download" information on holerith cards to sequential files and INTERAC would place them into JINSAM files.

All accessories are accessed thru the JINSAM menu and require security password to gain entrance.

JINSAM gives you FREEDOM OF CHOICE. Start with JINSAM 1.0 and upgrade hardware and data at any time. Choose from accessories at any time. The JINSAM Newsletter brings the latest updates, user input and uses and keeps an eye on the future.

JINSAM stands alone by placing "a lot of potential computing power in one integrated program package" (Fred Klein, Ferson div. of Bausch & Lomb). "The JINSAM package is justification for buying a system no matter what the hardware, be it Vector or Commodore or whatever the system" (Larry Colvin, Micro Computer Systems). It is a "grandfather" in this young field. JINSAM EXECUTIVE will be the third generation in development. All JINSAM systems are sophisticated and flexible yet easy to use. JINSAM is saving its users valuable time and money in government, educational and research institutions, business and industry worldwide.

JINSAM is a Commodore approved product. See your local dealer for a demonstration.

JINSAM Data Manager

. . . . Additional Information and nearest dealer
. . . . Newsletter Subscription ($5 US/$8 Worldwide)
. . . . JINSAM Demo Disk ($15, plus shipping & tax)
. . . . User's Guide ($40.00, plus shipping & tax)

Please send to:

Name ____________________
Position ____________________
Company ____________________
Address ____________________
City, State, Zip ____________________
Phone (   ) ____________________
Computer, Disk ____________________
Application ____________________

JINI MICRO-SYSTEMS, Inc.
Box 274C • Kingsbridge Station
Riverdale, New York 10463
(212) 796-6200

Dealer Inquiry Welcome

```
40 SP$=" ":DL$=CHR$(20)
45 M=79:REM DEFAULT TEXT STRING LE
   NGTH
50 EE=0:REM DISKERRROR CHANNEL CLO
   SED
60 PR=0
90 REM POKE144,49:REM DISABLE STOP
    KEY
100 PRINT"{CLEAR}      {REV}ABACUS S
    OFTWARE LINE EDITOR"
110 PRINT"{02 DOWN}    FUNCTIONS:"
130 PRINT
140 PRINTTAB(8);"A)PPEND-TO END OF ~
    TEXT"
145 PRINTTAB(10);"@ = 5 SPACE TAB"
146 PRINTTAB(10);"@+{REV}RETURN{OFF
    OFF} = SKIP LINE
150 PRINTTAB(8);"C)HANGE-STRING
160 PRINTTAB(8);"D)ELETE LINE(S)
170 PRINTTAB(8);"F)ILER COMMANDS
180 PRINTTAB(8);"I)NSERT BEFORE LIN
    E
190 PRINTTAB(8);"L)IST LINE(S)
200 PRINTTAB(8);"M)ENU DISPLAY
210 PRINTTAB(8);"P)RINT LINES(S)
220 PRINTTAB(8);"Q)UIT LINE EDITOR
230 PRINTTAB(8);"R)EPLACE LINE
235 PRINTTAB(8);"S)ET MARGINS
240 PRINT:PRINT"    ENTER SELECTION
    -> ";
250 GOTO510
500 PRINT:PRINT"{REV}ENTER{OFF} A,C
    ,D,F,I,L,P,Q,R,S,M)ENU->";

510 GETA$:IFA$=""THEN510
520 J=0:FORI=1TO11
530 IFA$=MID$("ACDFILRMQPS",I,1)THE
    NJ=I:I=11
540 NEXTI
550 PRINTA$
560 IFJ=0 THEN500
570 ONJGOTO1000,2000,3000,4000,5000
    ,6000,7000,100,8000,9000,2
    1000
1000 PRINT
1005 PRINT"{REV}APPEND{OFF} TO END O
     F TEXT"
1010 PRINT:PRINT LL">";
1020 GOSUB10000
1030 IFLEN(L$)=0THEN500
1040 T$(LL)=L$
1050 LL=LL+1
1060 GOTO1010
2000 REM CHANGE STRING
2010 PRINT:PRINT"{REV}CHANGE{OFF}";:
     GOSUB16000
2020 IFHI=0THEN500
2025 PRINT"{REV}CHANGE{OFF} STRING->
     ";:GOSUB10000
2030 L=LEN(L$)
2040 IFL=0THEN500
2050 IFL<4THEN2000
2060 DM$=LEFT$(L$,1):REM DELIMITER
2070 IFRIGHT$(L$,1)<>DM$THEN2000
2080 J=0:FORI=2TOL-1
2090 IFMID$(L$,I,1)=DM$THENJ=I
2100 NEXTI
2110 IFJ=0THEN2000
2120 IFJ=2THEN2000
2130 FR$=MID$(L$,2,J-2)
2140 IFJ+1=LTHENTS$="":GOTO2160
2150 TS$=MID$(L$,J+1,L-J-1)
2160 F=LEN(FR$)
2170 FORI=LOTOHI
2180 T=LEN(T$(I)):S=1:NL$=""
2190 FORJ=1TOT-F+1
2200 IFMID$(T$(I),J,F)<>FR$THEN2230
2210 NL$=NL$+MID$(T$(I),S,J-S)+TS$
2220 S=J+F:J=S-1
2230 NEXTJ
2240 IFS<>1THENNL$=NL$+RIGHT$(T$(I),
     T-S+1):T$(I)=NL$
2250 NEXTI
2260 GOTO500
3000 REM DELETE LINES
3005 PRINT:PRINT"{REV}DELETE{OFF} ";
     :GOSUB16000:REM GET RANGE
3010 IFNOTDFTHEN3015:REM NOT DEFAULT
      ON ENTIRE FILE
3011 PRINT"{REV}DELETE{OFF} ENTIRE F
     ILE?  ";
3012 GETA$:IFA$=""THEN3012
3013 PRINTA$:IFA$="N"THEN500
3014 IFA$<>"Y"THEN3011
3015 IFHI>LL-1THEN500
3020 IFHI=LL-1THENLL=LO:GOTO500
3030 J=HI-LO+1
3040 FORI=LOTOLL-J-1
3050 T$(I)=T$(I+J)
3060 NEXTI
3070 LL=LL-(HI-LO)-1
3080 GOTO500
4000 REM FILLER
4010 PRINT"{DOWN}{REV}FILER{OFF} ENT
     ER L)OAD OR S)AVE-> ;
4020 GETA$:IFA$=""THEN4020
4030 IFA$<>"L"ANDA$<>"S"THENPRINT:GO
     TO4000
4040 PRINTA$:M$=A$
4050 PRINT"{REV}ENTER{OFF} FILENAME-
     > ";
4070 GOSUB10000
4075 IFLEN(L$)=0THEN500
4076 IFLEN(L$)>12THEN4050
4080 FI$=L$
4090 PRINT"{REV}ENTER{OFF} D)ISK OR ~
     T)APE-> ";
4100 GETA$:IFA$=""THEN4100
4110 PRINTA$
4120 IFA$<>"D"ANDA$<>"T"THEN4090
4130 IFA$="D"THEN4160
4140 IFM$="L"THEN4400
4150 GOTO4200
4160 DR$="":IFLEFT$(FI$,2)<>"0:"ANDL
     EFFT$(FI$,2)<>"1:"THENDR$=
     "0:"
4170 GOTO4600
4200 REM TAPE SAVE
4210 IFLL=1THENPRINT"NO FILE TO SAVE
     ":GOTO500
4220 OPEN2,1,2,FI$+".SOURCE"
```

```
4230 FORI=1TOLL-1
4240 FORJ=1TOLEN(T$(I))
4250 PRINT#2,MID$(T$(I),J   ,1);
4260 NEXTJ
4270 PRINT#2,CHR$(255);
4280 NEXTI
4290 CLOSE2
4300 PRINTSPC(6);FI$;"  SAVED"
4310 GOTO5ØØ
4400 REMTAPE LOAD
4410 OPEN2,1,Ø,FI$+".SOURCE"
4430 LL=Ø:REMLINE COUNT
4440 LL=LL+1:T$(LL)=""
4450 GET#2,A$
4460 IFST=64THEN45ØØ:REM END OF FILE
4465 IFST<>ØTHEN PRINT"*** LOAD ERRO
    R ***":GOTO5ØØ
4470 IFA$=CHR$(255)THEN444Ø:REM END ~
    OF LINE
4480 T$(LL)=T$(LL)+A$
4490 GOTO445Ø
4500 CLOSE2
4510 PRINTSPC(6);FI$;" LOADED"
4520 LL=LL+1
4530 GOTO5ØØ
4600 REM DISK SAVE
4610 IFM$="L"THEN48ØØ
4620 IFLL=1THENPRINT"NO FILE TO SAVE
    ":GOTO5ØØ
4630 FL$="@Ø"+DR$+FI$+".SOURCE,S,W"
4640 OPEN2,8,2,FL$
4650 GOSUB2ØØØØ:REM ERROR CHECK
4660 FORI=1TOLL-1
4670 FORJ=1TOLEN(T$(I))
4680 PRINT#2,MID$(T$(I),J,1);
4690 NEXTJ
4700 PRINT#2,CHR$(255)
4710 NEXTI
4720 CLOSE2
4730 PRINTSPC(6);FI$;" SAVED"
4740 GOTO5ØØ
4800 REM DISK LOAD
4810 FL$=DR$+FI$+".SOURCE,S,R"
4820 OPEN2,8,2,FL$
4830 GOSUB2ØØØØ:REM ERROR CHECK
4835 IFE1<>ØTHEN5ØØ
4840 LL=Ø:REM LINE COUNT
4850 LL=LL+1:T$(LL)=""
4860 GET#2,A$
4870 IFST=64THEN45ØØ:REM END OF FILE
4880 IFST<<>ØTHENGOSUB2ØØØØ:GOTO5ØØ
4890 IFA$=CHR$(255)THEN485Ø:REM END ~
    OF LINE
4900 T$(LL)=T$(LL)+A$
4910 GOTO486Ø
4920 CLOSE2
4930 PRINTSPC(6);FI$;" LOADED"
4940 LL=LL+1
4950 GOTO5ØØ
5000 REM INSERT LINE
5010 PRINT:PRINT"{REV}INSERT{OFF} BE
    FORE ";:GOSUB17ØØØ:REM GET
      LINE #
5015 IFLO>LLORLO<1THEN5ØØØ
5020 PRINT:PRINTLO;">";
5030 GOSUB1ØØØØ:REM READ LINE
5040 IFLEN(L$)=ØTHEN5ØØ
5050 LL=LL+1
5060 FORI=LLTOLOSTEP-1
5070 T$(I)=T$(I-1)
5080 NEXTI
5090 T$(LO)=L$
5100 LO=LO+1
5110 GOTO5Ø2Ø
6000 REM LIST LINES
6010 PRINT:PRINT"{REV}LIST{OFF} ";:G
    OSUB16ØØØ:REM GET RANGE
6020 IFHI=ØTHEN5ØØ
6030 SS$="N":PRINT:FORI=LOTOHI:REM P
    ERFORM LIST
6040 PRINTI;">";T$(I)
6050 GETA$:IFA$=CHR$(18)THENFORJ=1TO
    1024:NEXTJ
6060 IFA$<>CHR$(3)THEN611Ø
6070 SS$="Y"
6080 GETA$:IFA$=CHR$(13)THENSS$="N":
    GOTO611Ø
6090 IFA$<>CHR$(32)THEN6Ø7Ø
6100 GOTO612Ø
6110 IFSS$="Y"THEN6Ø7Ø
6120 NEXTI
6130 GOTO5ØØ
7000 REM REPLACELINE
7010 PRINT:PRINT"{REV}REPLACE{OFF} "
    ;:GOSUB17ØØØ:REM GET LINE#
7020 IFLO>=LLORLO<1THEN7ØØØ
7030 PRINT:PRINTLO,">";
7040 GOSUB1ØØØØ:REM RED LINE
7050 IFLEN(L$)=ØTHEN5ØØ
7060 T$(LO)=L$
7070 GOTO5ØØ
8000 REM QUIT
8010 PRINT:PRINT"      {REV}LEAVE EDI
    TOR-ARE YOU SURE?{OFF} ";
8020 GETA$:IFA$=""THEN8Ø2Ø
8030 PRINTA$
8040 IFA$<>"Y"ANDA$<>"N"THEN8ØØØ
8050 IFA$="N"THEN5ØØ
8060 PRINT:PRINT"       {REV}** END L
    INE EDITOR **{OFF}"
8070 POKE144,46:REM ENABLE STOP KEY
8075 POKE59468,12:REM ENABLE UPPERCA
    SE AND GRAPHICS
8080 END
9000 REM PRINT LINE
9010 IFPR=ØTHENPR=4:OPENPR,PR
9020 PRINT:PRINT"{REV}PRINT{OFF} ";:
    GOSUB16ØØØ:REM GET RANGE
9025 INPUT"NUMBER OF SPACES BETWEEN ~
    LINES(1-2)";S1
9030 IF HI=ØTHEN5ØØ
9040 FORI=LOTOHI:REM PERFORM PRINT
9050 PRINT#PR,SPC(SP)"{DOWN}"+T$(I)
9055 IFS1=2THENPRINT#PR:REM DOUBLE S
    PACE
9060 NEXTI
9065 GOTO91ØØ:REM  ***  TO GET RID O
    F JUNK ***
9070 PRINT#PR
9080 PRINT#PR,"***";LL-1;"LINES IN B
```

DYNACOMP

Quality software for:*

ATARI	TRS-80 (Level II)**
PET	NORTH STAR
APPLE II Plus	CP/M Disks/Diskettes

(see Availability box)

DYNACOMP OFFERS THE FOLLOWING

- Widest variety
- Guaranteed quality
- Fastest delivery
- Friendly customer service
- Free catalog
- 24 hour order phone

CARD GAMES

BACCARAT (Atari only) — **Price: $18.95 Cassette/$22.95 Diskette**
This is the European card game which is the favorite of the Monte Carlo jet set. Imagine yourself at the gaming table with 007 to your left and Goldfinger to your right. Learn and play BACCARAT at your leisure on the Atari. Contains full high resolution color graphics and matching sound. Runs in 16K. Requires one joystick.

GIN RUMMY (Apple only) — **Price: $18.95 Cassette/$22.95 Diskette**
This is the best micro computer implementation of GIN RUMMY existing. The computer plays exceptionally well, and the HIRES graphics are superb. What else can be said?

POKER PARTY (Available for all computers) — **Price: $17.95 Cassette/$21.95 Diskette**
POKER PARTY is a draw poker simulation based on the book, POKER, by Oswald Jacoby. This is the most comprehensive version available for microcomputers. The party consists of yourself and six other (computer) players. Each of these players (you will get to know them) has a different personality in the form of a varying propensity to bluff or fold under pressure. Practice with POKER PARTY before going to that expensive game tonight! Apple cassette and diskette versions require a 32 K (or larger) Apple II.

CRIBBAGE 2.0 (TRS-80 only) — **Price: $14.95 Cassette/$18.95 Diskette**
This is simply the best cribbage game available. It is an excellent program for the cribbage player in search of a worthy opponent as well as for the novice wishing to improve his game. The graphics are superb and assembly language routines provide rapid execution. See the software review in 80 Software Critique.

THOUGHT PROVOKERS

MANAGEMENT SIMULATOR (Atari, North Star and CP/M only) — **Price: $19.95 Cassette/$23.95 Diskette**
This program is both an excellent teaching tool as well as a stimulating intellectual game. Based upon similar games played at graduate business schools, each player or team controls a company which manufacturers three products. Each player attempts to outperform his competitors by setting selling prices, production volumes, marketing and design expenditures etc. The most successful firm is the one with the highest stock price when the simulation ends.

FLIGHT SIMULATOR (Available for all computers) — **Price: $17.95 Cassette/$21.95 Diskette**
A realistic and extensive mathematical simulation of take-off, flight and landing. The program utilizes aerodynamic equations and the characteristics of a real airfoil. You can practice instrument approaches and navigation using radials and compass headings. The more advanced flyer can also perform loops, half-rolls and similar aerobatic maneuvers. Although this program does not employ graphics, it is exciting and very addictive. See the software review in COMPUTRONICS. Runs in 16K Atari.

VALDEZ (Available for all computers) — **Price: $15.95 Cassette/$19.95 Diskette**
VALDEZ is a computer simulation of supertanker navigation in the Prince William Sound/Valdez Narrows region of Alaska. Included in this simulation is a realistic and extensive 256 × 256 element map, portions of which may be viewed using the ship's alphanumeric radar display. The motion of the ship itself is accurately modelled mathematically. The simulation also contains a model for the tidal patterns in the region, as well as other traffic (outgoing tankers and drifting icebergs). Chart your course from the Gulf of Alaska to Valdez Harbor! See the software reviews in 80 Software Critique and Personal Computing.

BACKGAMMON 2.0 (Atari, North Star and CP/M only) — **Price: $14.95 Cassette/$18.95 Diskette**
This program tests your backgammon skills and will also improve your game. A human can compete against a computer or against another human. The computer can even play against itself. Either the human or the computer can double or generate dice rolls. Board positions can be created or saved for replay. BACKGAMMON 2.0 plays in accordance with the official rules of backgammon and is sure to provide many fascinating sessions of backgammon play.

CHECKERS 3.0 (PET only) — **Price: $16.95 Cassette/$20.95 Diskette**
This is one of the most challenging checkers programs available. It has 10 levels of play and allows the user to change skill levels at any time. Although providing a very tough game at level 4-8, CHECKERS 3.0 is practically unbeatable at levels 9 and 10.

CHESS MASTER (North Star and TRS-80 only) — **Price: $19.95 Cassette/$23.95 Diskette**
This complete and very powerful program provides five levels of play. It includes castling, en passant captures and the promotion of pawns. Additionally, the board may be preset before the start of play, permitting the examination of "book" plays. To maximize execution speed, the program is written in assembly language (by SOFTWARE SPECIALISTS of California). Full graphics are employed in the TRS-80 version, and two widths of alphanumeric display are provided to accommodate North Star users. See review in onComputing.

LEM LANDER (32K Apple Disk only) — **Price: $16.95 Diskette**
Pilot your LEM LANDER to a safe landing on any of nine different surfaces ranging from smooth to treacherous. The game paddles are used to control craft attitude and thrust. This is a real-time high res challenge!

FOREST FIRE! (Atari only) — **Price: $16.95 Cassette/$20.95 Diskette**
Using excellent graphics and sound effects, this simulation puts you in the middle of a forest fire. Your job is to direct operations to put out the fire while compensating for changes in wind, weather and terrain. Not protecting valuable structures can result in startling penalties. Life-like variables are provided to make FOREST FIRE! very suspenseful and challenging. No two games have the same setting and there are 3 levels of difficulty.

SPACE EVACUATION! (Apple, Atari and TRS-80 only) — **Price: $15.95 Cassette/$19.95 Diskette**
Can you colonize the galaxy and evacuate the Earth before the sun explodes? Your computer becomes the ship's computer as you explore the universe to relocate millions of people. This simulation is particularly interesting as it combines many of the exciting elements of classic space games with the mystery challenge of ADVENTURE.

MONARCH (Atari only) — **Price: $11.95 Cassette/$15.95 Diskette**
MONARCH is a fascinating economic simulation requiring you to survive an 8-year term as your nation's leader. You determine the amount of acreage devoted to industrial and agricultural use, how much food to distribute to the populace and how much should be spent on pollution control. You will find that all decisions involve a compromise and that it is not easy to make everyone happy. Runs in 16K Atari.

CHOMPELO (Atari only) — **Price: $11.95 Cassette/$15.95 Diskette**
CHOMPELO is really two challenging games in one. One is similar to NIM; you must bite off part of a cookie, but avoid taking the poisoned portion. The other game is the popular board game REVERSI. It fully uses the Atari's graphics capability, and is hard to beat. This package will run on a 16K system.

SPACE LANES (Available for all computers) — **Price: $10.95 Cassette/$14.95 Diskette**
SPACE LANES is a simple but exciting space transportation game which involves up to four players (including the computer). The object is to form and expand space transportation companies in a competitive environment. The goal is to amass more net worth than your opponent. The economics include stock purchases and company mergers. Watch your wealth grow!

AVAILABILITY

DYNACOMP software is supplied with complete documentation containing clear explanations and examples. Unless otherwise specified, all programs will run within 16K program memory space (ATARI requires 24K). Except where noted, programs are available on ATARI, PET, TRS-80 (Level II) and Apple (Applesoft) cassette and diskette as well as North Star single density (double density compatible) diskette. Additionally, most programs can be obtained on standard (IBM 3740 single density/double density compatible format) 8" CP/M floppy disks for systems running under MBASIC (for example, Altos, Xerox 820 and many others). 5¼" CP/M diskettes are available for the North Star and Osborne computer systems.

**ATARI, PET/CBM, NORTH STAR, CP/M, IBM, OSBORNE and XEROX are registered tradenames and/or trade marks.*

***Except where noted, all TRS-80 Model I software is available on cassette (only) for the TRS-80 Model III. Exceptions: VALDEZ, CRIBBAGE, GRAFIX, CHESSMASTER. TRS-80 diskettes are not supplied with either DOS or BASIC.*

AND MORE...

STARTREK 3.2 (Available for all computers) — **Price: $11.95 Cassette/$15.95 Diskette**
This is the classic Startrek simulation, but with several new features. For example, the Klingons now shoot at the Enterprise without warning while also attacking starbases in other quadrants. The Klingons also attack with both light and heavy cruisers and move when shot at! The situation is hectic when the Enterprise is besieged by three heavy cruisers and a starbase S.O.S. is received! The Klingons get even! See the software reviews in A.N.A.L.O.G., 80 Software Critique and Game Merchandising.

BLACK HOLE (Apple only) — **Price: $14.95 Cassette/$18.95 Diskette**
This is an exciting graphical simulation of the problems involved in closely observing a black hole with a space probe. The object is to enter and maintain, for a prescribed time, an orbit close to a small black hole. This is to be achieved without coming so near the anomaly that the tidal stress destroys the probe. Control of the craft is realistically simulated using side jets for rotation and main thrusters for acceleration. This program employs Hi-Res graphics and is educational as well as challenging.

SPACE TILT (Apple and Atari only) — **Price: $10.95 Cassette/$14.95 Diskette**
Use the game paddles to tilt the plane of the TV screen to "roll" a ball into a hole in the screen. Sound simple? Not when the hole gets smaller and smaller! A built-in timer allows you to measure your skill against others in this habit-forming action game.

ESCAPE FROM VOLANTIUM (Atari only) — **Price: $15.95 Cassette/$19.95 Diskette**
Bring the action and excitement of an arcade into your home with ESCAPE FROM VOLANTIUM! To escape you must maneuver your space ship around obstacles and laser blast the dragon (without being eaten). If he is killed with a direct shot (not just a leg lopped off), a door opens to the outside. However, the door does not stay open indefinitely. If you fail to escape in time, the door closes and a new dragon appears. Sometimes you can smash through the door by repeatedly chipping away at it. Other times it is impervious. At the higher levels of play more obstacles and dragons appear, adding to the excitement. Uses high resolution graphics and sound. Runs in 16K.

ALPHA FIGHTER (Atari only) — **Price: $14.95 Cassette/$18.95 Diskette**
Two excellent graphics and action programs in one! ALPHA FIGHTER requires you to destroy the alien starships passing through your sector of the galaxy. ALPHA BASE is in the path of an alien UFO invasion; let five UFO's get by and the game ends. Both games require the joystick and get progressively more difficult the higher you score! ALPHA FIGHTER will run on 16K systems.

THE RINGS OF THE EMPIRE (Atari only) — **Price: $16.95 Cassette/$20.95 Diskette**
The empire has developed a new battle station protected by rotating rings of energy. Each time you blast through the rings and destroy the station, the empire develops a new station with more protective rings. This exciting game runs on 16K systems, employs extensive graphics and sound and can be played by one or two players.

INTRUDER ALERT (Atari only) — **Price: $16.95 Cassette/$20.95 Diskette**
This is a fast paced graphics game which places you in the middle of the "Dreadstar" having just stolen its plans. The droids have been alerted and are directed to destroy you at all costs. You must find and enter your ship to escape with the plans. Five levels of difficulty are provided. INTRUDER ALERT requires a joystick and will run on 16K systems.

MIDWAY (Atari only) — **Price: $14.95 Cassette/$18.95 Diskette**
MIDWAY is an exciting extension of the game of Battleship. It mixes the challenges of strategy and chance. Your opponent can be another human or the computer. Color graphics and sound are both included. Runs in 16K.

TRIPLE BLOCKADE (Atari only) — **Price: $14.95 Cassette/$18.95 Diskette**
TRIPLE BLOCKADE is a two-to-three player graphics and sound action game. It is based on the classic video arcade game which millions have enjoyed. Using the Atari joysticks, the object is to direct your blockading line around the screen without running into your opponent(s). Although the concept is simple, the combined graphics and sound effect lead to "high anxiety".

GAMES PACK I (Available for all computers) — **Price: $10.95 Cassette/$14.95 Diskette**
GAMES PACK I contains the classic computer games of BLACKJACK, LUNAR LANDER, CRAPS, HORSERACE, SWITCH and more. These games have been combined into one large program for ease in loading. They are individually accessed by a convenient menu. This collection is worth the price just for the DYNACOMP version of BLACKJACK.

GAMES PACK II (Available for all computers) — **Price: $10.95 Cassette/$14.95 Diskette**
GAMES PACK II includes the games CRAZY EIGHTS, JOTTO, ACEY-DUCEY, LIFE, WUMPUS and others. As with GAMES PACK I, all the games are loaded as one program and are called from a menu. You will particularly enjoy DYNACOMP's version of CRAZY EIGHTS.

Why pay $7.95 or more per program when you can buy a DYNACOMP collection for just $10.95?

MOON PROBE (Atari and North Star only) — **Price: $11.95 Cassette/$15.95 Diskette**
This is an extremely challenging "lunar lander" program. The user must drop from orbit to land at a predetermined target on the moon's surface. You control the thrust and orientation of your craft plus direct the rate of descent and approach angle. Runs in 16K Atari.

SPACE TRAP (Atari only, 16K) — **Price: $14.95 Cassette/$18.95 Diskette**
This galactic "shoot'em up" arcade game places you near a black hole. You control your spacecraft using the joystick and attempt to blast as many of the alien ships as possible before the black hole closes about you.

CHIRP INVADERS (PET/CBM only) — **Price: $14.95 Cassette/$18.95 Diskette**
CHIRP INVADERS is an addictive game using action graphics. A Federation space station must be reached before the Chirps conquer the Earth. Stationary obstacles, moving meteors, and the attacking Chirps must all be avoided for a successful journey. Good luck.

ADVENTURE

CRANSTON MANOR ADVENTURE (North Star and CP/M only) — **Price: $29.95 Diskette**
At last! A comprehensive Adventure game for North Star and CP/M systems. CRANSTON MANOR ADVENTURE takes you into mysterious CRANSTON MANOR where you attempt to gather fabulous treasures. Lurking in the manor are wild animals and robots who will not give up the treasures without a fight. The number of rooms is greater and the associated descriptions are much more elaborate than the current popular series of Adventure programs, making this game the top in its class. Play can be stopped at any time and the status stored on diskette. Not available in 5¼" CP/M format.

GUMBALL RALLY ADVENTURE (North Star only, 48K) — **Price: $21.95 Diskette**
Take part in this outlaw race from the east coast to the west coast. The goal is to find your way to the finish line while maintaining the highest possible speed. You may choose one of five cars available at the garage. The choice will affect your speed and range. Remember to take spare parts and don't get caught speeding!

UNCLE HARRY'S WILL (North Star only, 40K) — **Price: $24.95 Diskette**
Uncle Harry has died and has left you everything. However, he has neglected to mention where everything is! Instead, his will consists of a poem which contains clues. You will have to travel all over the United States both by car and on foot to solve the puzzle, and there are over 300 locations to probe. Be careful and watch out for red herrings!

SPEECH SYNTHESIS

DYNACOMP is now distributing the new and revolutionary TYPE-'N-TALK™ (TNT) speech synthesizer from Votrax. Simply connect TNT to your computer's serial interface, enter text from the keyboard and hear the words spoken. TNT is the easiest-to-program speech synthesizer on the market. It uses the least amount of memory and provides the most flexible vocabulary available anywhere!

List price $375. DYNACOMP'S price $329.95. Please add $5.00 for shipping and handling

TALK TO ME (T'N'T Atari only, 24K) — **Price: $14.95 Cassette/$18.95 Diskette**
This program presents a superb tutorial on speech synthesis using the Atari 800 and TYPE 'N TALK™. TALK TO ME will illustrate normal word generation as well as phoneme generation. The documentation includes many helpful programming tips.

MISCELLANEOUS

CRYSTALS (Atari only) — **Price: $ 9.95 Cassette/$13.95 Diskette**
A unique algorithm randomly produces fascinating graphics displays accompanied with tones which vary as the patterns are built. No two patterns are the same, and the combined effect of the sound and graphics are mesmerizing. CRYSTALS has been used in local stores to demonstrate the sound and color features of the Atari. Runs in 16K Atari.

NORTH STAR SOFTWARE EXCHANGE (NSSE) LIBRARY
DYNACOMP now distributes the 23 volume NSSE library. These diskettes each contain many programs and offer an outstanding value for the purchase price. They should be part of every North Star user's collection. Call or write DYNACOMP for details regarding the contents of the NSSE collection.

Price: $9.95 each/$7.95 each (4 or more)
The complete collection may be purchased for $149.95

BUSINESS and UTILITIES

MAILMASTER (Atari diskette only) — **Price: $39.95 Diskette**
MAILMASTER is a very versatile software package for managing and manipulating mail lists and mini data bases. Each disk can hold over 700 customer entries containing name, address, three 3-letter key words and a phone number. The display is marked so that entries may be made and edited with ease. The status (e.g., disk space left, options, etc.) is shown at all times. Labels may be printed 1,2 or 3 up, and all sorting (zip code and alphabetic) is performed by a fast machine language program.

SORTIT (North Star only) — **Price: $29.95 Diskette**
SORTIT is a general purpose sorting program written in 8080 assembly language. This program will sort sequential data files generated by NORTH STAR BASIC. Primary and optional secondary keys may be numeric or one to nine character strings. SORTIT is easily used with files generated by DYNACOMP's MAIL LIST program and is very versatile in its capabilities for all other BASIC data file sorting.

PERSONAL FINANCE SYSTEM (Atari and North Star only) — **Price: $39.95 Diskette**
PFS is a single diskette, menu-oriented system composed of ten different programs. Besides recording your expenses and tax deductible items, PFS will sort and summarize expenses by payee, and display information on expenditures by any of 26 user defined codes by month or by payee. PFS will even produce monthly bar graphs of your expenses by category! This powerful package requires only one disk drive, minimal memory (24K Atari, 32K North Star) and will store up to 600 records per disk (and over 1000 records per disk by making a few simple changes to the programs). You can record checks plus cash expenses so that you can finally see where your money goes and eliminate guesswork and tedious hand calculations. Contains high speed machine language sort.

FAMILY BUDGET (Apple and Atari only) — **Price: $34.95 Diskette**
FAMILY BUDGET is a very convenient financial record-keeping program. You will be able to keep track of cash and credit expenditures as well as income on a daily basis. You can record tax deductible items and charitable donations. FAMILY BUDGET also provides a continuous record of all credit transactions. You can make daily cash and charge entries to any of 21 different expense accounts as well as to 5 payroll and tax accounts. Data are easily retrieved giving the user complete control over an otherwise complicated (and unorganized!) subject.

INTELINK (Atari only) — **Price: $49.95 Diskette**
This software package contains a menu-driven collection of programs for facilitating efficient two-way communications through a full duplex modem (required for use). In one mode of operation you may connect to a data service (e.g., The SOURCE or MicroNet) and quickly load data such as stock quotations onto your diskette for later viewing. This greatly reduces "connect time" and thus the service charge. You may also record the complete contents of a communications session. Additionally, programs written in BASIC, FORTRAN, etc. may be built off-line using the support text editor and later "uploaded" to another computer, making the Atari a *very* smart terminal. Even Atari BASIC programs may be uploaded. Further, a command file may be built off-line and used later as controlling input for a time-share system. That is, you can set up your sequence of time-share commands and programs, and the Atari will transmit them as needed; batch processing. All this adds up to saving both connect time and your time.

TEXT EDITOR II (CP/M) — **Price: $29.95 Diskette/$33.45 Disk**
This is the second release version of DYNACOMP's popular TEXT EDITOR I and contains many new features. With TEXT EDITOR II you may build text files in chunks and assemble them for later display. Blocks of text may be appended, inserted or deleted. Files may be saved on disk/diskette in right justified/centered format to be later printed by either TEXT EDITOR II or the CP/M ED facility. Futher, ASCII CP/M files (including BASIC and assembly language programs) may be read by the editor and processed. In fact, text files can be built using ED and later formatted using TEXT EDITOR II. All in all, TEXT EDITOR II is an inexpensive, easy to use, but very flexible editing system.

DFILE (Atari and North Star diskettes only) — **Price: $19.95**
This handy program allows North Star and Atari disk users to maintain a specialized data base of all files and programs in the stack of disks which invariably accumulates. DFILE is easy to set up and use. It will organize your disks to provide efficient locating of the desired file or program.

FINDIT (North Star only) — **Price: $19.95**
This is a three-in-one program which maintains information accessible by keywords of three types: Personal (eg: last name), Commercial (eg: plumbers) and Reference (eg: magazine articles, record albums, etc). In addition to keyword searches, there are birthday, anniversary and appointment searches for the personal records and appointment searches for the commercial records. Reference records are accessed by a single keyword or by cross-referencing two or three keywords.

SHOPPING LIST (Atari only) — **Price: $12.95 Cassette/$16.95 Diskette**
SHOPPING LIST stores information on items you purchase at the supermarket. Before going shopping, it will remind you of all the things you might need, and then display (or optionally print) your shopping list and the total cost. Adding, deleting, changing and storing data is very easy. Runs with 16K.

TAX OPTIMIZER (North Star only) — **Price: $59.95 Diskette**
The TAX OPTIMIZER is an easy-to-use, menu oriented software package which provides a convenient means for analyzing various income tax strategies. The program is designed to provide a quick and easy data entry. Income tax is computed by all tax methods (regular, income averaging, maximum and alternate minimum tax). The user may immediately observe the tax effect of critical financial decisions. TAX OPTIMIZER has been thoroughly field tested in CPA offices and comes complete with the current tax tables in its data files. TAX OPTIMIZER is tax deductible!

UTIL (Apple only, 48K) — **Price: $19.95 Diskette**
UTIL is a disk-oriented utility system which permits examining and changing of the contents of DOS 3.2 and 3.3 diskettes at the bit (nibble or byte) level. With UTIL you can easily examine the contents of a diskette sector by sector, restructure the sector pointers, reallocate sectors (e.g. bad sectors may be "hidden"), and perform many other sophisticated operations. For the experienced programmer.

TURNKEY AND MENU (Atari only) — **Price: $17.95 Diskette**
TURNKEY is a utility program which allows you to create autoboot/autorun diskettes easily. Simply load and run TURNKEY, load the program diskette to be modified, and answer the questions! The TURNKEY diskette also comes with DOS 2.0 and includes another program, MENU. MENU lists the contents of your diskette alphabetically, and permits the running of any BASIC program on the diskette by typing a single key. TURNKEY and MENU provide you with the ability to run any program on your diskette by simply turning on the computer and pressing a single key.

STOCKAID (Atari only) — **Price: $29.95 Diskette**
STOCKAID provides a powerful set of tools for stock market analysis. With STOCKAID you can display point and figure charts, as well as bar charts with oscillators. You can also examine long term moving averages and on-balance volume features. STOCKAID allows you to input daily data with a single diskette storage capability of 239 days × 16 stocks. Included are stock dividend and split adjustment capabilities. A very professional package!

EDUCATION

HODGE PODGE (Apple only, 48K Applesoft or Integer BASIC) — **Price: $19.95 Cassette/$23.95 Diskette**
Let HODGE PODGE be your child's teacher. Pressing any key on your Apple will result in a different and intriguing "happening" related to the letter or number of the chosen key. The program's graphics, color and sound are a delight for children from ages 1½ to 7. HODGE PODGE is a non-intimidating teaching device which brings a new dimension to the use of computers in education. See review in Infoworld.

TEACHER'S AIDE (Atari only) — **Price: $13.95 Cassette/$17.95 Diskette**
TEACHER'S AIDE consists of three basic modules contained in one program. The first module provides addition and subtraction exercises of varying levels of difficulty. The second module consists of multiplication problems in which the student may be tested both on the final answer and/or on the subtotal answers in the long hand procedure. Several levels of complexity are provided here as well. The third module consists of division problems; one particularly nice feature of the division module is that the long hand division steps can be displayed along with the remainder in order to clearly demonstrate the procedure by which the remainder is derived. Using TEACHER'S AIDE is not merely a drill, but rather a learning experience.

ORDERING INFORMATION

All orders are processed and shipped within 48 hours. Please enclose payment with order and include the appropriate computer information. If paying by VISA or Master Card, include all numbers on card. Purchase orders accepted.

Shipping and Handling Charges
Within North America: Add $2.00
Outside North America: Add 15% (Air Mail)

Delivery
All orders (excluding books) are sent First Class.

Quantity Discounts
Deduct 10% when ordering 3 or more programs. Dealer discount schedules are available upon request.

8" CP/M Disks
Add $2.50 to the listed diskette price for each 8" floppy disk (IBM soft sectored CP/M format). Programs run under Microsoft MBASIC or BASIC-80.

5¼" CP/M Disks
All software available on 8" CP/M disks is also available on 5¼" disks, North Star format.

Ask for **DYNACOMP** programs at your local software dealer. Write for detailed descriptions of these and other programs from **DYNACOMP**.

DYNACOMP, Inc. (Dept. E)
1427 Monroe Avenue
Rochester, New York 14618
24 hour order phone: (716)442-8731 recording
Office phone (9AM-5PM EST): (716)442-8960

VISA

MasterCard

New York State residents please add 7% NYS sales tax.

STATISTICS and ENGINEERING

DIGITAL FILTER (Available for all computers) — **Price: $39.95 Cassette/$43.95 Diskette**
DIGITAL FILTER is a comprehensive data processing program which permits the user to design his own filter function or choose from a menu of filter forms. The filter forms are subsequently converted into non-recursive convolution coefficients which permit rapid data processing. In the explicit design mode the shape of the frequency transfer function is specified by directly entering points along the desired filter curve. In the menu mode, ideal low pass, high pass and bandpass filters may be approximated to varying degrees according to the number of points used in the calculation. These filters may optionally also be smoothed with a Hanning function. In addition, multi-stage Butterworth filters may be selected. Features of DIGITAL FILTER include plotting of the data before and after filtering, as well as display of the chosen filter functions. Also included are convenient data storage, retrieval and editing procedures.

DATA SMOOTHER (Not available for Atari) — **Price: $19.95 Cassette/$23.95 Diskette**
This special data smoothing program may be used to rapidly derive useful information from noisy business and engineering data which are equally spaced. The software features choice in degree and range of fit, as well as smoothed first and second derivative calculation. Also included is automatic plotting of the input data and smoothed results.

FOURIER ANALYZER (Available for all computers) — **Price: $19.95 Cassette/$23.95 Diskette**
Use this program to examine the frequency spectra of limited duration signals. The program features automatic scaling and plotting of the input data and results. Practical applications include the analysis of complicated patterns in such fields as electronics, communications and business.

TFA (Transfer Function Analyzer) — **Price: $19.95 Cassette/$23.95 Diskette**
This is a special software package which may be used to evaluate the transfer functions of systems such as hi-fi amplifiers and filters by examining their response to pulsed inputs. TFA is a major modification of FOURIER ANALYZER and contains an engineering-oriented decibel versus log-frequency plot as well as data editing features. Whereas FOURIER ANALYZER is designed for educational and scientific use, TFA is an engineering tool. Available for all computers.

HARMONIC ANALYZER (Available for all computers) — **Price: $24.95 Cassette/$28.95 Diskette**
HARMONIC ANALYZER was designed for the spectrum analysis of repetitive waveforms. Features include data file generation, editing and storage/retrieval as well as data and spectrum plotting. One particularly unique facility is that the input data need not be equally spaced or in order. The original data is sorted and a cubic spline interpolation is used to create the data file required by the FFT algorithm.

FOURIER ANALYZER, TFA and HARMONIC ANALYZER may be purchased together for a combined price of $49.95 (three cassettes) and $59.95 (three diskettes).

REGRESSION I (Available for all computers) — **Price: $19.95 Cassette/$23.95 Diskette**
REGRESSION I is a unique and exceptionally versatile one-dimensional least squares "polynomial" curve fitting program. Features include very high accuracy; an automatic degree determination option; an extensive internal library of fitting functions; data editing; automatic data, curve and residual plotting; a statistical analysis (eg: standard deviation, correlation coefficient, etc.) and much more. In addition, new fits may be tried without reentering the data. REGRESSION I is certainly the cornerstone program in any data analysis software library.

REGRESSION II (PARAFIT) (Available for all computers) — **Price: $19.95 Cassette/$23.95 Diskette**
PARAFIT is designed to handle those cases in which the parameters are imbedded (possibly nonlinearly) in the fitting function. The user simply inserts the functional form, including the parameters (A(1), A(2), etc.) as one or more BASIC statement lines. Data, results and residuals may be manipulated and plotted as with REGRESSION I. Use REGRESSION I for polynomial fitting, and PARAFIT for those complicated functions.

MULTILINEAR REGRESSION (MLR) (Available for all computers) — **Price: $24.95 Cassette/$28.95 Diskette**
MLR is a professional software package for analyzing data sets containing two or more linearly independent variables. Besides performing the basic regression calculation, this program also provides easy to use data entry, storage, retrieval and editing functions. In addition, the user may interrogate the solution by supplying values for the independent variables. The number of variables and data size is limited only by the available memory.

REGRESSION I, II and MULTILINEAR REGRESSION may be purchased together for $51.95 (three cassettes) or $63.95 (three diskettes).

ANOVA (Not available on Atari cassette or for PET/CBM) — **Price: $39.95 Cassette/$43.95 Diskette**
In the past the ANOVA (analysis of variance) procedure has been limited to the large mainframe computers. Now DYNACOMP has brought the power of this method to small systems. For those conversant with ANOVA, the DYNACOMP software package includes the 1-way, 2-way and N-way procedures. Also provided are the Yates 2^{K-P} factorial designs. For those unfamiliar with ANOVA, do not worry. The accompanying documentation was written in a tutorial fashion (by a professor in the subject) and serves as an excellent introduction to the subject. Accompanying ANOVA is a support program for building the data base. Included are several convenient features including data editing, deleting and appending.

BASIC SCIENTIFIC SUBROUTINES, Volumes 1 and 2 (Not available for Atari)
DYNACOMP is the exclusive distributor for the software keyed to the popular texts *BASIC SCIENTIFIC SUBROUTINES, Volumes 1 and 2* by F. Ruckdeschel (see advertisements in BYTE magazine). These subroutines have been assembled according to chapter. Included with each collection is a menu program which selects and demonstrates each subroutine.

Volume 1
Collection #1: Chapters 2 and 3 - Data and function plotting; complex variables and functions.
Collection #2: Chapter 4 - Extended matrix and vector operations.
Collection #3: Chapters 5 and 6 - Random number generators (Poisson, Gaussian, etc.); series approximations.
Price per collection: $14.95 Cassette/$18.95 Diskette

All three collections are available for $39.95 (three cassettes) and $49.95 (three diskettes).

Volume 2
Collection #1: Chapter 1 - Linear, polynomial, multidimensional, parametric least squares.
Collection #2: Chapter 2 - Series approximation techniques (economization, inversion, reversion, shifting, etc.)
Collection #3: Chapter 3 - Functional approximations by iteration and recursion.
Collection #4: Chapter 4 - CORDIC approximations to trigonometric, hyperbolic, exponential and logarithmic functions.
Collection #5: Chapter 5 - Table interpolation, differentiation and integration (Newton, LaGrange, splines)
Collection #6: Chapter 6 - Methods for finding the real roots of functions.
Collection #7: Chapter 7 - Methods for finding the complex roots of functions.
Collection #8: Chapter 8 - Optimization by steepest descent.
Price per collection: $14.95 Cassette/$18.95 Diskette

All eight collections are available for $99.95 (eight cassettes) and $129.95 (eight diskettes).
Because the texts are a vital part of the documentation, *BASIC SCIENTIFIC SUBROUTINES, Volumes 1 and 2* are available from DYNACOMP:

BASIC SCIENTIFIC SUBROUTINES, Vol 1 (319 pages): $19.95 + 75¢ postage
BASIC SCIENTIFIC SUBROUTINES, Vol 2 (790 pages): $23.95 + $1.50 postage

See reviews in KILOBAUD and Dr. Dobbs.

ROOTS (Available for all computers) — **Price $10.95 Cassette/$14.95 Diskette**
In a nutshell, ROOTS simultaneously determines all the zeroes of a polynomial having real coefficients. There is no limit on the degree of the polynomial, and because the procedure is iterative, the accuracy is generally very good. No initial guesses are required as input, and the calculated roots are substituted back into the polynomial and the residuals displayed.

ACTIVE CIRCUIT ANALYSIS (ACAP) (48K Apple only) — **Price: $25.95 Cassette/$29.95 Diskette**
ACAP is the analog circuit designer's answer to LOGIC SIMULATOR. With ACAP you may analyze the response of an active or passive component circuit (e.g., a transistor amplifier, band pass filter, etc.). The circuit may be probed at equal steps in frequency, and the resulting complex (i.e., real and imaginary) voltages at each component juncture examined. By plotting the magnitude of these voltages, the frequency response of a filter or amplifier may be completely determined with respect to both amplitude and phase. In addition, ACAP prints a statistical analysis of the range of voltage responses which result from tolerance variations in the components. ACAP is easy to learn and use. Simply describe the circuit in terms of the elements and their placement, and execute. Circuit descriptions may be saved onto cassette or diskette to be recalled at a later time for execution or editing. ACAP should be part of every circuit designer's program library.

LOGIC SIMULATOR (Apple only; 48K RAM) — **Price: $24.95 Cassette/$28.95 Diskette**
With LOGIC SIMULATOR you may easily test your complicated digital logic design with respect to given set of inputs to determine how well the circuit will operate. The elements which may be simulated include multiple input AND, OR, NOR, EXOR, EXNOR and NAND gates, as well as inverters, J-K and D flip-flops, and one-shots. The response of the system is available every clock cycle. Inputs may be clocked in with varying clock cycle lengths/displacements and delays may be introduced to probe for glitches and race conditions. At the user's option, a timing diagram for any given set of nodes may be plotted using HIRES graphics. Save your breadboarding until the circuit is checked by LOGIC SIMULATOR.

NUMBERKRUNCHER (TRS-80 only) — **Price: $69.95 Cassette/$73.95 Diskette**
This program is the most complete numerical analysis system available for the TRS-80. It can handle up to 255 data sets, each set having a six character name. It includes complete data editing facilities and convenient data input/output capability. The analyses available are multiple linear regression and correlation determination of residuals, data transformations and extensive graphics generation, including axis naming, and more. The supporting documentation is extremely well written and well organized, and includes appendices which describe the numerical procedures used in the program.

STATSORT (TRS-80 only) — **Price: $39.95 Cassette/$43.95 Diskette**
STATSORT consists of several menu selected programs which allow the user to create (build, edit, merge), format and print files, (machine) sort them on any field, and numerically analyze (maximum, minimum, average, variance, standard deviation) tabulated data. STATSORT is well documented and easy to use. The cassette version can also be employed to create a data type which can be read by the Radio Shack Advanced Statistical Package.

STATTEST (TRS-80 only) — **Price: $19.95 Cassette/$23.95 Diskette**
This is a statistical inference package which helps you make wise decisions in the face of uncertainty. In an interactive fashion you can build and edit data files and test the differences in means, variances and proportions. STATTEST will also perform data analysis as well as do linear correlation and regression. This menu-directed statistical workhorse is rounded out with a chi-square contingency test and a (uniform and normal) random sample generator. The documentation is written by a college professor who guides you through the various tests.

ABOUT DYNACOMP

DYNACOMP is a leading distributor of small system software with sales spanning the world (currently in excess of 50 countries). During the past three years we have greatly enlarged the DYNACOMP product line, but have maintained and improved our high level of quality and customer support. The achievement in quality is apparent from our many repeat customers and the software reviews in such publications as COMPUTRONICS, 80 Software Critique, A.N.A.L.O.G., Creative Computing and Kilobaud. Our customer support is as close as your phone. It is always friendly. The staff is highly trained and always willing to discuss products or give advice.

```
    UFFER ***"
9090 PRINT#PR
9100 GOTO500
10000 REM INPUT A LINE OF TEXT
10010 L$=""
10020 PRINT"$_{LEFT}";
10030 GETA$:IFA$=""THEN10030
10040 IFA$=CHR$(13)THENPRINT" ":RETUR
    N
10045 IFA$="@"THENA$="     ":REM TAB
10050 IFLEN(L$)=(M-5)THENGOSUB22000
10055 IFLEN(L$)>MTHENGOTO15000:REM SP
    ECIFIES MAX LENGTH OF STRI
    NG
10060 IFA$>=SP$ANDA$<=CHR$(95)THEN101
    00
10065 IFA$>=CHR$(161)ANDA$<=CHR$(223)
    THEN10100
10070 IFA$<>DL$THENGOTO10030
10080 IFLEN(L$) >0THENPRINTA$;:L$=LEF
    T$(L$,LEN(L$)-1)
10090 GOTO10020
10100 L$=L$+A$:PRINTA$;:GOTO10020
15000 REM LINE INPUT ERROR
15010 PRINT:PRINT"{REV}ERROR{OFF} LIN
    E TRUNCATED"
15020 RETURN
16000 PRINT"RANGE(LOW,HIGH)-> ";
16010 GOSUB10000:REM INPUT RANGE
16020 LO=1:HI=LL-1:REM DEFAULT LIST A
    LL
16025 L=LEN(L$)
16030 DF=0:IFL=0THENDF=-1:GOTO16150
16040 J=0:FORI=1TOL
16050 A$=MID$(L$,I,1)
16060 IFA$>="0"ANDA$<="9"THEN16090
16070 IFA$="-"THENJ=I:GOTO16090
16080 J=99:I=99
16090 NEXTI
16100 IFJ=99THEN16000
16110 IFJ=0THENLO=VAL(L$):HI=LO:RETUR
    N
16120 IFJ>1THENLO=VAL(LEFT$(L$,J-1))
16130 IFJ<LTHENHI=VAL(RIGHT$(L$,L-J))
16140 IFLO>HITHEN16000
16150 RETURN
17000 PRINT"-LINE#->";
17010 GOSUB10000:REM INPUT LINE#
17020 L=LEN(L$)
17030 IFL=0THEN17000
17040 J=0
17050 FORI=1TOL
17060 A$=MID$(L$,I,1)
17070 IFA$>="0"ANDA$<="9"THEN17090
17080 J=99:I=L
17090 NEXTI
17100 IFJ=99THEN17000
17110 LO=VAL(L$)
17120 RETURN
20000 IFEE=0THENEE=15:OPENEE,8,EE
20010 INPUT#EE,E1,E2$,E3,E4
20020 IFE1=0THENRETURN
20030 PRINTE1;",";E2$;",";E;",";E4
20040 PRINT"*** DISK ERROR ***"
20050 CLOSE2
20060 RETURN
21000 REM SET MARGINS
21010 PRINT:PRINT"{REV}SET MARGIN{OFF
    OFF}"
21020 PRINTTAB(7)"MARGIN SIZE:{DOWN}N
    )ONE
21025 PRINTTAB(19)"S)MALL  (1/2INCH) ~
                          M)ED
    IUM (1.0INCH)
21027 PRINTTAB(19)"L)LARGE (1.5INCH)
21030 PRINTTAB(19)"O)WN DESIGN"
21035 GETMS$:IFMS$=""THEN21035
21050 IFMS$="N"THENM=79:SP=0:GOTO500
21060 IFMS$="S"THENM=74:SP=5:GOTO500
21070 IFMS$="M"THENM=69:SP=10:GOTO500
21075 IFMS$="L"THENM=64:SP=15:GOTO500

21080 IFMS$="O"THENPRINT:PRINT
21085 INPUT"  INCHES FOR LEFT MARGIN ~
                  (10 CHARAC
    TERS/INCH)=";SP
21090 INPUT"  INCHES FOR RIGHT MARGIN
                  (10 CHARAC
    TERS/INCH)=";RM
21100 SP=INT(LM*100)/10:M=INT(79-RM*1
    0):GOTO500
22000 REM BELL
22020 FORII=1TO 5:POKE59467,16:POKE59
    466,85:POKE59464,115:NEXTI
    I
22030 POKE59467,0:POKE59466,255:POKE5
    9464,254
22040 RETURN
```

©

Single Board Computers	Bare Board	Kit	Assm.
80-153 1k RAM 2k EPROM	$19.95	$ 89.95	$110.95
80-280 1k RAM 2k EPROM	$19.95	$119.95	$129.95
81-260 2k RAM 4k EPROM	$39.95	$149.95	$199.95
81-030 4k RAM 16k EPROM	$89.95	*$249.95	$349.95
8085 3 chip system	$24.95		
8088 5 chip system	$29.95		
Apple II boards	**Bare Board**	**Kit**	**Assm.**
79-295 Parallel I/O	$19.95	$59.95	$ 69.95
81-132 A-D 16 Channel	$29.95	$69.95	$ 89.95
80-085 12k EPROM (2716)	$39.95	$49.95	$ 59.95
80-244 EPROM PROGRAMMER	$24.95	$39.95	$ 49.95
80-144 Display add + data	$25.95	$42.95	$ 49.95
80-297 Parallel Printer Interface		$69.95	$ 79.95
Extender Board		$12.95	
81-088 Speech Synthesizer (Votrax SC-01)			$139.90
Other Products	**Bare Board**	**Kit**	**Assm.**
CRT Controller	$39.95		
79-287 A-D and D-A	$19.95	$59.95	$ 79.95
79-282 S S Switch 1 ch.	$ 6.95	$10.95	$ 13.95
79-282 S S Switch 4 ch.	$24.95	$39.95	$ 49.95
80-160 +5 Volt 500 Ma.	$ 8.95	$16.95	$ 20.95
80-161 ±12 Volt 120 Ma.	$ 8.95	$18.95	$ 22.95
81-120 Parallel Input Speech Synthesizer			$149.90

Send for complete Catalog and OEM pricing.
*Partially Populated price.

JOHN BELL ENGINEERING, INC.

ALL PRODUCTS ARE AVAILABLE FROM **JOHN BELL ENGINEERING**
MC P.O. BOX 338 • REDWOOD CITY, CA 94064 VISA
ADD SALES TAX IN CALIFORNIA
ADD 5% SHIPPING & HANDLING • 3% FOR ORDERS OVER $100
10% OUTSIDE U.S.A.

SEND FOR CATALOG **(415) 367-1137**

Mathematics, Basic Skills

Paper Exercises in Arithmetic

For use with
**PET/CBM Computer & Printer*
plus
Compatible Disk System

The Teacher's Aide

Computer programs designed for use by the classroom teacher as a primary source of exercises in mathematics, basic skills. Through simple question and answer, and with the use of only one computer system, a teacher may satisfy all individualized, in-class and homework requirements for drill in arithmetic. Students work directly upon exercise sheets. Difficulty level is easily adjustable. Answers are always provided. 23 programs included, covering integers, decimals, fractions, percent and much more.

On Disk $99.99

Algebra

Explicitly Produced Exercises in Algebra

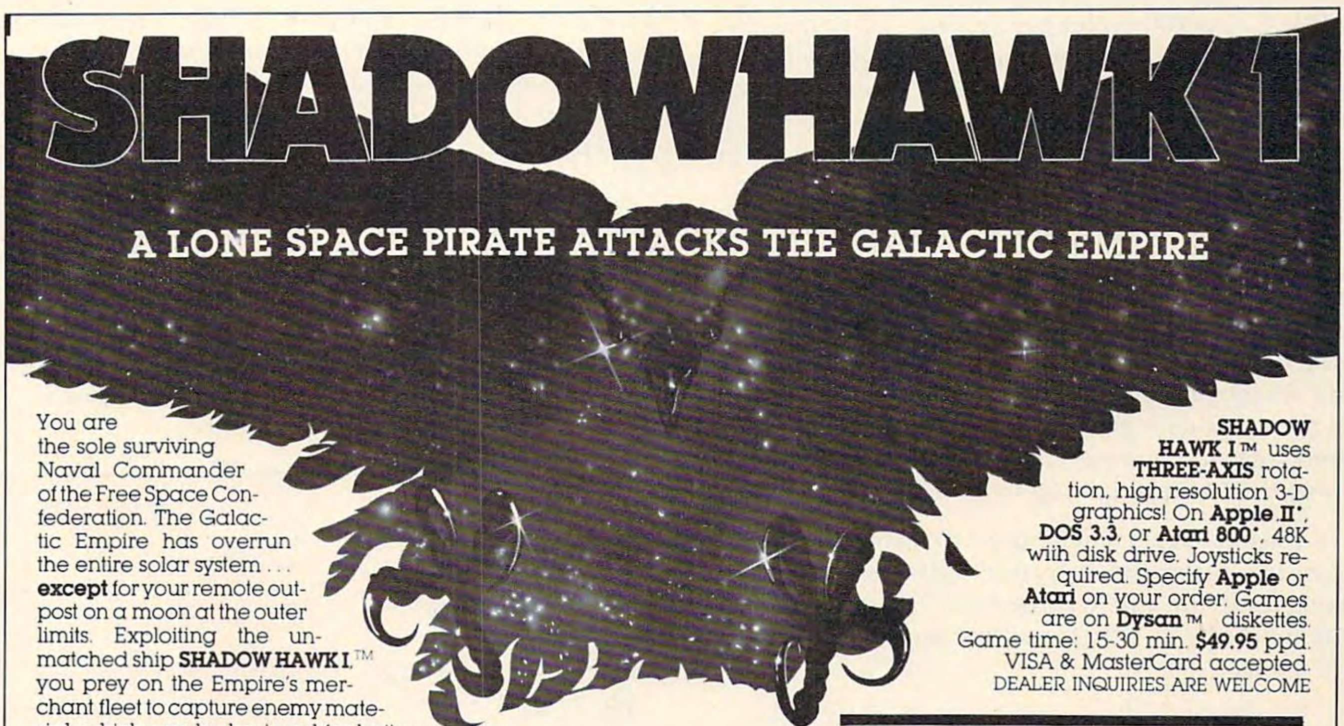
Sixteen programs in linear and fractional equations, simultaneous equations, quadratics, signed and complex number arithmetic.

On Disk $99.99

(Arizona residents please add 4% sales tax.)
Please add $1.50 for postage and handling.

T'Aide Software Company
P.O. Box 65
El Mirage, Arizona 85335

**PET is a trademark of Commodore Business Machines, Inc.*



You are the sole surviving Naval Commander of the Free Space Confederation. The Galactic Empire has overrun the entire solar system . . . **except** for your remote outpost on a moon at the outer limits. Exploiting the unmatched ship **SHADOW HAWK I**,™ you prey on the Empire's merchant fleet to capture enemy material, which can be bartered for better weaponry, shielding, missiles, etc., for **SHADOW HAWK I**.™ But the Empire's interceptors, corvettes, lancers, destroyers, and cruisers are probing the galaxy for you. You must evade them and the deadly battle stations throughout. Your skill is measured by nine rankings, up to **STAR LORD**. Warning: You must be very, very good to reach **STAR LORD** rank! Very good indeed!

SHADOW HAWK I™ uses **THREE-AXIS** rotation, high resolution 3-D graphics! On **Apple II***, **DOS 3.3**, or **Atari 800***, 48K wiih disk drive. Joysticks required. Specify **Apple** or **Atari** on your order. Games are on **Dysan**™ diskettes. Game time: 15-30 min. **$49.95** ppd. VISA & MasterCard accepted.
DEALER INQUIRIES ARE WELCOME

7561 Crater Lake Hwy.
White City, OR 97503
(503) 826-4640

***Apple II** and **Atari 800** are registered trademarks, respectively, of Apple Computer, Inc. and Atari, Inc.

HORIZON SIMULATIONS

Transposition

Janet Whitehead
Saint John, N.B., Canada

Editor's Note: This transposition algorithm works on both Atari and PET/CBM. We hope to see some exciting harmony or other musical applications as suggested by Janet at the end of her article. — RTM

On first observation, the sound command SOUND V,N,T,L seemed to have a lack of pattern for the sequence of numbers representing the note *N*. I recalled a question in a high school mathematics book that stated that the frequency of A above middle C was 440 cycles per second. To obtain the next higher note on a musical scale multiply by $2^{1/12}$, for a lower note divide by $2^{1/12}$; thus one can find the frequency of each of the twelve notes in an octave. (I am considering each octave as containing twelve notes, the five black notes as well as the seven white ones on a piano).

This, I thought, must be the basis for the sequence of numbers used for notes in Atari BASIC. As the value of *N* is from 0 to 255 (one byte), the frequency was not used, but some multiple of it. As frequency increases the pitch increases, but the value of *N* decreases as the pitch increases. Therefore, to increase the pitch you divide *N* by $2^{1/12}$ instead of multiplying by $2^{1/12}$.

This property can be used to transpose music. To raise a composition by one-half tone, one only needs to divide the *N* value by $2^{1/12}$, for a full tone divide by $(2^{1/12})^2$, for a tone and one-half by $(2^{1/12})^3$ etc.

To illustrate these properties, here are two simple programs. Program 1 prints the sequence of number used for *N* in the sound command. To obtain the sequence in *Atari BASIC* by Albrecht et al. an original *N* value of 259 was used instead of 255. If you find that these give values for *N* which produce sharp or flat tones, just change the 259.

Program 1:

Line 40: T1 finds successive values of $(2^{1/12})^0$, $(2^{1/12})^1$, $(2^{1/12})^2$ etc.

Line 50: Successive one-half tones, *N* values, are calculated.

Line 60: The results are printed.

Notice that, for notes one octave apart, the ratio of the two *N* values is 2:1.

Program 2:

This plays a few bars of music to illustrate how a piece of music can be transposed through one octave.

Line 100 – Sets the voice to 0, the tone as 10, and the loudness at 10.

Line 110-120 – M is the value of $2^{1/12}$ and T1 is the number of half-tones to transpose the music.

Line 140 – M1 calculates the value of $(2^{1/12})^{T1}$ which is the factor by which each *N* value must be divided to raise a piece by T1 half tones.

Line 150 – *A* holds the original value of *N*, and *B* indicates the length of time it is to be played. *A* is then transposed the desired number of half tones. The note is then played.

Line 170 – As the program plays the few bars of music through each successive half tone for one octave, this line increases the amount *N* is to be transposed one half a tone. As the data must be read each time, it needs to be RESTOREd.

Caution: If you exceed an *N* value of 255 in your transposition, you will get a very high pitched note. Only one byte is used for *N*, so 257 would be 1.

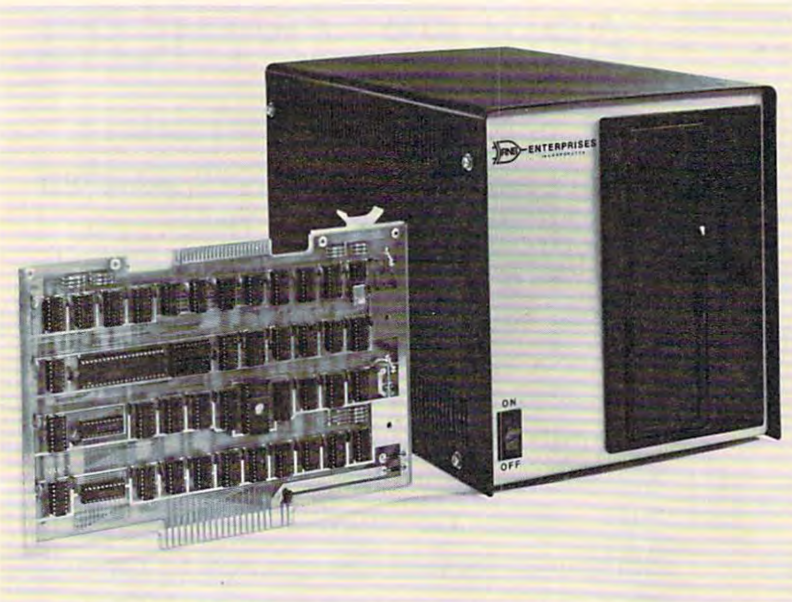
Perhaps some reader can expand on this to play chords or generate harmony. Knowing very little about music, I will have to leave that task to someone else.

Program 1.

```
10 M=2^(1/12)
20 T=0
30 FOR I=1 TO 40
40 T1=M^T
50 N=259/T1
60 PRINT I,INT(N+0.5)
70 T=T+1
80 NEXT I
90 END
```

VAK-7 8″ FLOPPY DISK SYSTEM

FOR AIM-65, SYM-1



The VAK-7 Disk System was specifically designed for use with AIM-65 and SYM-1 Microcomputer Systems. The VAK-7 is a complete full size (8″) Floppy Disk System. This system will Read, Write and Format IBM Single and Dual Density diskettes. The VAK-7 is available with both Single and Dual Sided Disk Drives. Each Disk Drive comes with its own cabinet and Power Supply. The VAK-7 can handle up to 4 disk drives, totaling more than 4.98 Megabytes of storage.

The VAK-7 Disk System incorporates both advanced hardware and innovative software designs. The addition of the VAK-7 produces a very powerful and useful computer system. Unlike most other disk systems, there is no requirement for the user to provide RAM to hold the Disk Operating System software. No valuable time is wasted loading in the DOS from cassette tape, because the VAK-7 DOS software is in onboard ROM. The VAK-7 is located above the 32K boundary (8000 HEX), leaving the user with a full 32K bytes of contiguous address space for his own use.

AIM-65—Allows the user to save and load object code thru the AIM Monitor; to load, save, and append Text thru the AIM Editor; to load, save, and append Basic Programs thru the BASIC INTERPRETER; to assemble directly from disk single or multiple file programs.

SYM-1—Allows the user to save and load Files for use with the SYM Monitor, SYM Basic, and RAE-1.

ADDITIONAL COMMANDS:

ACTIVATE A DELETED FILE	FORMAT A DISK
COMPRESS A DISK	DELETE A DISK FILE
RENAME A DISK FILE	INITIALIZE A DISK
COPY A DISK	LIST CATALOG

SPECIFICATIONS:

- Completely assembled, tested, and burned in.
- Occupies address 8000-8FFF for AIM-65, $9000-9FFF for SYM-1, or $E000-EFFF for KIM-1.
- IBM Format; Single Density (128 bytes/sector); Dual Density (256,512, or 1024 bytes/sector).
- All ICs are in sockets.
- Fully buffered address and data bus.
- Standard KIM-4* BUS (both electrical pin-out and card size).
- Designed for use with a regulated power supply, but has provisions for adding regulators for use with an unregulated power supply.
- Dimensions: Board—10″ wide x 7″ high (including card-edge). Cabinet—9.25″ wide x 10″ high x 16″ deep.
- Power Requirements: +5v DC @ 2 Amps.
  117 AC 60Hz @ 2 Amps.
- Shipping Weight: 25 lbs.

**KIM-4 is a product of MOS Technology/C.B.M.*

PRICING:

VAK-7 **$1,299.00**
Controller and One Single-Sided Drive

VAK-7A **$599.00**
Additional Single-Sided Drive with Cabinet and Power Supply

VAK-7B **$1,599.00**
Controller and One Dual-Sided Drive

VAK-7C **$899.00**
Additional Dual-Sided Drive with Cabinet and Power Supply

CALL OR WRITE FOR FREIGHT CHARGES

RNB ENTERPRISES INCORPORATED

4030 N. 27th Avenue, Suite D
Phoenix, AZ 85017
(602) 265-7564

Program 2: Atari Version

```
100 V=0:T=10:L=10
110 M=2^(1/12)
120 T1=0
130 ? :? "HAPPY BIRTHDAY"
140 M1=M^T1
150 FOR X=1 TO 26:READ A,B:A=INT(A/M1+0
5):SOUND V,A,T,L
160 FOR I=1 TO B:NEXT I:SOUND V,0,0,0:NE
XT X
170 T1=T1+1
180 IF T1<12 THEN RESTORE 190:GOTO 140
185 END
190 DATA 122,64,122,64,109,128,122,128,9
2,128,97,256
200 DATA 122,64,122,64,109,128,122,128,8
2,128,92,256
210 DATA 122,64,122,64,61,128,73,128
220 DATA 92,64,92,64,97,128,109,128
230 DATA 69,64,69,64,73,128,92,128,82,12
8,92,256
```

Program 3: CBM Version

```
100 POKE59467,16:POKE59466,15:POKE5
    9464,0:S=59464
110 M=2^(1/12)
120 T1=0
130 PRINT"HAPPY BIRTHDAY
140 M1=M^T1
150 FORX=1TO26:READA,B:A=INT(A/M1+.
    5):POKES,A
160 FORI=1TOB+B/2:NEXTI:POKES,0:NEX
    TX
165 REM VALUE OF B IS INCREASED HER
    E TO EQUALIZE THE DIFFEREN
    CES IN SPEED
168 REM BETWEEN THE CBM/PET AND THE
     ATARI
170 T1=T1+1
180 IFT1<12THENRESTORE:GOTO140
185 POKE59467,0:END
190 DATA122,64,122,64,109,128,122,1
    28,92,128,97,256
200 DATA122,64,122,64,109,128,122,1
    28,82,128,92,256
210 DATA122,64,122,64,61,128,73,128
220 DATA92,64,92,64,97,128,109,128
230 DATA69,64,69,64,73,128,92,128,8
    2,128,92,256
```

©

HUNTINGTON COMPUTING

ONE OF THE WORLD'S LARGEST INVENTORIES

Now Selling Atari®, PET®, TRS-80® Software

Atari®

Poker Solitaire (cass.)	$14.95	now	**$12.74**
Gomoku (cass.)	$19.95	now	**$16.94**
Reversi (cass.)	$19.95	now	**$16.94**
Cypher Bowl (cass.)	$29.95	now	**$25.44**
Rescue at Rigel (cass.)	$29.95	now	**$25.44**
Star Warrior (cass.)	$39.95	now	**$33.94**
Invasion Orion (cass.)	$24.95	now	**$21.24**
Datestones of Ryn (cass.)	$19.95	now	**$16.94**
Conflict 2500 (cass.)	$15.00	now	**$12.74**
Empire of the Overmind (cass.)	$35.00	now	**$29.64**
Tanktics (cass.)	$24.00	now	**$20.44**
Atari* Mailing List (disk)	$24.95	now	**$21.24**
Atari* Character Generator (disk)	$19.95	now	**$16.94**
Text Wizard (disk)	$99.95	now	**$84.44**
Atari* Character Gen. (cass.)	$15.95	now	**$13.54**
Le Stick	$39.95	now	**$33.94**
Checker King (cass.)	$19.95	now	**$16.94**
MicroChess (cass.)	$19.95	now	**$16.94**
Survival/Adventure (disk)	$24.95	now	**$21.24**
3-D Supergraphics (disk)	$39.95	now	**$33.94**
3-D Supergraphics (cass.)	$39.95	now	**$33.94**
Mind-Bogglers I (disk)	$19.95	now	**$16.94**
Mind-Bogglers I (cass.)	$15.95	now	**$13.54**
VersaWriter Graphics Tablet	$300.00	now	**$254.44**
Hidden Words	$17.50	now	**$15.44**
Spatial Relations	$17.50	now	**$15.44**
Word-Scramble	$15.00	now	**$13.54**
Preschool Fun	$15.00	now	**$13.54**
Fastgammon (cass.)	$19.95	now	**$16.94**
Assembler (cass.)	$24.95	now	**$21.14**
6502 Disassembler (cass.)	$11.95	now	**$10.14**
6502 Disassembler (disk)	$14.95	now	**$12.64**
Tank Trap (cass.)	$11.95	now	**$10.14**
Tank Trap (disk)	$14.95	now	**$12.64**
Tari Trek (cass.)	$11.95	now	**$10.14**
QS Forth (disk)	$79.95	now	**$67.94**
Starbase Hyperion (disk)	$22.95	now	**$19.44**
Name That Song (cass.)	$14.95	now	**$12.64**
Jaw Breaker (disk)	$29.95	now	**$25.44**
Pornopoly (disk)	$29.95	now	**$25.44**
The Broker (CCI) (disk)	$99.95	now	**$84.94**
Super Modem Pak (CCI) (Disk)	$49.95	now	**$42.44**
Atari* Snuff (CCI) (disk)	$29.95	now	**$25.44**
Utility Man (CCI) (disk)	$99.95	now	**$84.94**
Tanktics (cass)	$24.00	now	**$20.44**
Fantasyland (disk)	$59.95	now	**$50.94**
Empire of the Overmind (cass)	$30.00	now	**$25.44**
Bridge 2.0 (cass)	$17.95	now	**$15.24**
Nominoes Jigsaw (cass) Dynacomp	$16.95	now	**$14.34**
Intruder Alert (cass)	$16.95	now	**$14.34**
Alpha Fighter (disk)		now	**$16.04**
Compu-read (disk)	$29.95	now	**$25.44**
Letter Perfect (disk)	$150.00	now	**$127.44**
Sammy Sea Serpent (Cass) PDI	$16.95	now	**$14.34**
Cribbage (Thesis) (Cass)	$15.00	now	**$13.54**
Visicalc	$200.00	now	**$149.00**
Kross 'N Quotes PDI (Cass)	$16.95	now	**$14.34**
Star Raiders (cart)	$39.95	now	**$33.94**
Stock Charting	$24.95	now	**$21.14**
Adventure International	**10% - 15% off sug. retail**		
Atari* Safari (CDS) (disk)			**$33.94**
Atari* Safari (CDS) (cass)			**$25.44**
Typing Tutor (IMAGE) (Cass)			
16K Ram (Atari*)			
Computation (Thesis) (cass)	$15.00	now	**$13.94**

Pet®

AVALON HILL GAME COMPANY			
B-1 Nuclear Bomber (cass.)	$15.00	now	**$12.77**
Midway Campaign (cass.)	$15.00	now	**$12.77**
No. Atlantic Convoy Raider (cass.)	$15.	now	**$12.77**
Nukewar (cass.)	$15.00	now	**$12.77**
Conflict 2500 (cass.)	$15.00	now	**$12.77**
Planet Miners (cass.)	$15.00	now	**$12.77**
Computer Acquire (cass.)	$20.00	now	**$16.97**
Lords of Karma (cass.)	$20.00	now	**$16.97**
AUTOMATED SIMULATIONS			
Introductory 3-Pack (disk) (Rescue, Morloc's, and Datestones)	$49.95	now	**$39.97**
Rescue at Rigel (cass.)	$29.95	now	**$25.47**
Temple of Apshai (cass.)	$39.95	now	**$33.97**
Hellfire Warrior (cass.)	$39.95	now	**$33.97**
Starfleet Orion (cass.)	$24.95	now	**$21.27**
Invasion Orion (cass.)	$24.95	now	**$21.27**
Morloc's Tower (cass.)	$19.95	now	**$16.97**
Datestones of Ryn (cass.)	$19.95	now	**$16.97**
PERSONAL SOFTWARE			
VisiCalc (disk)	$199.95	now	**$167.77**
Checker King (cass.)	$19.95	now	**$16.97**
Gammon Gambler (cass.)	$19.95	now	**$16.97**
MicroChess (cass.)	$19.95	now	**$16.97**
Bridge Partner (cass.)	$19.95	now	**$16.97**
Time Trek (cass.)	$19.95	now	**$16.97**
UNITED SOFTWARE OF AMERICA			
KRAM (disk)	$99.95	now	**$84.97**
Super KRAM (disk)	$175.00	now	**$148.77**
Request (disk)	$225.00	now	**$191.27**
Thinker (disk)	$495.00	now	**$420.77**
Space Intruders (cass.)	$19.95	now	**$16.97**
All MICRO-ED			**10% Off List**
All Microcomputer Workshops			**15% Off List**
VIC SOFTWARE			
Addcom Missile Commander			**$16.66**
Channel Vic Data Logger			**13.16**
MMA Star Command			**$8.96**
TIS Basic Programming I			**$17.46**

VISICALC
Special for Pet, Atari & Apple
Regular $200.00 List
$149.00

TRS-80®

BIG FIVE SOFTWARE			
Super Nova (cass.)	$15.95	now	**$13.58**
Galaxy Invasion (cass.)	$15.95	now	**$13.58**
Attack Force (cass.)	$15.95	now	**$13.58**
Cosmic Fighter (cass.)	$15.95	now	**$13.58**
Meteor Mission II (cass.)	$15.95	now	**$13.58**
BRODERBUND SOFTWARE			
Galactic Trilogy (disk)	$39.95	now	**$33.98**
Galactic Empire (cass.)	$14.95	now	**$12.68**
Galactic Trader (cass.)	$14.95	now	**$12.68**
Galactic Revolution (cass.)	$14.95	now	**$12.68**
Tawala's Last Redoubt (cass.)	$19.95	now	**$16.98**
DATASOFT			
Iago (disk)	$24.95	now	**$21.18**
Football Classics (disk)	$24.95	now	**$21.18**
Arcade-80 (disk)	$24.95	now	**$21.18**
Iago (cass.)	$19.95	now	**$16.98**
Football Classics (cass.)	$19.95	now	**$16.98**
Arcade-80 (cass.)	$19.95	now	**$16.98**
Sigmon (COLOR) (cass.)	$29.95	now	**$25.38**
SECS (COLOR) (cass.)	$29.95	now	**$25.38**
ACORN SOFTWARE			
Invaders From Space (disk)	$20.95	now	**$17.78**
Duel-N-Droids (disk)	$20.95	now	**$17.78**
Pinball (disk)	$20.95	now	**$17.78**
Pigskin (disk)	$20.95	now	**$17.78**
Quad (disk)	$20.95	now	**$17.78**
Basketball (disk)	$20.95	now	**$17.78**
Gammon Challenger (disk)	$20.95	now	**$17.78**
Everest Explorer (disk)	$20.95	now	**$17.78**
SuperScript (disk)	$29.95	now	**$25.38**
System Savers (cass.)	$14.95	now	**$12.68**
Invaders From Space (cass.)	$14.95	now	**$12.68**
Duel-N-Droids (cass.)	$14.95	now	**$12.68**
Pinball (cass.)	$14.95	now	**$12.68**
Pigskin (cass.)	$14.95	now	**$12.68**
Quad (cass.)	$14.95	now	**$12.68**
Basketball (cass.)	$14.95	now	**$12.68**
Gammon Challenger (cass.)	$14.95	now	**$12.68**
Everest Explorer (cass.)	$14.95	now	**$12.68**
All Adventure International			**10% to 15% Off List**
All Automated Simulations			**10% to 15% Off List**
All Avalon Hill			**10% to 15% Off List**
All Hayden			**10% to 15% Off List**
All Microsoft			**10% to 15% Off List**

Apple®

See full page of Apple® products elsewhere in this magazine.

Gorgon	$39.95	now	**$33.99**
Word Star	$375.00	now	**$289.00**
Mail Merge	$125.00	now	**$106.19**
Super Sort	$200.00	now	**$169.99**
VisiCalc 3.3	$200.00	now	**$149.00**
Wurst of Huntington Computing			**$19.99**
Nibble Express	$12.95	now	**$11.99**
Soft Porn Adventure	$29.95	now	**$25.39**
Time Lord	$29.95	now	**$25.39**
French Hangman	$29.95	now	**$25.39**
Alicia-Sp. bilingual reader	$29.95	now	**$25.39**
H&H Stock Trader	$190.00	now	**$161.49**
Grow (CIA)	$35.00	now	**$31.49**
All Edu-Ware			**10% to 15% Off List**
VersaCalc	$100.00	now	**$84.99**
Hebrew	$60.00	now	**$50.99**
All Serendipity			**10% to 15% Off List**
All Sirius			**15% Off List**
All Sybex Courses			**10% to 15% Off List**
Win at the Races	$39.95	now	**$33.99**
Disk Prep	$25.00	now	**$21.19**
PLE Chip	$60.00	now	**$50.99**

We maintain a huge inventory of software for Apple* and hardware. Call us toll free (outside Calif.) for the latest programs. We also stock a large supply of computer books. Visit us in person at our new 3300-square foot store at 1945 South Dairy in Corcoran, Calif.

Call Toll-Free
800-344-4111
(Outside California)

APPLE is a registered trademark of Apple Computer, Inc.

HUNTINGTON COMPUTING
Post Office Box 1235
Corcoran, California 93212

Order by Phone 800-344-4111
In California (209) 992-5411

Call us for any Software not listed here. Most Prices are 15% Off List

VISA

master charge

We take MasterCard or VISA (Include card # and expiration date) California residents add 6% tax. Include $2.00 for postage. Foreign and hardware extra. Send for free catalog. Prices subject to change.

TELECOMMUNICATIONS

Why 300 Baud?

Michael E. Day
Chief Engineer, Edge Technology

Time plays a very important role in telecommunications. Every industry builds up its own group of words and terms to define those things it deals with that cannot be readily described with everyday language. The telecommunications field has more than its share of buzz words. Close analysis shows that many of these words deal with time and time-related things.

Time is important in all aspects of telecommunication, from the transmission of a single bit of information, to the overall system performance. Although there are no exact limits to the use of time as it relates to telecommunication, there are practical limits and deterioration effects which must be considered when designing a system.

There are many factors which must be taken into account when designing a telecommunications system. The purpose of the system generally determines what methods will be used. Of major importance is the amount of information flow that will occur. This will generally determine the minimum acceptable system communication speed.

Short Messages Permit Slow Speeds

If the message to be sent is relatively short, and little or no response is expected, then very slow system speeds can be acceptable. Western Union makes use of this on their TELEX network which operates at 6.7 characters per second. A slow system speed has the advantage of using only a small amount of the communications bandwidth. This means that more systems can be installed on the same communications link and can thus reduce the per-user cost of the communications system. The TELEX system serves users who must be sure that their messages get to the parties to which they were sent in a short period of time, but at low cost. (In some ways it can be cheaper to send a TELEX than it is to send a letter).

Another low cost system structure is the message forwarding system. Here higher communication speeds are allowed, but direct communication with the target party is not allowed. Instead the message is built up at a location close to the sending party and then transmitted to the target party at a time when the cost is at the lowest rate. In the case of *packet switchers,* it is sent when a communications link becomes available.

When a large amount of data is to be sent, the speed of the communications link becomes important. Transmission at higher speeds requires a larger communications bandwidth. Because of this, there is an associated increase in the cost of the communications link as well as the equipment required.

Between these extremes lie general information communications systems involving a low to medium amount of data transfer. Often this occurs in conjunction with user interaction with the data flow. This is the area where most computer use occurs.

Although the exact system configurations change as technology changes, the general structure of use remains relatively constant. Any change in one area of the telecommunications field tends to affect the other areas in some way.

The Three Common Computer Modems

There are three types of modems that the average computerist encounters: 1. The 103 style modem (the most commonly used). 2. The 202 style modem (far less common, but it has the advantage of higher speed at a reasonable cost, but with an increase in complexity of use). 3. Finally, the newer 212A style modem combines the higher speed of the 202 with the ease of use of the 103 (but costs more).

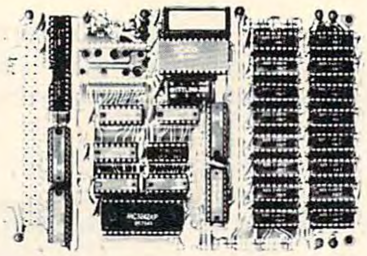
The 202 modem is sort of an ugly duckling in the telecommunications field. It has the advantage of higher speed than the 103, yet it can be built at a lower cost. The problem comes from its complexity of use.

The 202 is a half duplex modem. This means that it can only transmit in one direction at a time and, thus, requires some amount of computer control over its use. If the intended use requires high speed at lost cost, this is normally considered an acceptable trade off. If the computer control requirement is not acceptable, the higher cost 212A modem is used to achieve high speed without computer control. If cost is important, the 103 is used.

Although the 103 is more expensive to make on a price/performance basis, there is a demand for the low cost 103's. They are mass produced and mass distributed at low profit margins, and are often of lower quality.

Acoustic Versus Direct Connect

The modems come in two major classifications, the acoustic type and the direct connect type. Until recently, the acoustic was the most popular with the general user. This was due largely to regulations which inhibited the direct connection of modems to the phone lines. There are still many regulations which inhibit this, but they have been reduced to the point that the general user at least can consider



BETA 32K BYTE EXPANDABLE RAM FOR 6502 AND 6800 SYSTEMS
AIM 65 KIM SYM PET S44-BUS

- Plug compatible with the AIM-65/SYM expansion connector by using a right angle connector (supplied).
- Memory board edge connector plugs into the 6800 S44 bus.
- Connects to PET using an adaptor cable.
- Uses +5V only, supplied from the host computer.
- Full documentation. Assembled and tested boards are guaranteed for one full year. Purchase price is fully refundable if board is returned undamaged within 14 days.

Assembled with 32K RAM	$349.00
& Tested with 16K RAM	329.00
Bare board, manual & hard-to-get parts...	99.00
PET interface kit. Connects the 32K RAM board to a 4K or 8K PET	$ 69.00

See our full-page ad in BYTE and INTERFACE AGE

wabash

8" or 5¼" flexible diskettes certified 100% error free with manufacturer's 5-year limited warranty on all 8" media. Soft-sectored in boxes of 10. 5¼" available in 10-sector.

(Add $3.00 for plastic library cases)

8" single sided, single density	$27.50
8" single sided, double density	35.50
8" double sided, double density	45.50
5¼" single sided, single density	27.50
5¼" single sided, double density	29.50
5¼" single sided, double density, 10-sector	$29.50

TERMS: Minimum order $15.00. Minimum shipping and handling $3.00. Calif. residents add 6% sales tax. Cash, checks, Mastercard, Visa and purchase orders from qualified firms are accepted. (Please allow two weeks for personal checks to clear before shipment.) Product availability and pricing subject to change without notice.

INTERNATIONAL ORDERS: *Add 15% to purchase price for all orders. Minimum shipping charge is $20.00. Orders with insufficient funds will be delayed. Excess funds will be returned with your order. All prices are U.S. only.*

PAPER TIGER PRINTERS

IDS 460G 9x9 Dot Matrix Printer	$890.00
IDS 560G Wide Carriage Printer	1099.00

TERMINALS

ADDS Viewpoint	$569.00
TeleVideo 910	579.00
TeleVideo 912C	679.00
TeleVideo 920C	729.00
TeleVideo 950	929.00

8" DISK DRIVES

Shugart 801R	$399.00
NEC FD1160 (double sided)	569.00

DYNAMIC RAMS

4116 (200ns) .... set of 8	$24.00
4164 (64Kx1)	$18.00

BETA COMPUTER DEVICES
1230 W. COLLINS AVE.
ORANGE, CA 92668
(714) 633-7280

VISA

HAVE YOUR CAKE AND EAT IT TOO

IT'S A PIECE OF CAKE TO CONNECT AN Actek TRIX I INTERFACE TO YOUR OLIVETTI PRAXIS 30 OR 35 CORRECTING ELECTRONIC TYPEWRITER. DAISY WHEEL QUALITY AT DOT MATRIX PRICES!

- 10+ CPS
- 15 MINUTE INSTALLATION
- HALF SPACE JUSTIFICATION
- CABLE REMOVES IN SECONDS
- TYPEWRITER FUNCT. UNIMPAIRED
- AVAILABLE NOW: ATARI & APPLE
- OTHER DIRECT CONNECTIONS AND RS232 AVAILABLE SOON
- PRINT AND PRINT #N OPERATE
- NO INTERFACE NEEDED: USES FRONT CONNECTOR -ATARI USES CONTROLLER PLUG -APPLE
- PRICE: $215 - APPLE ADD $10 †BASED ON WARDS PR30 PRICE
- TYPEWRITER AND SERVICE WIDELY AVAILABLE

Actek
12225 SW 2nd/SUITE 200 -E
P.O.B. CCC
BEAVERTON, OR 97075

EPROM PROGRAMMER KITS

Shown assembled. EPROM not included.

For single supply 2516, 2716 & 2758 EPROMS. Verify erased. Program - entire or partial. Auto verify after programming. Transfer contents to RAM for modifying or duplicating.

Select Documentation for:	Use with:
6502	6820 PIA or 6522 VIA
6800	6820 PIA
6809	6820 PIA
8080/8085/Z80	8255 PPI

Documentation includes schematic, instructions for construction, check-out and use, and software listing for specified MPU.

Complete kit of parts (includes Welcon 24 pin ZIF socket)	$ 39.95
Bare PC board and Documentation	$ 15.00
Software listings for additional MPUs (with purchase of Kit or PC board)	$ 5.00

Ordering: Specify MPU. Add $1 for postage & handling. Overseas add $2. Ariz. residents add 5% sales tax. Check or Money Orders for $U.S. Phone orders ship C.O.D. 602-839-8902.

Micro Technical Products
814 W. Keating Ave., Dept. M • Mesa, Arizona 85202

it as an option. It is actually possible to build a direct connect modem less expensively than the acoustic version, but, until the regulations are further reduced and demand increases, this will probably not happen.

Acoustic versions of the 202 modem are essentially nonexistent. The main reason for this is the telephone. The telephone uses a carbon microphone to convert sound into electricity. The carbon microphone is very rugged, cheap to build, has a high gain factor, and has natural filtering characteristics in the frequency range of the human voice. Unfortunately, it is also very noisy and disruptive to the signal that it converts.

The 202 cannot work within this environment. The 212A modem fares a little better because the part of the signal it uses for data transfer suffers a minimal amount of disruption from the carbon microphone. The 103, however, works reasonably well with the carbon microphone. This is due to the narrow bandwidth used for transmission, as well as the high redundancy in the signal used. Although some 103 modems are capable of operation at speeds of up to 600 baud, the disruption of the carbon microphone tends to limit the speed to 300 baud. If the phone system is in good workig order, it is possible to operate as high as 450 baud. Alternatively, if the system is in poor condition, it may require that operation be reduced to 200 baud, 150 baud, or even as low as 110 baud. Generally speeds below 110 baud do not actually achieve any improvement in reliability of operation on the 103.

If problems are encountered, or higher speeds are desired, the direct connect modems are generally required. If only a slight improvement is needed, some degree of improvement can sometimes be obtained by replacing the carbon microphone with a capacitor (or condenser) microphone (sometimes referred to as a *supermike*). The capacitor microphone is more expensive to make than the carbon, but does not disrupt the signal. There is one disadvantage to most capacitor microphones. It requires power to operate, and it gets this power from the phone line. If another phone comes on the line, it can steal the power away from the capacitor microphone and inhibit it from working. This is particularly true if the other phone has a carbon microphone in it which tends to drop the power well below the capacitor microphone's normal operating level.

The 103 type modem has become a standard for medium speed communications over the telephone network not because someone felt it should be, or because someone made it so. It became the standard because it was the optimal solution to the problem at hand. ©

Reading The Status Register

Bob Sullivan
Oak Park, IL

Here is a way to quickly analyze the flags in the 6502 Status Register. When you are working in machine language, the branch (BNE, BCC, etc.) instructions automatically check these flags for you and make their "decisions" based on the condition (or status) of a flag. However, there are times when *you* need to analyze the Status Register. When debugging, for example, you might place a BReaK instruction to stop the program and allow you to examine the condition of a flag.

Each flag is a bit within the Status Register byte. When you see that the SR has $F1 in it, how quickly can you determine that the Overflow flag is set? The table below does the job:

CARRY BIT-right digit
Set if odd: 1,3,5,7,9,B,D,F
Clear if even: 2,4,6,8,A,C,E

ZERO BIT-right digit
Set if: 2,3,6,7,A,B,E,F
Clear if: 0,1,4,5,8,9,C,D

INTERRUPT BIT-right digit
Set if: 4,5,6,7,C,D,E,F
Clear if: 0,1,2,3,8,9,A,B

DECIMAL MODE-right digit
Set if: 8,9,A,B,C,D,E,F
Clear if: 0,1,2,3,4,5,6,7

BREAK-left digit
Set if odd: 1,3,5,7,9,B,D,F
Clear if even: 0,2,4,6,8,A,C,E

OVERFLOW
Set if: $C0 to $FF
Clear if: $00 to BF

SIGN BIT
Positive: $00 to $7F
Negative: $80 to $FF

©

Book Review:

Some Common Basic Programs

Jim Butterfield
Toronto, Canada

This book, a collection of 76 short programs, is by Lon Poole and Mary Borchers. There are editions for several popular machines, with additional authorships: Carroll Donahue contributed to the PET/CBM edition, Karl Koessel to the TRS-80 edition, and David M. Castlewitz to the Apple II edition. There's an Atari edition, too. Cassette and diskette versions are available.

The book is well established. It may be a definitive answer to the question: What serious things can I do on my small computer without using data files?

Three Major Areas

The programs break into three major sections, with a few programs left over at the end. It might have been useful to have broken the material up into chapters. The first twenty programs are financial; the next twenty-seven, mathematical; and the third section contains twenty statistical programs. Nine miscellaneous programs are tacked on the end.

The financial programs are quite well commented, and often include optional coding for performing monthly (as opposed to yearly) calculations. Serious users will probably want to combine several short programs together; the book does not show how to do this, but it is not a difficult task. Users should not use these programs as the final word; financial methods differ in different organizations and the serious user will hopefully know what modifications will be needed for his circumstances.

The mathematical programs are somewhat more cryptic; it is expected that the user is quite familiar with the material. For example, three Integration programs are given; the user is expected to choose Simpson's Rule, Trapezoidal Rule, or Gaussian Quadrature according to his estimate of which will suit his needs. Fair enough; the book does not attempt to be a text, but just gives the relevant coding.

The statistical section is a mixed bag in terms of user levels. Some programs such as Average Growth Rate are easy for the naive user to understand; others such as F-Distribution require a comprehension of statistics.

Up Front

None of the programs is huge: all are easy to type in. Worked-through examples allow the user to check that his coding is, indeed, correct.

For the non-mathematical computer owner, the programs may seem to be rather obscure. In this case, the book may serve as a challenge and an indication of the resources he can tap if he wants. Such things as linear programming and regressions can be remarkably powerful tools to use in business ...if you know that they are there. It might be useful to see a companion guide to this book, explaining just how effective some of these mathematical techniques can be, even to the small user. The book doesn't try to do this: it just plunks down the coding.

There may be debate on whether the best mathematical and programming techniques are used in all cases; to me, this doesn't matter. The first thing to do is to find any way of approaching a problem. After you have one way, you can look for better ways; but finding that first one can be hard.

Son-Of-Some-Common...?

There are a couple of books that look like sequels to the well-established SCBP. *Practical Basic Programs,* edited by Lon Poole, gives more programs in a similar vein. *Science and Engineering Programs,* Apple II Edition edited by John Heilborn, delves more deeply into mathematics and statistics. The books are similar in organization to *Some Common Basic Programs,* but the programs are longer and there are fewer of them. Readers who found the first book useful will undoubtedly want to go after its successors.

Some Common Basic Programs is not a textbook. It doesn't teach you what to do with the programs. But it does give you working programs with documentation.

For those who know the methodologies, it will be a useful reference. For those who don't, it may open up new horizons: things that you didn't know a computer can do. In that case, you'll need to look elsewhere to learn the principles of the new technology. It's often an education to discover the existence of things you don't know. Or at least the start of an education.

Some Common Basic Programs. Lon Poole and Mary Borchers. Osborne/McGraw Hill. Berkeley. (1981) ©

NEW PRINTERS. NEW PERIPHERALS. SAME OLD RELIABLE QUALITY AND VALUE.

1982 will find more OEM's, businesses, dealers and personal computer users turning to MICROTEK than ever before.

TekWriter-1

80 Column Dot Matrix Printer
(Formerly BYTEWRITER-1)

The Tekwriter-1 printer is, dollar for dollar, the finest value in the industry. And we've proved it by comparing the Tekwriter-1 to the Epson MX-80. Our print speed is 14 lines per minute faster, our life expectancy is longer, the character sets are the same, and the interface, warranty and printhead replacement cost are all identical.* But the biggest difference is the price. The Tekwriter-1 is about $300 less.

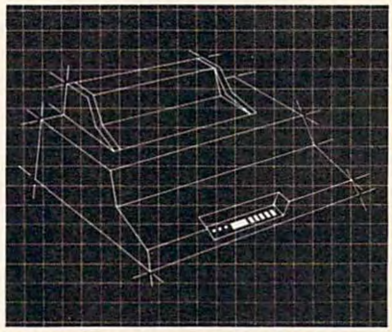
Our extensive testing has proved that the Tekwriter-1 interfaces problem-free to most parallel Centronics and serial (RS-232) computers.

The Tekwriter-1 is tough to beat for performance and quality.

*Data Source: Epson MX-80 Operation Manual

Parallel $349
Serial $389

TekWriter-2



NEW! 80/132 Column Dot Matrix Printer

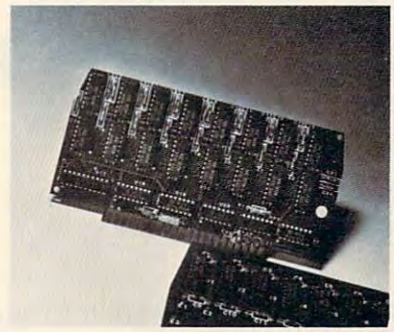
The Tekwriter-2 is perfectly suited to personal, business or OEM applications. Tekwriter-2 is designed to accept single sheet, roll or pin feed paper. It has a 9-wire dot matrix impact print head which produces crisp characters and has underlining capability. The printer is manufactured to run extremely quietly even while operating at peak output levels.

Tekwriter-2 is especially well suited to handle an abundance of text entry because of its data buffer expansion capability to 25K. This ability makes it an efficient graphics generator.

Parallel interface (Centronics type). Interfaces all models of TRS-80, Apple, and Atari 400/800, and most computers with Centronics printer interface.

$695

Peripherals



16K Apple Memory Board

Expands Apple II to 64K RAM Memory. Works with MICROSOFT Z-80 Softcard, Apple PASCAL and Visicalc software.

16K-32K Atari Memory Board

4116 RAM (200NS) Compatible with Atari 400/800

Parallel/Seriel Data Buffer Converter

Interfaces with most computers and printers on the market today. Switch selectable parallel or serial input/parallel or serial output. Data buffer 2K standard — expandable to 62K. Serial I/O BAUD rates switch independently selectable.

Atari Parallel or Serial Printer Cables

Pre-tested. Centronics or (RS-232) compatible.

APPLE PARALLEL INTERFACE CARD

Quantity and OEM discounts available.

Continuing our quest for excellence.

MICROTEK inc.

MICROTEK
9514 Chesapeake Drive
San Diego, CA 92123
(714) 278-0633
Outside CA call
Toll Free (800) 854-1081
TWX. 910-335-1269

Multitask: A Realtime, Multitasking Operating System Emulator

Hal Bredbenner
Raleigh, NC

Most home applications of microprocessors are very basic and straightforward with the micros spending 99% of their existence in loops waiting for a key depression or an interrupt. The majority of systems with interrupts use them just for utilitarian functions such as screen refreshes, clock increments, and keyboard scans. If you have a PET, Apple, Atari, etc., you know how the machine really cannot do more than one operation at a time, and it also has a hard time responding quickly to outside inputs. The BASIC program described by this article emulates a way to allow many seemingly concurrent operations to occur with a fast system response time to the outside world. Keep in mind that this is an emulator. To actually be realized, the concept would have to be written in machine code form; however, the model shows the concept on the screen where it can be analyzed and easily understood.

There are two terms which should be understood before we continue. The first is realtime. An ideal realtime system is one that responds to a changed input immediately. This response will be a change in an output condition or an internal recognition of the input change. Ideally, this response is immediate. However, in reality, some time elapses before the realtime system can respond. The faster the response time, the more efficient the system.

The second term to be defined is multitasking. An ideal multitasking system is one that allows multiple operations to take place simultaneously in one system. Obviously, a single micro can do only one thing at a time, but through scheduling of desired tasks and assigning priorities to each operation, the appearance of more than one action at a time is accomplished.

For example, let's design a hypothetical simple realtime, multitasking system. The system will be a home security system that logs all its data and, upon command from a keyboard, produces a paper tape output of this data. It also has a CRT display and a battery back-up power system. The system must scan various inputs from the house and control outputs which would be interfaced to lights, alarms, and an electric fence. Since the electric fence poses a safety problem (for the good guys!), an Emergency Stop input would be needed. Internal DC power supplies should be monitored to detect system tampering and, if incoming AC power is out of tolerance (for example, a brownout), an alarm should be sounded. Prior to the back-up power system running out, the system should be shut down. When properly programmed, the system should monitor and accomplish all these tasks concurrently (if required) with a fast response time.

Logically breaking down the system software requirements, we can see that some tasks need to be done on a regular basis while others only need to be done upon command from the keyboard or other input. The regular tasks we will call Auto Rescheduling tasks. The tasks are called:

I0 DRIVER	**Reads inputs and writes outputs**
AC PWR CK	**Tests incoming AC power**
DC PWR CK	**Tests internal DC power**
READ KYBD	**Scans the system keyboard**
REFRESH	**Refreshes the CRT display**

These Auto Rescheduling tasks are regularly performed by the system and scheduled to be done again once they have been completed.

The remaining tasks are to be performed only when an outside input requires them. In our emulation, we can schedule one of these tasks by pressing its number on the numeric keys of the keyboard. In the hypothetical system, they would be initiated by the power fail detect circuitry, the system keyboard, or perhaps the Emergency Stop button. These tasks are called:

E-STOP	**Starts an emergency sequence**
PWR FAIL	**Initiates power down sequence**
MOVE FILE	**Transfers data to output buffer**
MEM TEST	**Exercise and test RAM memory**
PUNCH DATA	**Produces a paper tape**

The system would require at least these ten operations and, through the use of the emulator, we can see how the tasks are prioritized, scheduled, and executed.

In our system, we would require one master interrupt signal to drive the entire process. Each time this interrupt occurs, the operating system would perform the same actions. The first action is to read the status of all the system inputs and write, from a RAM buffer area, any new output data.

NEW COMMODORE ADD-ONS
RAM-ROM:
GETS RID OF SAFING ROM

***MX-910 CBM/PET RAM/ROM;**
Allows multi ROM protected programs using the same socket to be put onto diskette/cassette, no need to insert protect ROM in socket after initial load, eliminates need for ROM switch box, write protect in software, decoded for dual ROM socket usage, 4K expandable to 8K, easy internal CBM installation: $119.95

***MX-232 CBM/PET TO RS-232C INTERFACE:**
Low cost, bidirectional, 50 to 19,200 baud rate, full modem controls, parity allows for two RS-232C CBM ports, installs easily inside CBM: $199.95

SX-100 IEEE-488/PET MODEM SOFTWARE:
Best 810 modem software, by 8010 developer, works with Source/Micronet/CBM to disk/CBM to CBM: $79.95

***MX-200 IEEE-488/PET PARITY MODEM/SOFTWARE**
Talk to a host computer requiring parity, all features of SX232: $399.95

***MX-903 CBM/PET AUTORESET:**
Easily installed reset circuit: $19.95

ECX COMPUTER COMPANY
2678 NORTH MAIN ST.
WALNUT CREEK, CA 94596
(415)944-9277

CLEAR STAND
FOR
ATARI® 400/800*
APPLE®II**

- Beautifully polished high impact acrylic
- Holds up to 15" Dia. TV/monitor at eye level
- Open design allows maximum air flow
- Easy access to top sides & rear of processor

Dealer Inquiries Welcome

Send order with payment to:
BYTM Systems, Inc.
389 Fifth Ave. (Suite 400)
New York, N.Y. 10016

Allow 4-6 weeks delivery

Mail the coupon today

Send ______ Clear Stand @ $59.95
Add N.Y. Sales Tax for N.Y. Delivery

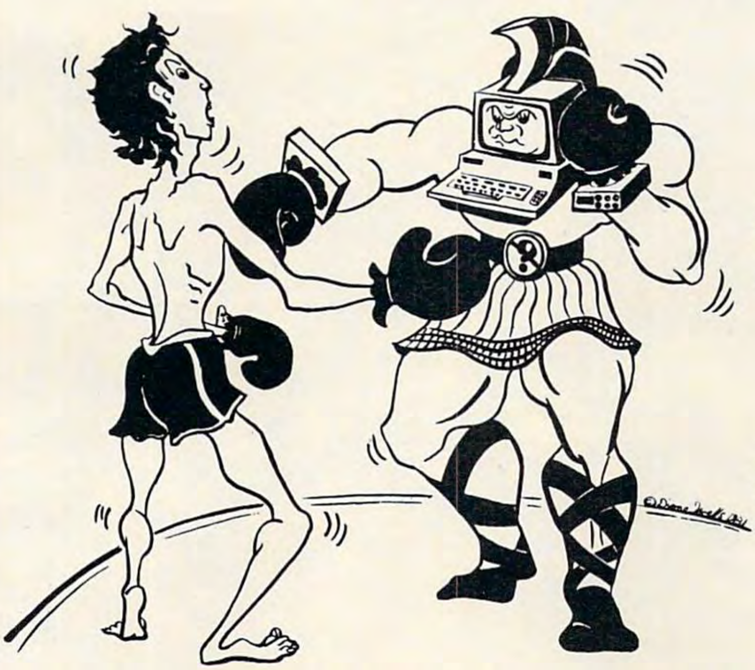
Ship to: ____________________
Address: ____________________
City: __________ State: ______ Zip: ______

*TM of Atari, Inc.
**TM of Apple Computers, Inc.

DON'T ASK PRESENTS:

ABUSE

For the APPLE II/II+ and the ATARI 400/800



Match wits with your computer in an insult war! With **ABUSE** your computer becomes a slightly demented, smart-aleck insult-exchanger.

- **Millions** of **different** insults.
- Understands and responds to user input.
- Hybrid BASIC/machine language program.
- Game feature: become a Master of **ABUSE.**
- **Many** surprises to discover.

Release your aggressions! Inflict **ABUSE** on anyone who's got it coming!

APPLE: REQUIRES 48K RAM AND APPLESOFT

ATARI: REQUIRES 40K RAM AND BASIC CARTRIDGE

At your computer store or send **$19.95** + $2.00 handling to:
DON'T ASK – 2265 Westwood Blvd. B-150 – Los Angeles, CA 90064 – (213) 397-8811

Calif. residents add 6% sales tax.

Dealer inquiries welcome

ATARI is a trademark of ATARI INC.
APPLE is a trademark of APPLE COMPUTER, INC.

ATARI
48K — w/disk

CRYSTALWARE

THE FINEST IN
FANTASY GAME SOFTWARE

APPLE
48K — w/disk

TRS-80
48 — w/disk

"Having previewed over fifty of your competitors' games, I can assure you that your use of scrolling far exceeds anything I've seen for the Atari and, of course, for the Apple. I'm very impressed by the dedication and quality that your company exhibits by virtue of this demo." **David Sosna** — Associate Producer, Universal Pictures

Crystal has done its best to become the Porsche of the computer game industry. New scrolling techniques, video disk games, a real-life fantasyland — our mad programmers toil onward with little food or sleep to produce some incredible firsts in the microcomputer world. If you are an unappreciated genius and want to join our staff to help create the world of tomorrow today, give me a call. Our magazine *Crystal Vison* will within the next month have a circulation of 80,000 and we look forward very soon to producing our first full length motion picture. I'd like to thank my friends at Votrax and Axlon for giving us the tools (128K RAM for Atari and a vocal text synthesizer) to truly produce some programming miracles.

★ ★ ★ NEW RELEASES ★ ★ ★

THE CRYPT — One evening you awake at sunset to find yourself in what appears to be an endless cemetery. Although defenseless, you must somehow find your way out or perish from the hideous assaults of flesh eating zombies, rats, vampires, werewolves, and other repulsive monstrosities. To escape you may have to descend into the catacombs beneath the cemetery. This game is a little different from the others of our series because we use a lot of static graphics to set the mood. It is similar in some respects (without any copying intended) to those of our friends at On-Line who produce excellent static graphic adventures. You must use all your common sense and a great deal of courage to escape from this perilous adventure alive. We have made it so nearly impossible that the first player to do it successfully will receive a $200.00 prize. **$49.95 2 disks**

QUEST FOR POWER by Mark Benioff — An extraordinary game with the adventure and magic of Arthurian legend. Join Galahad as he leaves Camelot in search of the Scroll of Truth. Explore the treacherous depths of the Caves of Somerset, visit the medieval city of Essex. Along the way you will meet powerful wizards and great prophets. The villages of Sunderland and Leeds dot your path. Somewhere in an evil castle called Skenfirth, lurks the devil himself, while the Evil Giant Gogmogo, hungry for human prey, roams the forests. In Fantasyland tradition we include 64 full screens of hires scrolling and some sensational graphic and animation sequences. Well worth the **$39.95 1 disk**; enjoyable to all ages.

★ ★ ★ GALACTIC EXPEDITION ★ ★ ★

The year is 3021, almost 100 years since the expedition to the Sands of Mars has returned. The Starship Herman now rests quietly in the Zikon Museum in New Brisbane. It's nearly 80 years since World War III, the Ames Research Center celebrates its 150th anniversary, and you stand at the unveiling of a truly technological wonder — the first ion-propelled vessel, saucer shaped Lady Joanne, its viewport of pure diamond, its hull of synthetic emeralds. The Martian glyphs of the Meshim and those of Lemuria have now been deciphered and it appears that a much greater mystery is about to unravel. 7 planes and 7 doors — 7 guardians and 7 candles. 7 strange new worlds await the ultimate adventurer to unlock a timeless secret. The starship may seem strange and unfamiliar to our veteran adventurers, faced with its marvelous new technology; this craft must be flown by constant monitoring of ion stabilizers. During your galactic expedition you are surrounded by the flickering heavens, beset by meteor showers and time-warps. Each unique world holds one of the 7 keys to unlock the Great Mystery. The games all run off the Main Module which also is a game unto itself.

From Earth to Moon — On the Moon's dark side lie entrances to caverns extending to the moon's hollow core which contains a timeless secret. Here live a race of burrowing creatures, who have built vast earthen cities with storehouses full of precious stones. Gravity is extremely critical and you must use all your skills to manually land your craft. This first Master Disk contains the dos needed to run additional scenarios. Its price is **$39.95** and includes 64 screens of Hires graphics.

Mists of Venus — On Venus' ever hot surface are endless jungles and swamps. The air is unbreathable and spacesuits and oxygen must be carried. This world is especially treacherous with all sorts of loathsome creatures and hardly any place dry enough to land your ship. Beneath the green seas our adventurer may find the second key to solving the Mystery. **$29.95** (must have Master Disk to run)

Planet Herman — It is hard to tell where Herman's atmosphere ends and the surface begins. Much of this adventure will have the feeling of a starship submarine. Navigating around Herman is very dangerous but with a computer on board Lady Joanne it may be just possible. This senario costs **$29.95** and needs the Master to run.

The Asteroid Belt — Every play something oids. A combination of the best machine language sub-routines of our new Crystaloids with a fast moving adven ure game. Penal colonies, lurking pirates, and some unusual forms of scavenger life exist here. It's difficult to travel in the Asteroid Belt without getting blown up. Perhaps you should find some expert help by rescuing a pilot, who is also a sentenced thief or murderer, from one of the penal colonies. There are places for trading and you may wish to indulge yourself with a visit to the sensual Pleasure Planet. **$29.95** (needs Master Disk)

Uranus - World of Ice — A freezing place with nights of —200° F. Bring along Thermasuits, as well as some Laars with which to battle the Grungik, a 12 foot tall relative of Big Foot, fond of human flesh. Uranus also has a secret inner labyrinth with tropical flora and fauna. However, the King of the Ice Planet, Norion may have his own idea about your trespassing. Without proper clothing, weapons and supplies, your stay here may be very exciting and very short. **$29.95** (needs Master Disk to run)

Jupiter - World of Dwarfs — How would it feel to weigh 300 or so lbs.? A trip to Jupiter should fill you in fast. There is a particularly interesting red spot on Jupiter and a curious set of moons. Picking up some antigravs will help. Landing should really tax your energies. In the Jupiterian atmosphere, you fall fast! Be prepared to use 10 times the normal amount of fuel. Better find the 6th key quickly before your fuel and food are exhausted. **$29.95** (needs Master Disk)

The Crystal Planet - You will have to embark on this final portion of your expedition ignorant of what you may encounter here on this mysterious planet, excepting that the 7th world holds the ultimate key to winning the contest. **$29.95** (needs Master Disk)

The Contest — To the Winner with the highest score, who solves the mystery by November of 1982 will go $5000.00 in cash. Good Luck!

★ ★ ★ ★ ★ ★

GLAMIS CASTLE — According to ancient legend and records this castle is one of the most haunted sites in Great Britain. One Lady Glamis, known to be in league with the devil, liked to send out a destructive demon to harrass the townspeople. She finally was burnt at the stake on Castle Hill, cursing as she died all future generations of the Lyon family. Her demon still seems to haunt that spot, murdering the curious who stray up to Castle Hill after dark. The curse stipulated that each succeeding generation would have at least one child, often female, who would be a vampire. When an heir comes of age, there is a secret ceremony in which the heir, his father, and the steward take crowbars and chip away plaster concealing a hidden chamber, known only to them, that Earl Patie used when he gambled with the devil. Another tradition says that a creature, half-man, half-beast stalks the passages in the walls of Glamis to insure the fulfilling of the curse. The mystery, of course, is to determine the location of this secret chamber. Our game, occupying 2 disks, will have as exact a replica of the castle as possible. It's definitely one of a kind! And we will be offering a $500 prize to the first person daring enough to solve the centuries-old mystery of Glamis Castle. **$49.95 2 disks.**

Crystal has many other fine fantasy and space games. For a copy of CRYSTAL VISION which includes a complete catalog please send $3.00 to the address below.

Our order lines are open 24 hrs. a day 7 days a week.

(408) 778-2966 **CRYSTAL COMPUTER** **17120 Monterey Rd., Morgan Hill, CA 95037**

master charge VISA

The next action of the operating system (OS) will be to determine if the inputs demand the scheduling of any tasks. For example, let's suppose the person using the system entered a request for a memory test from the keyboard by pressing an "M." The OS would recognize this as a MEM TEST request and read, from a lookup table, three things about the task: the task location, its execution time, and a basic priority value.

The three pieces of data about the task and the time of its request would then be entered into a queue table. This queue table is a list of all the tasks that are scheduled in the system at any one time. When all the required new tasks are entered into the queue, the OS then assigns a calculated priority to each task. This is done in various ways, some extremely complex; however, the most basic way is to make the calculated priority a function of both the task's basic priority and its elapsed time in the queue. This simple calculation is done in our emulator. After a new calculated priority value has been entered into the queue for each pending task, the OS then looks again at the calculated priorities and merely selects the most urgent task, the one with the highest priority, to be performed. The OS then passes control to that task. The selected task runs on the micro until it finishes or another interrupt occurs and, in either case, a return to the OS is made and a new task begun.

An OS like the one we have just described will respond quickly to any input and that is the main design goal for a realtime system.

Notice that if a task is running and another task is calculated to have a greater priority, the first task can be suspended while the more urgent task runs. This can happen if one task has been waiting in the queue for some time while another executing task has most of the processor time. Again, this determination of task suspension can be extremely complex, yet in our emulator it is made solely on the basis of the calculated priorities of the tasks. It also should be noted that because of memory resources, scheduling queues are limited in size. Because of this, any tasks that are requested when the queue is full are ignored and must be requested again later.

To be actually designed in a system, this type of OS requires careful planning. One consideration is the frequency of occurrence of the master interrupt signal. The maximum response time to any input would be one cycle of the interrupt and yet too fast an interrupt will tend to bind the processor down with OS tasks instead of real life tasks. Another consideration is that separate stack pointers should be kept for all active, scheduled, or suspended tasks so that, at any time, resumption of that task's execution will not be ruined by some other task's dealings. The use of a RAM buffer for output storage allows any programmed task to see what is in an output port and, if that RAM output buffer location is changed or modified during next interrupt cycle, the OS will automatically write that new data to the port. This is a good way of synchronizing output and also preventing interference between different program modules.

This emulator program is a model of the operating system required by the hypothetical home security system we talked of earlier. The emulator is written in BASIC and obviously is not as fast as the machine code OS would be. However, the basic design of the OS is graphically shown and is simple to understand. When run, the emulator displays the active task, its time of execution, the average time of response to those tasks in the queue, the entire scheduling queue, and a list of the available system tasks (See Figure 1). The Auto Rescheduling tasks are initially placed in the queue and the OS begins highlighting the active task as it "executes." At any time, by pressing one of the available task numbers (0-3 and 7), a new task will be added to the queue and serviced as the OS permits. The queue in the emulator will hold only 10 entries and then a "QUEUE FULL" response will be given to further inputs. Notice that the Auto Rescheduling tasks are added again to the queue as they are completed. Line 2335 of the program is the algorithm used to calculate the priority of the tasks in the queue. Experimentation with different prioritizing schemes will produce some very interesting results. Try your own algorithm and compare the average response times.

The realtime, multitasking Operating System Emulator given here requires less than 4K of memory and can be run on any Commodore system (most other systems would only require the modification of the cursor positioning characters). It is an excellent tutorial program that graphically shows how some of the most complex OS actually do what they do. Microcomputers can become as powerful as minis and mainframes with this kind of programming. I urge you to try to accomplish this type of OS in machine code. The resulting power of the microprocessor would be amazing.

Program 1.

```
110 REM  MULTITASKER EMULATOR
111 REM  WRITTEN BY HAL BREDBENNER
120 REM
130 FORTN=ØTO9
140 READAR(TN),EX(TN),PR(TN),TN$(TN)
150 NEXTTN
160 DATAØ,1ØØ,1,"E-STOP   ",Ø,1ØØ,1,"~
   ~PWR FAIL ",Ø,9Ø,3,MOVE FILE
170 DATAØ,5Ø,8,"MEM TEST ",1,3Ø,3,IO ~
   ~DRIVER,1,3Ø,9,AC PWR CK
```

```
180 DATA1,3Ø,9,DC PWR CK,Ø,9Ø,7,PUNCH~
   ~DATA,1,3Ø,2,READ KYBD
190 DATA1,3Ø,3,"REFRESH  "
200 REM
210 FORQN=ØTO4
220 READQT(QN),QT$(QN),QE(QN),QP(QN),~
   ~QA(QN),QW(QN)
230 PT(QN)=INT((QW(QN)/QP(QN))*1ØØ)
240 NEXTQN
250 DATA9,"REFRESH  ",3Ø,2,1,Ø,8,READ~
   ~ KYBD,3Ø,2,1,Ø
260 DATA6,DC PWR CK,3Ø,9,1,Ø,5,AC PWR~
   ~ CK,3Ø,9,1,Ø
270 DATA4,IO DRIVER,3Ø,2,1,Ø
290 PRINT"{CLEAR}{REV}MULTI-TASKING O~
   ~PERATING SYSTEM EMULATOR"
300 PRINT"{Ø3 DOWN}{REV}SCHEDULER{OFF~
   ~}
310 PRINT"{14 DOWN}{REV}AVAILABLE TAS~
   ~KS...{DOWN}
320 PRINT"Ø- E-STOP    1-PWR FAIL     ~
   ~2-MOVE FILE
330 PRINT"3-MEM TEST   4-IO DRIVER*   ~
   ~5-AC PWR CK*
340 PRINT"6-DC PWR CK* 7-PUNCH DATA   ~
   ~8-READ KYBD*
345 PRINT"9-REFRESH*     *-AUTO RESCHE~
   ~DULING{Ø2 UP}
350 PRINT"{HOME}{Ø5 DOWN}{REV}TASK NA~
   ~ME{OFF} {REV}TIME LEFT{OFF} {REV}~
   ~PRIORITY{OFF} {REV}TIME IN QUEUE{~
   ~OFF}
1000 REM-------------------
1010 REM REAL OPERATING SYSTEM AREA
1020 REM-------------------
1025 Q=Ø
1030 GOSUB28ØØ:REM ADVANCE ACTIVE TAS~
   ~K
1040 GOSUB25ØØ:GOSUB27ØØ:REM ADVANCE ~
   ~QUEUE
1070 GOSUB25ØØ:REM GET TASK AND ADD
1080 IFQE(Q)<1ØTHEN13ØØ
1085 GOSUB24ØØ:REM PACK QUEUE TABLE
1090 GOSUB25ØØ:GOSUB23ØØ:REM DETERMIN~
   ~E HIGHEST
1100 IFPT(XP)>PT(Q)THENQ=XP
1110 GOSUB25ØØ:GOSUB2ØØØ:GOTO1Ø3Ø
1300 REM-----------------
1310 REM DELETE FINISHED TASK
1320 REM-----------------
1322 CC=CC+1:TC=TC+QW(Q):AC=INT(TC/CC~
   ~*1ØØ)/1ØØ
1325 FT$=QT$(Q)+" "+STR$(AC)+"MSEC."
1330 QT$(Q)=""
1335 IFQA(Q)=1THENTN=QT(Q):GOSUB257Ø
1337 GOTO1Ø85
2000 REM-----------------------
2010 REM DISPLAY QUEUE
```

```
2020 REM------------------------
2025 GOSUB2100
2030 PRINT"{HOME}{05 DOWN}
2050 FORQN=0TO9
2060 IFQT$(QN)=""THENPRINT"                    ~
~                                        ":GOTO~
~2080
2065 PRINTTAB(10)"                            ~
~              {UP}"
2066 PT(QN)=INT((QW(QN)/QP(QN))*100)
2067 IFQN=QTHENX$="{REV}":GOTO2070
2068 X$="{OFF}"
2070 PRINTX$QT$(QN)"{OFF}","  "QE(QN),~
~"  "PT(QN),"    "QW(QN)
2080 NEXTQN
2085 PRINTQF$
2090 RETURN
2100 REM------------------------
2110 REM DISPLAY ACTIVE TASK
2120 REM------------------------
2130 PRINT"{HOME}{02 DOWN}{REV}ACTIVE~
~ TASK:{OFF} "QT$(Q)"   {REV}TIME L~
~EFT:{OFF}          {09 LEFT}"QE(Q)~
~
2135 PRINTTAB(10)"{REV}AVERAGE RESPON~
~SE:{OFF}         {07 LEFT}"AC
2140 RETURN
2200 REM------------------------
2210 REM    DISPLAY TIME
2220 REM------------------------
2230 PRINT"{HOME}{04 DOWN}"TAB(27)"{R~
~EV}TIME:{OFF}";
2240 PRINTLEFT$(TI$,2)+":"+MID$(TI$,3~
~,2)+":"+RIGHT$(TI$,2)
2250 RETURN
2300 REM------------------------
2310 REM DETERMINE HIGHEST Q PRIORITY~
~
2320 REM------------------------
2330 X=0:FORQN=0TO9
2333 IFQP(QN)=0THENGOTO2350
2335 PT(QN)=INT((QW(QN)/QP(QN))*100)
2340 IFPT(QN)>XTHENX=PT(QN):XP=QN
2350 NEXTQN
2360 REM EXIT WITH XP=HIGHEST PRIORIT~
~Y
2370 RETURN
2400 REM------------------------
2410 REM  PACK QUEUE TABLE
2420 REM------------------------
2430 FORQN=0TO9
2440 IFQT$(QN)=""THENQT(QN)=0:QE(QN)=~
~0:QP(QN)=0:QA(QN)=0:PT(QN)=0:QW(Q~
~N)=0
2450 NEXTQN
2460 FORQN=0TO9
2470 IFQT$(QN)<>""THENNEXTQN:GOTO2490~
~
2480 QT$(QN)=QT$(QN+1):QT(QN)=QT(QN+1~
~):QE(QN)=QE(QN+1):QP(QN)=QP(QN+1)~
~
2485 QA(QN)=QA(QN+1):QW(QN)=QW(QN+1)
2486 QT$(QN+1)=""
2487 NEXTQN
2490 REM TABLE IS NOW PACKED
2495 RETURN
2500 REM------------------------
2510 REM  GET TASK AND ADD TO QUEUE
2520 REM------------------------
2525 QF$="                    "
2530 GETX$:IFX$=""THENRETURN
2535 IFX$="4"ORX$="5"ORX$="6"ORX$="8"~
~ORX$="9"THENRETURN
2540 REM ADD TASK TO QUEUE
2550 IFVAL(X$)<0ORVAL(X$)>9THENRETURN~
~
2560 TN=VAL(X$)
2570 GOSUB2400:REM PACK QUEUE TABLE
2580 I=0
2590 IFI>9THENQF$="{REV}QUEUE FULL!{O~
~FF}":GOTO2620
2595 IFQT$(I)<>""THENI=I+1:GOTO2590
2600 QT(I)=TN:QT$(I)=TN$(TN):QE(I)=EX~
~(TN):QP(I)=PR(TN):QA(I)=AR(TN)
2610 QW(I)=0:PT(I)=INT((QW(I)/QP(I))*~
~100)
2620 GOSUB2000
2630 RETURN
2700 REM------------------------
2710 REM INC QUEUE AND PRIORITIES
2720 REM------------------------
2730 FORQN=0TO9
2740 IFQT$(QN)=""THENNEXTQN:GOTO2780
2750 QW(QN)=QW(QN)+1
2760 PT(QN)=INT((QW(QN)/QP(QN))*100)
2770 NEXTQN
2780 RETURN
2800 REM------------------------
2810 REM ADVANCE ACTIVE TASK
2820 REM------------------------
2830 QE(Q)=QE(Q)-10
2850 PT(Q)=INT((QW(Q)/QP(Q))*100)
2860 RETURN
READY.
```

Figure 1.

```
MULTI-TASKING OPERATING SYSTEM EMULATOR

ACTIVE TASK: MOVE FILE   TIME LEFT: 60

SCHEDULER AVERAGE RESPONSE: 12.57
TASK NAME  TIME LEFT  PRIORITY  TIME IN QUE

DC PWR CK    30         344        31
AC PWR CK    30         344        31
MEM TEST     50         262        21
MOVE FILE    60         666        20
PUNCHDATA    90         285        20
READ KYBD    30         450         9
REFRESH      30         200         6
IO DRIVER    30         100         3


AVAILABLE TASKS:

0- E-STOP    1-PWR FAIL    2-MOVE FILE
3-MEM TEST   4-IO DRIVER*  5-AC PWR CK*
6-DC PWR CK* 7-PUNCH DATA  8-READ KYBD*
9-REFRESH*    *-AUTO RESCHEDULING
```

©

THE APPLE® GAZETTE

Plotting Polar Graphs With The Apple II

Marvin L. De Jong
The School of the Ozarks
Pt. Lookout, MO 65726

You do not need long programs to make a computer perform a useful task in teaching mathematics. One of the more arduous tasks in trigonometry or analytic geometry is graphing functions in polar coordinates. For many polar curves, this task takes a lot of time, and not much learning takes place. On the other hand, it is an ideal task for the computer, the program to plot a polar graph is easily understood by the students, and it gives them a tool with which they can experiment with many graphs. Program 1 shows the simplest possible version. We shall discuss it shortly, but first here is a brief explanation of what we are trying to accomplish.

Suppose we have a relation between R, the distance from a point called the *pole,* and θ (Greek symbol theta), the angle measured counterclockwise from the *polar axis.* The pole is analogous to the *origin* in X-Y Cartesian coordinates, and the polar axis lies along the *X-axis.* The relation between R and θ is usually described by an equation of the form

$$R = F(\theta).$$

The equation

$$R = 90*SIN(2*\theta)$$

is one example. Refer to Figure 1 for an illustration of some of these concepts, including a graph of the equation R=90*SIN(2*θ), called a *four-leaved rose.*

The key to using a computer to graph polar coordinates is the transformation formulas

$$X = R*COS(\theta)$$
$$Y = R*SIN(\theta)$$

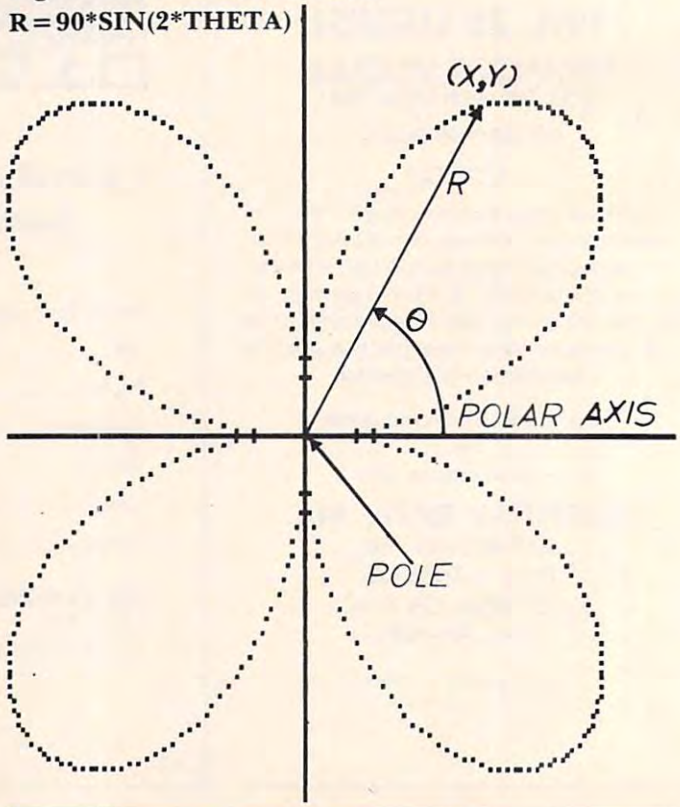
and, of course, the computer's ability to perform a PLOT X,Y instruction.

A "bare bones" approach to plotting polar graphs is given in Program 1. The student inputs the starting angle and the angle at which the graph is to end. These angles are in degrees. Line 30 initializes the HIRES mode with text on the lower part of the screen of the video monitor. Line 60 converts the angle to radians (pi radians = 180°).

Line 70 in Program 1 is the equation to be graphed. The entire program may be left unchanged while line 70 is modified to graph a large variety of polar functions.

Line 90 and 100 convert the polar coordinates (R,θ) to X-Y coordinates. Note that since the origin of the Apple II coordinate system is in the upper left-hand corner of the screen, we have *translated* it so the origin of our coordinate system is at (85,85). Furthermore, since Y is positive *downward* on the Apple, and we would prefer the more traditional "Y positive upward" convention, we use a negative sign in the Y-transformation equation. The results are plotted with the instruction on line 120. The instruction on line 130 increments the angle by one degree. Points will be continued to be plotted until

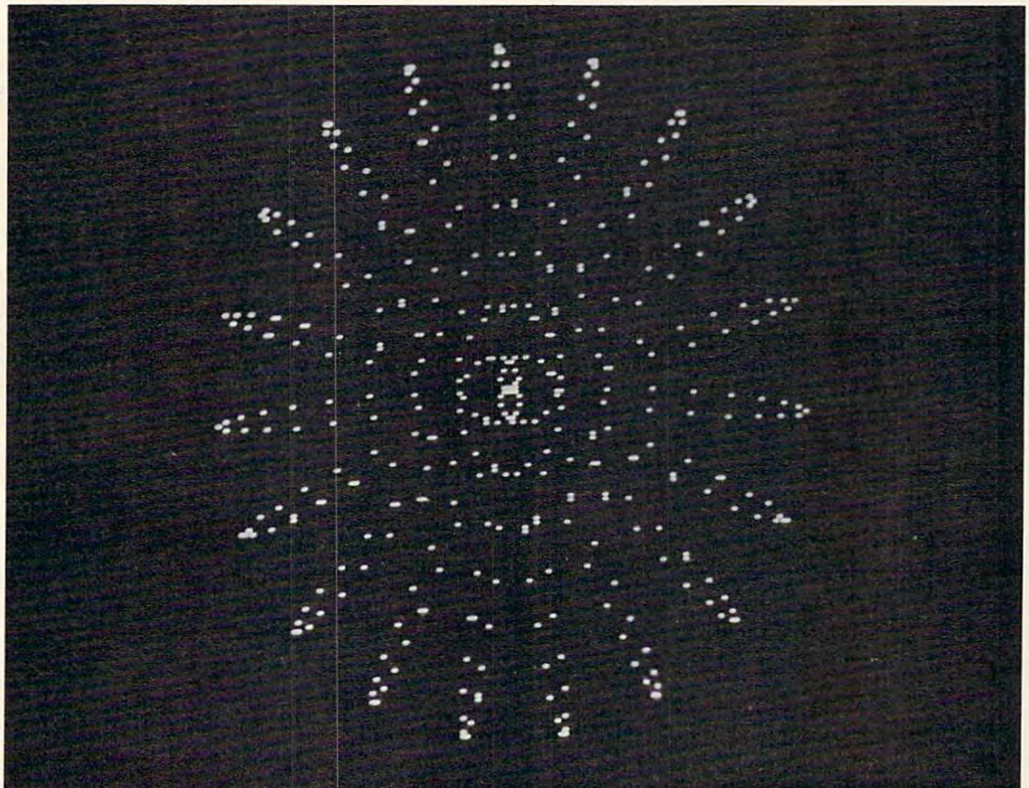
Figure 1. A Four-Leaved Rose.
R=90*SIN(2*THETA)



the angle exceeds the ending angle. The student can watch the points being plotted and see the corresponding R and θ values printed underneath the graph.

A photograph of the screen of the video monitor after the graph R = 85*SIN(19*THETA) was

Figure 2: A Nineteen-Leaved Rose



plotted is shown in Figure 2. Notice that different X and Y scale factors on the screen produce a slight distortion that is not important as far as the present application is concerned.

Of course, it is always possible to add a few bells and whistles. Program 2 represents a few non-essential, but nice, additions to the first program. The coordinate axes are drawn and the X and Y values are rounded to their nearest integer values before plotting. Also, we have made use of the entire screen with the HGR2 instruction on line 30. The scale of the graph was reduced so that we could plot the finished result on our little printer. If you are using a video monitor or a large printer, then you will want to keep the scale as large as possible (replace all the 80's with 90's).

Some of our results are given in the figures that follow. In Figure 3 we show a graph of R=80*SIN(3*THETA) a *three-leaved rose*. Figure 4 is a graph of a *13-leaved rose*, R=80*SIN(13*THETA). The *cardioid* R=40*(1+COS(THETA)) is illustrated in Figure 5. Figure 6 is the famous *Spiral of Archimedes*, R=6*THETA. Figure 7 is similar, but not identical to the *Limacon of Pascal*. We chose R=80*COS(THETA/3) for this figure. Figure 8 illustrates the *Litus* described by the equation R=25*(2+SIN(3*THETA)). Figure 9 has no name, but its equation is R=25*(2+SIN(3*THETA)).

Finding where two polar curves intersect is sometimes difficult. If you have a printer you can simply graph the polar curves, overlay their graphs, and find approximate points of intersection.

Students seem to enjoy working with these programs. They are simple enough so the students can modify the various parameters rather easily, giving them a chance to experiment freely. At the

Figure 3. A Three-Leaved Rose.

R=80*SIN(3*THETA)

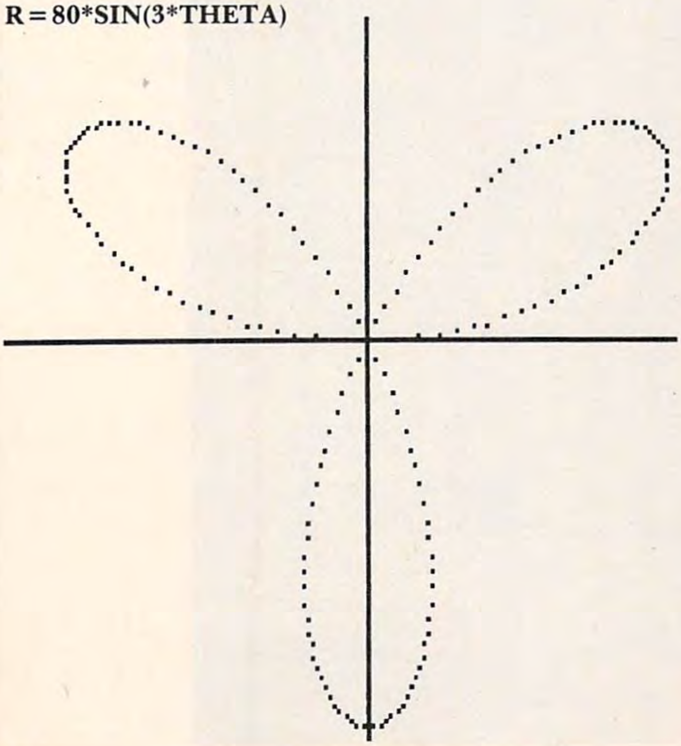


Figure 4. A Graph of R=80*SIN(13*THETA), A 13-Leaved Rose.

R=80*SIN(13*THETA)

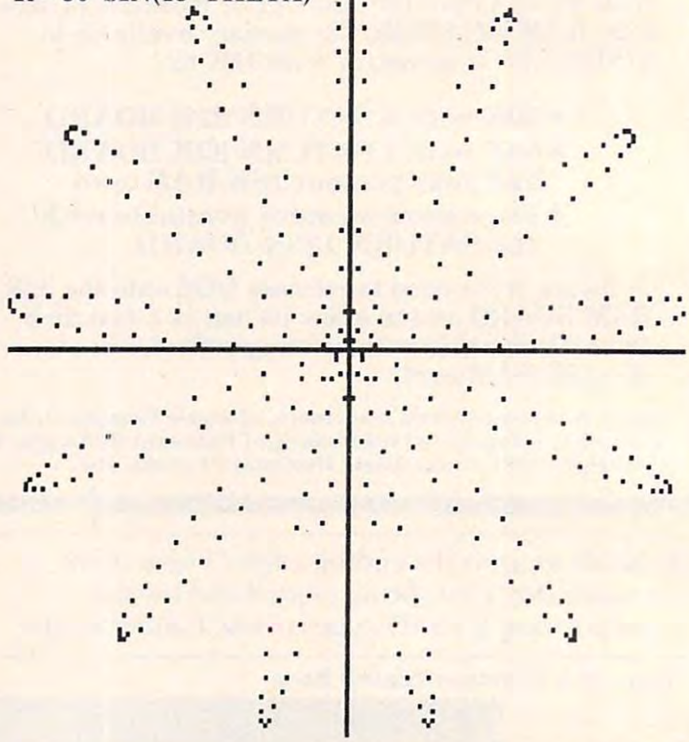
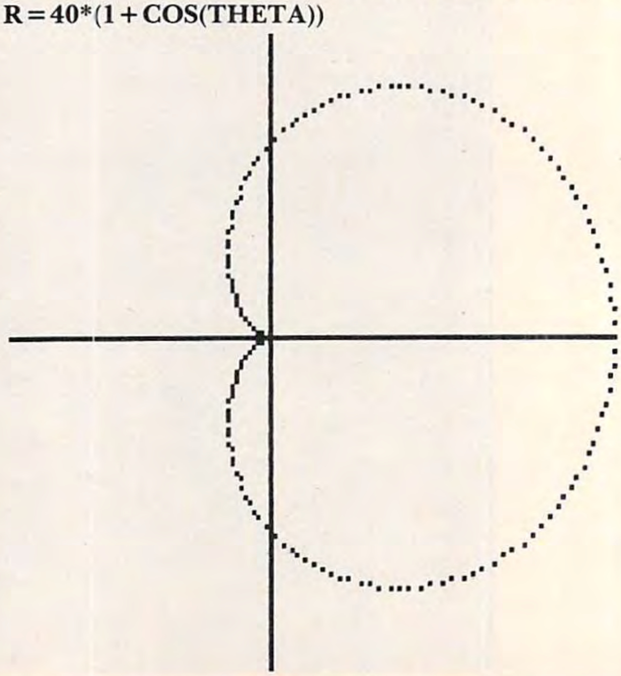


Figure 5. The Cardioid R=40*(1+COS(THETA)).

R=40*(1+COS(THETA))



very least, the programs release them from the drudgery of plotting points by hand.

Figure 6. Spiral of Archimedes with R=6*THETA.

R=6*THETA

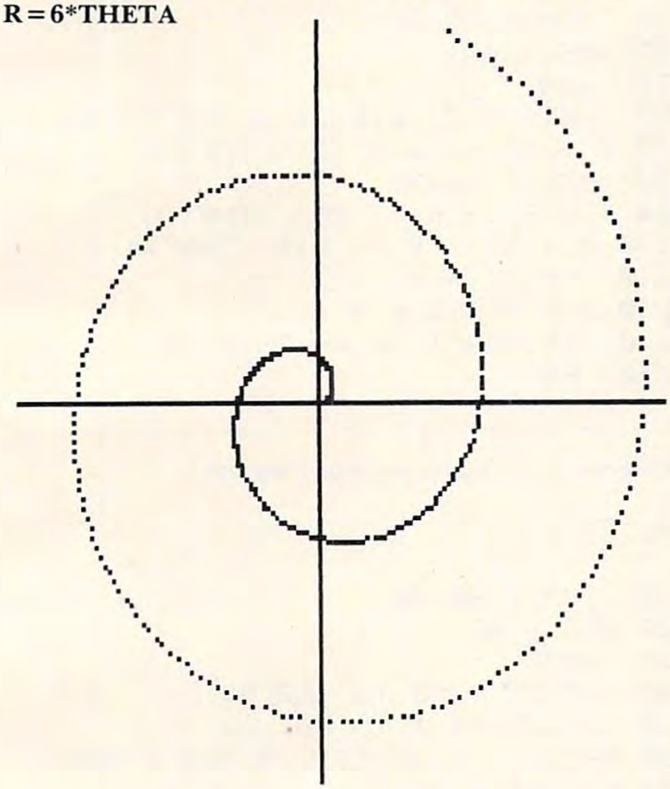


Figure 7. A Graph of R=80*COS(THETA/3).

R=80*COS(THETA/3)

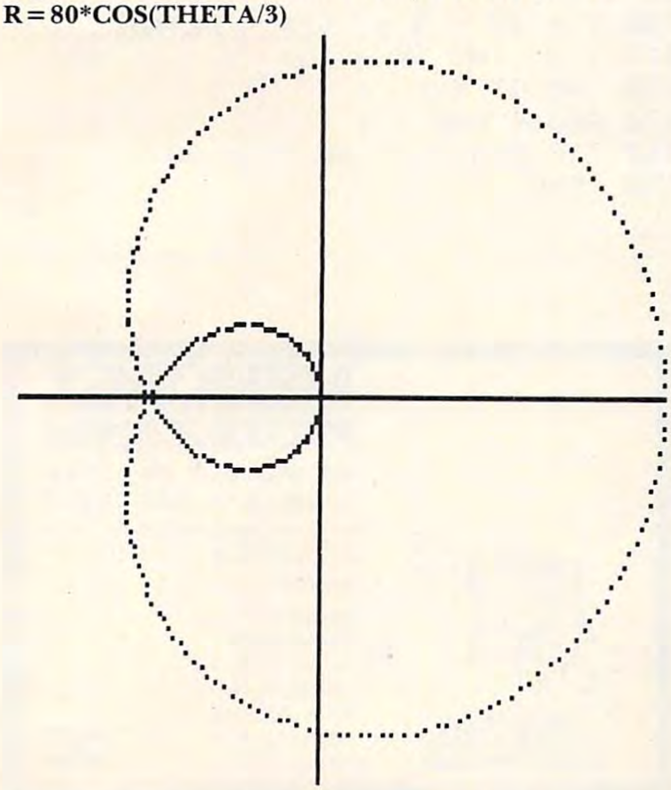


Figure 8. A Graph of R = SQR(3600/THETA).

R = SQR(3600/THETA)

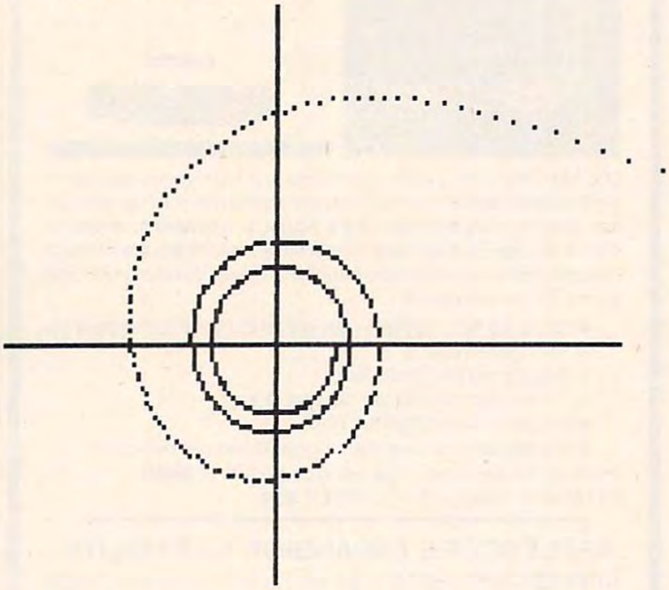
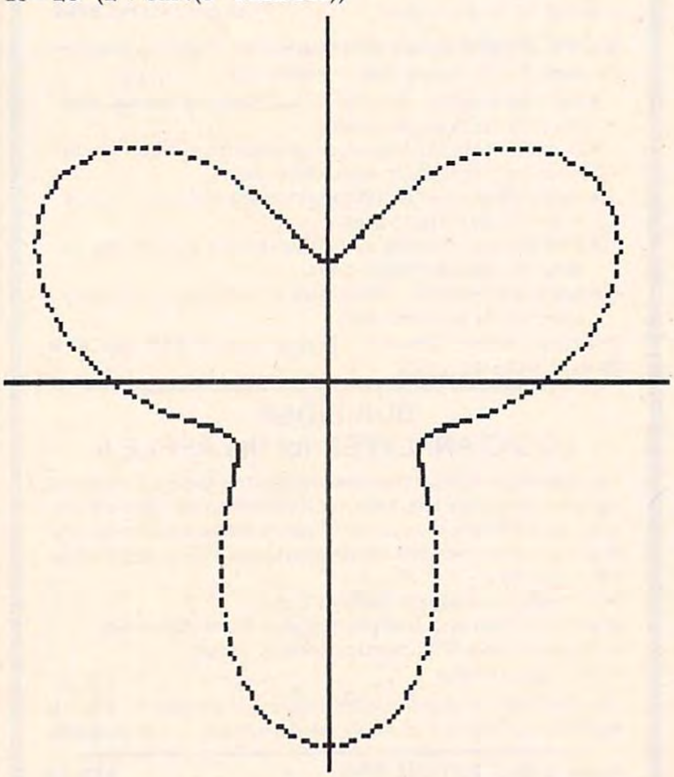


Figure 9.
Untitled Graph with R = 25*(2 + SIN(3*THETA)).

R = 25*(2 + SIN(3*THETA))



Program 1. A Simple Program to Graph Polar Functions

```
←LIST

10  INPUT AA,AB
20 ANG = AA
30  HGR
60 THETA = 3.1415926 * ANG / 180
70 R = 85 *  SIN (2 * THETA)
80  PRINT R,ANG
90 X = 85 + R *  COS (THETA)
100 Y = 85 - R *  SIN (THETA)
120  HPLOT X,Y
130 ANG = ANG + 1
140  IF ANG <  = AB THEN 60
150  END
```

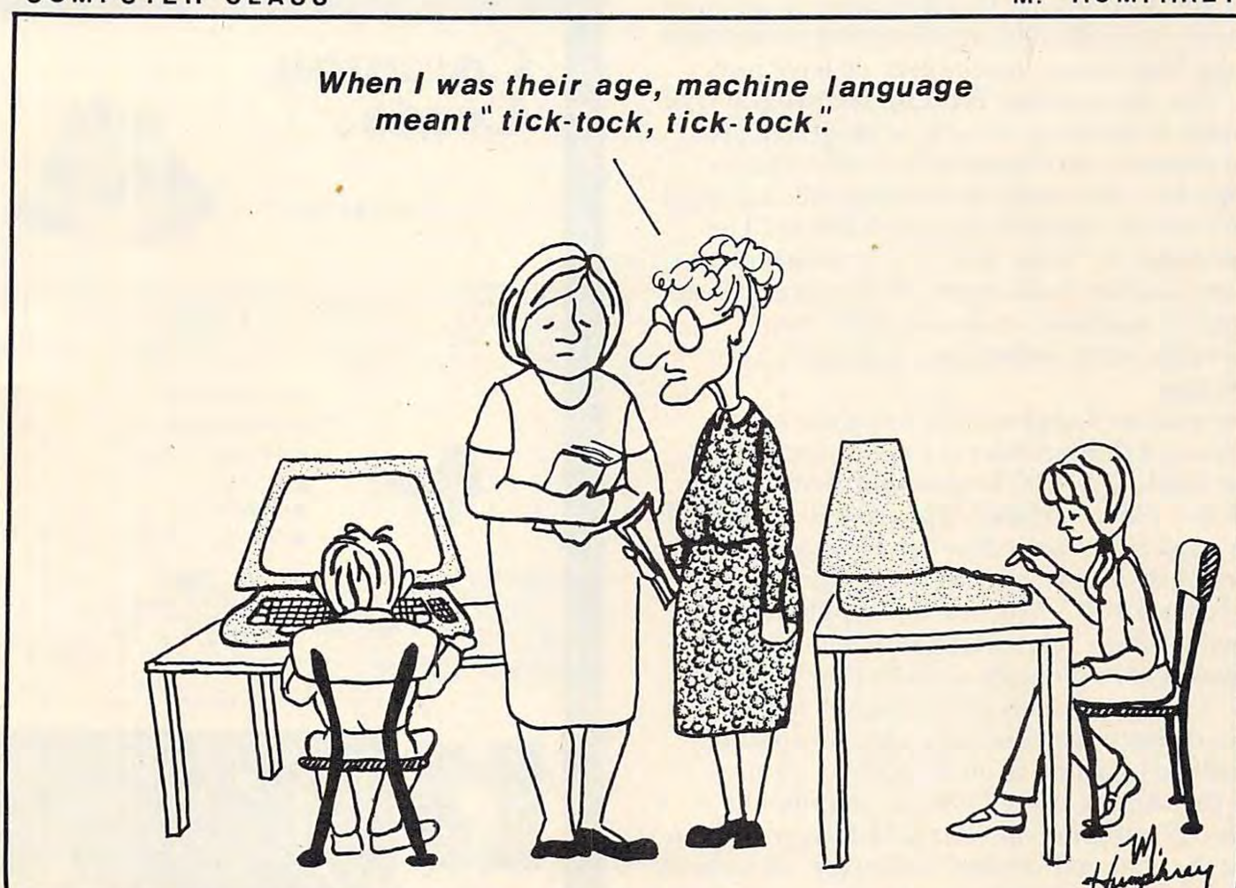
Program 2. An Elaboration of Program 1.

```
←LIST

10  INPUT AA,AB
20 ANG = AA
30  HGR2
40  HPLOT 1,80 TO 160,80
50  HPLOT 80,1 TO 80,160
60 THETA = 3.1415926 * ANG / 180
70 R = 80 *  SIN (3 * THETA)
80 X = 80 + R *  COS (THETA)
90 X =  INT (X + .5)
100 Y = 80 - R *  SIN (THETA)
110 Y =  INT (Y + .5)
120  HPLOT X,Y
130 ANG = ANG + 1
140  IF ANG <  = AB THEN 60
150  END
```

©

COMPUTER CLASS M. HUMPHREY



Disassembling Machine Language Programs Without Leaving BASIC

John R. Vokey and H. Cem Kaner
McMaster University
Hamilton, Canada

One of the nice features of the Apple computer is that it has a built in mini-assembler and disassembler. The mini-assembler is all you need for entry of short machine language programs. In fact, for short programs, this free piece of software has proven more flexible and less error prone than two of the "full blown" assemblers we have purchased. The disassembler is useful for programs of any length. If you have a machine language program in memory, the disassembler will translate the program's code from meaningless hexadecimal numbers into an assembly language listing. The listing includes no labels, just instructions and addresses, but this is still quite informative. It is not too hard, for example, to decode fairly large sections of the code underlying Applesoft from such listings.

The standard approach to using the mini-assembler and disassembler is to jump into the monitor (via CALL -151 from either BASIC) and to work from there. These steps are well described in your *Apple II Reference Manual.* However, it is also possible to access some of these monitor commands from BASIC. The one line Applesoft program below allows you to disassemble machine code anywhere in memory without ever leaving BASIC. This is especially convenient if you are trying to debug a machine language subroutine which will be CALLed from BASIC. You can change the routine using POKEs, examine the changes using this line in your CALLing program, and test the changed version's behavior, all without leaving Applesoft.

The program works by passing the user-specified START location of the code to be disassembled to the monitor program counter (labelled PC in the program). It then calls the monitor LIST subroutine which we label disassemble in the program. This routine disassembles the next 20 lines of machine code, incrementing the monitor program counter locations appropriately, and returns control to BASIC. The BASIC program then compares the value of the monitor program counter to the user specified value FINISH. If there is more to be done before location FINISH is reached, the program waits until you press any key, then continues the listing. Once FINISH is reached, the program ends.

As an example of the use of the program, if you set START to 65118 and set FINISH to 65140, you will disassemble the disassembler.

```
1000 DISASSEMBLE = 65121: PC = 58:
POKE PC, START - INT (START / 256)
* 256: POKE PC + 1,START / 256: FOR
I = 0 TO 1: HOME: CALL DISASSEMBLE:
PRINT: PRINT TAB (13);"<PRESS ANY
KEY>": GET Z$: I = ( PEEK (PC + 1)
* 256 + PEEK (PC)) > FINISH: NEXT I
```

©

Named GOSUB With Variable Passing

Mike Smith
Calgary, Canada

In **COMPUTE!** # 12, I described a machine language program which would allow subroutines to be called by name rather than by number. This article is an extension of that idea. It describes a machine language program which allows parameters to be passed in and out of subroutines.

One of the nicer features of FORTRAN and PASCAL is their ability to pass variables into a subroutine. This feature is very useful when you wish to do the same operation on a large number of variables. Passing parameters into subroutines is convenient since the variable names used outside the subroutine don't have to be the same as used for the calculation within the subroutine. This makes programming and documentation easier. In addition, subroutines of this type can be used as a sort of multi-line function.

A Brief Example Of Parameter Passing

Suppose that you wish to perform a complicated operation upon variables A, B and C and have the answer returned in D. Then you wish to have the same operation performed upon the variables A1, B1 and C1 and have that answer returned in D1.

In FORTRAN that program would look like this:

```
CALL COMPL(A,B,C,D)
     (call subroutine with first variable set)
CALL COMPL(A1,B1,C1,D1)
     (then with the second set)
.........
(Use D and D1 in calculations)
.........
SUBROUTINE COMPL(W,X,Y,Z)
     (use dummy variables with subroutine)
(Complicated calculation using W, X and Y)
.........
Z = .....
RETURN
```

In Applesoft BASIC things are a little more difficult. First, you must call the subroutine by a *number* rather than by a *name*. A second problem is that you can't pass the names of variables into the subroutine. Instead, you must move (reassign) the values into the variable names used in the subroutine. An equivalent Applesoft BASIC program would look something like:

```
10  W=A : X=B : Y=C
     (reassign first set of variables)
20 GOSUB 1000
30 D=Z
40 W=A1 : X=B1 : Y=C1
     (reassign second set)
50 GOSUB 1000
60 D1=Z
70 .........
   (Use D and D1 in calculations)
   .........

1000 (Complicated calculation using W, X and Y)
     .........
1100 Z= .....
1110 RETURN
```

Having to remember the subroutine number is no great problem if you are the person who did the programming, provided you only did the programming a week or so ago, and have not yet forgotten what subroutine number was needed for what. Having to reassign variables, as in statement 40, is no great problem either, provided you don't have a large number of different variables that need to be worked on. But why do something that the computer can make easier to understand and do?

The program described in this article uses the Applesoft BASIC *ampersand* command (&) to allow the naming of subroutines and the easy passing of numerical data. With the machine code routine installed in memory, the Applesoft program above becomes:

```
10  COMPL = 1000
     (establish the subroutines name)
20  & GOSUB COMPL !COMPL(0),A,B,C,D!
     (pass the parameters)
30  & GOSUB COMPL !COMPL(0),A1,B1,C1,D1!
40  .........
    (Use D and D1 in calculations)
    .........

1000  & GET !COMPL(0),W,X,Y,Z!
     (identify the dummy variables)
1010  (Complicated calculation using W, X and Y)
     ........
1100  Z = ....
1110  & RETURN !COMPL(0)!
```

In addition to passing parameters, Applesoft will now support GOTO and GOSUB statements that have names instead of numbers. For example

```
JUMP = 1000 : & GOTO JUMP or COMPL = 1000 :
   & GOSUB COMPL
FIRST = 1000 : DEUX = 2000 : ON X GOSUB FIRST,
   DEUX
```

I decided to develop this parameter passing routine because I am repeatedly asked to translate FORTRAN program with subroutines into Applesoft. Most of those subroutines pass variables. Making sure that I didn't duplicate names and that

I reassigned the right variable, was too much of a hassle. Hence this routine.

Loading The Program

The machine language program as described in this article is too long to put in a normally unused area of memory. The cassette buffer (at $300) will only accept around $CF locations before running into the DOS pointers at $3D0.

The program could be placed high in memory, just below the normal HIMEM. The HIMEM pointers must then be adjusted so that the program is not touched by Applesoft when strings are used. However, this means that people using 48K and 32K Apples, with or without the Program Line Editor at the top of memory, will all need different programs. The modifications are simple, if you know how. Therefore, I have adopted the technique of moving LOMEM up $200 bytes and storing the machine language code in the space created. Then everybody gets the same code.

Before entering the demonstration BASIC program, type:

POKE 104,10 : POKE 2560,0 : NEW

These three instructions adjust LOMEM and the various Applesoft RUN, LOAD or SAVE programs. The pointers can be shifted down to their normal place by typing FP.

After the BASIC program has been run, the machine code can be saved by the command BSAVE VARIABLE.PASS, A$803,L$181. The program will stay active, below your BASIC program, until you power down or do an FP.

To reload the ML program the next time you power up, type BRUN VARIABLE.PASS either from the keyboard or as part of your HELLO program. The LAST line of the HELLO program should be PRINT CHR$(4);"BRUN VARIABLE.PASS".

The first couple of statements of the hex code are the machine language equivalent of POKE 104,10 : POKE 2560,0 : NEW. That means that you only have to adjust the memory the first time you enter in the code. If you forget to adjust the memory before running the demonstration BASIC program, you will receive the message SYNTAX ERROR in 34057, a non-existent line. Simply type NEW : POKE 104,10 : POKE 2560,0 : NEW, reload the program from disk and RUN again. If you didn't adjust LOMEM, then, when the BASIC program stored the machine language program, it did so all over itself, causing a gigantic mess.

There is a sneaky reason for starting the machine language program at $803 (2051) rather than at $800, the start of the empty memory area. Suppose that, for some reason or another, you need to enter FP to recover from your program doing something strange. Typing FP causes 0's to be written at locations $800-$802 to indicate that there is no longer a program in the memory. This misses the ML program since it starts at $803. Thus, a quick CALL 2051 and ABRACADABRA, the pointers shift and the program is back in business.

The details of the demonstration and machine language programs are given after the description of the new SYNTAX of the instructions and limitations of the new commands.

Syntax For The New Commands

& GOSUB NAME !NAME(0),A,B,.....!

The name of the subroutine must be predefined before the subroutine is called (e.g. NAME = 1000).

The first parameter after the exclamation mark *must* be an array; otherwise, a BAD SUBSCRIPT ERROR occurs. It is suggested that the name of this array be the same as the name of the subroutine; for ease of remembering rather than necessity. If more than ten parameters are to be passed by the routine, the array must be DIMensioned to the number of parameters. No check is performed to see if the array is large enough for all the parameters used.

The other parameters must be numerical, either real variables (A, B etc.) or elements of a real array (A(1), B(1) etc.). The arrays don't have to be predimensioned unless their length is greater than ten. Errors will occur on attempting to pass a string (TYPE MISMATCH) or an integer (SYNTAX). It should be noted that it is the *value* of the array element that is moved and *not* the array itself. This means that you can't pass over the whole array by passing over the first element of an array. (c.f. In FORTRAN, it is the address which is passed and not the value of the array element. So, the whole array can be accessed from FORTRAN subroutine if you know the first address. In Applesoft, memory is continually being repositioned. The address of any variable is therefore continually changing, making any address stored very quickly invalid.)

The parameters do not need to have been defined before calling the subroutine. The machine language program makes use of Applesoft routines which automatically allocate space in the memory for new arrays and variables.

& GET !NAME(0),P,Q,.....!

This should be the first statement of the subroutine. The subroutine can't be recursive (it can't call itself).

This command does not extend an existing Applesoft command as did the & GOSUB, & RETURN and & GOTO commands. Therefore I had to use a different command. I decided to use GET, Since to me, this new command goes and *gets* the parameter values. If you would prefer a different command, such as LOAD, then the modification to allow this is simple. To have a different command, POKE its token into location 2600 ($828) before BSAVEing the program. For example, POKE 2600 ,167 will change this command to be & RECALL !......! rather than & GET !......!. (See page 121 of the *Applesoft Manual* for a list of the tokens).

The first parameter after the exclamation mark must be the same array used in the & GOSUB statement, otherwise unexpected values will be put into the parameters (P etc).

The other parameters must be real, otherwise a TYPE MISMATCH or SYNTAX ERROR will result. Either real variables (P) or elements of real arrays (P(1)) may be used. Again, the parameters don't have to be predefined before the subroutine call, unless they are arrays of length greater than ten. If the arrays need to be DIMensioned remember to do it *outside* the subroutine. Otherwise a REDIMENSIONED ARRAY ERROR will result on the second subroutine call.

The number of parameters in the & GET statement should be the same as the number of parameters in the & GOSUB statement. If this condition is not met, strange values could arrive in the parameters of the & GET statement.

& RETURN !NAME(0)!

The array used in the & RETURN statement should be the same array as used in the & GOSUB and & GET statements. As this array is used to temporarily store text pointers to the & GOSUB and & GET statements, strange results could result if the wrong array is used. However, it is probable that, instead of funny results, a SYNTAX ERROR will occur. The likelihood of the wrong array pointing to valid names in separate locations in memory is very small.

If the number of parameters in the & GET statement is not the same as the number of parameters in the & GOSUB statement, unpredictable values will be put into the parameters.

& GOTO NAME and & GOSUB NAME

The name of the subroutine must be established before it is called. If these commands are used, a normal RETURN is all that is needed. If & GET and & RETURN are used, a SYNTAX ERROR will occur.

& ON X GOSUB FNAME, SNAME and & ON X GOTO FNAME, SNAME

These ON X.... commands are supported, provided that no parameters are passed. That means that & ON X GOSUB FNAME, SNAME is permitted but & ON X GOSUB FNAME !FNAME(0),A,B,C,D!, SNAME !SNAME(0),A,B,C,D! is not. I felt that passing parameters in ON X... statements made the statements very unwieldy. The original idea behind introducing these new commands was to make the programs more readable rather than less. Multiline IF...THEN commands would do the same job, in a more readable fashion. For those people interested in implementing the unweildy ON... version, I have included the additional code needed (lines 176-189).

Warning

Warning on renumbering and crunching programs: Renumbering programs will not change the values of variables. Therefore, they will not change the pointers to the subroutines called by these new commands. This must be done by hand after the renumbering is complete. Utilities that crunch programs will not recognize the fact that the subroutines are being called and therefore will remove them as dead code. To overcome this removal problem, a dummy line that calls all subroutines, must be added to the program. After crunching, delete the dummy line. For example:

```
10 NAME=1000 : FIRST=2000
   (define the subroutines)
20 IF X=0 THEN GOSUB 1000 : GOSUB 2000 :
   GOTO 20 (dummy line to be removed after
   crunching)
```

Note that the dummy line is an IF..THEN statement that loops to itself. This means that a CRUNCHER, such as the one in DAKIN 5 PROGRAMMING AIDS 3.3, will leave that line alone, making it easy to remove.

BASIC Program Description

Line 180 – Establishes the machine language program.

Line 200 – Establishes the name of the subroutines to be called.

Line 220 – Demonstrates the command & GOSUB without passing any variables.

Line 250 – A loop is used to show that the stack is not corrupted by using these new commands. An OUT OF MEMORY ERROR will occur for 25 GOSUB calls without a proper return.

Line 260-280 – Establish random numbers for use in the variables.

Lines 290-320 – Demonstrates the & GOSUB command using both simple variables and arrays elements. The example subroutine adds together the first two numbers passed to it. The result is passed back in the third parameter.

Line 360 – Demonstrates that the subroutine call operated and that parameters were passed both ways.

Line 370 – Delay loop.

Line 1000 – Subroutine called without passing variables.

Line 2000 – New subroutine showing that variables were passed and used within the subroutine.

Line 5000-5070 – Machine language loading subroutine. It first checks that the DATA statements have been typed in correctly. Each DATA statement is the value of 16 locations plus the sum of the previous 16 locations used as a simple checksum. A typo error is indicated if the checksum is not the sum of the previous 16 locations.

Line 5080-5120 – Checks that POKEs have been performed.

Line 5130-5140 – POKEs the routine into memory.

Line 5150 – This establishes the AMPERSAND vector (&) pointers. This call is not necessary if the machine code is BRUN, but is necessary if the subroutine is BLOADed. Note that the CALL from BASIC is not the start of the ML program. If we did CALL the start of the program, an automatic NEW would occur, wiping out the demonstration program.

Machine Code Description

Briefly, the machine language program works as follows:

& GOSUB NAME!NAME(0),A,B...! The text pointers to the variable A are stored in the first two bytes of NAME(0). Then the value of A is moved into NAME(1), B into NAME(2) and so on.

& GET !NAME(0),W,X,...! The text pointers to the variable W are stored in the second two bytes of NAME(0). The value of NAME(1) is moved into W, NAME(2) into X and so on.

& RETURN !NAME(0)! The text pointer to W are recovered. The current values of W, X .. are moved into NAME(1), NAME(2) etc. Then the text pointer to A is recovered. The values in NAME(1), NAME(2) ... are moved into A, B....

The method of implementing the other commands is described in **COMPUTE!** #12.

Lines 15-31 – Zero page usage.

Lines 33-43 – Definition of tokens.

Lines 45-61 – Pointers to Applesoft routines. Internal Applesoft routines are used to cut down the amount of code required.

ADJMEM and AMPER. **Lines 65-77** – Do the machine language equivalent of POKE 104,0 : POKE 2560,0 : NEW. Then set the AMPERSAND vector.

ENTRY. **Lines 80-92** – Check on which of the new commands is required.

GOTO. **Lines 94-99** – Front end of the normal Applesoft GOTO routine moved and modified to allow variables and numbers to be used in the GOTO statement.

GOSUB. **Lines 101-134** – Handling of the & GOSUB command.

Line 101 – Front end of the normal Applesoft GOSUB routine moved and modified to allow variables and numbers in subroutine calls.

Line 121 – Is the first parameter an array? This array is used for the storage of the text pointers and the parameters. The stack would get too full if it were used.

Line 130: – Store text pointers.

Line 132 – Move the other parameters into the array for storage.

GET. **Lines 136-140** – Handling of the & GET command.

Line 136 – Locate the storage array.

Line 137 – Store the text pointers.

Line 139 – Move values stored in the array into the new parameters.

ARRay-GET. **Lines 142-149** – Gets and stores the location of the storage array after checking the leading exclamation mark.

ON. **Lines 151-189** – Handling of the ON X... command.

Line 152 – Get the value of X.

Line 154 – Determine if ON..GOTO or ON.. GOSUB.

Line 163 – Decrement X until find the subroutine requested.

Line 167 – Step over the values not being used.

Line 172 – Return to BASIC if subroutine not found.

Line 176-189 – Adding these instructions will allow the passing of parameters in ON X... commands.

RETURN. **Lines 191-213** – Handling of the & RETURN command.

Line 191 – Locate the storage array.

Line 198 – Store the current text pointers.

Line 199 – Recover the text pointers from the & GET statement.

Line 200 – Move the current values of the parameters in the & GET statement into storage.

Line 201 – Reset the storage array pointers.

Line 205 – Recover the text pointers from the & GOSUB statement.

Line 206 – Move the values in the storage array into the parameters used in the & GOSUB statement.

Line 207 – Recover the current text pointers and perform a normal RETURN.

CHecK-ARRay. **Lines 215-228** – Checks and adjusts the pointers to the storage array if new variables have been introduced during the commands & GOSUB and & GET.

Modifications to the next two subroutines, PARSTO and STOPAR will allow the passing of INTEGER parameters.

PARameters-to-STOrage. **Lines 230-243** – Moves the current values of the parameters in the & GOSUB and & GET commands into the storage array. Checks for integers and strings.

STOrage-to-PARameters. **Lines 244-257** – Moves the values in the storage array into the parameters in the & GOSUB and & GET commands.

STOre-TeXT-pointers. **Lines 259-264** – Stores the current text pointers into the zeroth element (NAME(0)) of the storage array. The Y register is preset.

GET-TeXT-pointers. **Lines 266-271** – Recovers the text pointers stored in the zeroth element of the storage array according to the value set in the Y register.

ADJust-PoinTers. **Lines 273-276** – Adjust the pointers to the storage array if they have shifted because a new variable has been made. Note that the pointers don't have to be adjusted if a new array has been made. All new arrays will be placed above the storage array in memory as the storage array is defined first.

COMmand-END. **Lines 278-283** – Looks for the final exclamation mark (!) of the command or other parameter. Pops the last subroutine address off of the stack allowing a quick return to BASIC if at the command's end.

References

"Applesoft Internal Entry Points" by Applesoft Computer Inc. in Apple Orchard March/April 1980, p. 12.

"Some Routines in Applesoft Basic" by J. Butterfield in COMPUTE!, September/October 1980, p. 68.

"Resolving Applesoft and Hires Graphics Memory Conflicts" by J. Schmoyer in COMPUTE!, April 1981, p. 76.

"Using Named GOSUB and GOTO Statements in Applesoft BASIC" by M. Smith in COMPUTE!, May 1981, p. 64.

```
100 ******************************
110 REM * MIKE SMITH                *
120 REM * 304, 86TH AVENUE SE       *
130 REM * CALGARY, ALBERTA          *
140 REM * CANADA  T2H 1N7           *
150 ******************************
160 REM
170 REM SET UP THE MACHINE CODE
180 GOSUB 5000
190 REM   SET UP THE SUBROUTINE NAM
    ES
200 DEMO =1000:ADDIT = 2000
210 REM  DEMONSTRATE NAMED GOSUB AN
    D GOTO
220 &  GOSUB DEMO:JUMP = 240: &  GO
    TO JUMP
230 REM DEMONSTRATE STACK OKAY
240 PRINT "HERE BY NAMED GOTO": PRI
    NT
250 FOR J = 1 TO 25
260 REM MAKE UP NUMBERS
270 K = INT (10 * RND (1)): = INT (
    10 *  RND (1))
280 P = INT ( 10 * RND (1)):Q(1) = ~
    INT (10 * RND (1))
290 REM
300 &  GOSUB ADDIT!ADDIT(0),K,L,M!
310 REM   DEMONSTRATE PASSING OF AR
    RAY ELEMENT
320 &  GOSUB ADDIT!ADDIT(0),P,Q(1),
    R!
330 REM
340 REM PRINT AND SHOW THAT HAVE US
    ED SUBROUTINE
350 REM
360 PRINT K;" + ";L;" = ";M: PRINT ~
    P;" + ";Q(1);" = ";R: PRIN
    T
370 FOR Z = 1 TO 500: NEXT Z
380 NEXT J: STOP
970 REM
980 REM  DEMONSTRATION SUBROUTINE
990 REM
1000 PRINT : PRINT "HERE BY THE GOSU
    B CALLED DEMO"
1010 PRINT : RETURN
1960 REM
1970 REM  SUBROUTINE ADDIT
1980 REM
1990 REM  DEMONSTRATE PASSING BACK O
    F ARRAY ELEMENT
2000 &  GET !ADDIT(0),T,U,V(4)!
2010 V(4) = T + U
2020 &  RETURN !ADDIT(0)!
4970 REM
4980 REM  MACHINE CODE ESTABLISHED
4990 REM
5000 BOT = 8 * 256 + 3:HIGH = 9 * 25
    6 + 10 * 16 + 2
5010 REM FLAG FOR CHECKSUM
5020 OK = 1:LINE = 6000
5030 FOR J = BOT TO HIGH STEP 16
5040 CHECK = 0: FOR K = J TO J + 15:
     READ IT:CHECK = CHECK + I
    T: NEXT K
5050 READ NUM: IF NUM < > CHECK THEN
     PRINT "TYPO IN LINE "LINE
    ::OK = 0
5060 LINE = LINE + 10: NEXT J
5070 IF OK = 0 THEN STOP
5080 PRINT  : INPUT "DID YOU REMEMBE
    R THE POKES? ";A$
5090 IF LEFT$ (A$,1) = "Y" THEN 5130

5100 PRINT : PRINT "SAVE THIS PROGRA
    M AND THEN"
5110 PRINT : INVERSE : PRINT "NEW:PO
    KE104,10:POKE2560,0:NEW": ~
    NORMAL : PRINT
5120 PRINT "THEN RELOAD AND RUN.": S
    TOP
5130 RESTORE : FOR J = BOT TO HIGH S
```

```
    TEP 16
5140 FOR K = J TO J + 15: READ IT: P
    OKE (K),IT: NEXT K: READ I
    T: NEXT J
5150 PRINT : PRINT "BLOAD OKAY": CAL
    L BOT + 12: RETURN
5970 REM
5980 REM  MACHINE CODE DATA
5990 REM
6000 DATA 169,10,133,104,169,0,10,32
    ,75,214,169,76,141,245,168
    8
6010 DATA 3,169,31,141,246,3,169,8,1
    41,247,3,96,201,171,240,25
    ,1894
6020 DATA 201,176,240,36,201,190,240
    ,106,201,180,208,3,76,181,
    8,201,2448
6030 DATA 177,208,3,76,230,8,76,201,
    222,32,66,8,76,65,217,32,1
    697
6040 DATA 177,0,32,123,221,76,82,231
    ,169,3,32,214,211,165,185,
    72,1993
6050 DATA 165,184,72,165,118,72,165,
    117,72,169,176,72,32,66,8,
    32,1685
6060 DATA 183,0,201,0,240,38,201,58,
    240,34,201,44,240,30,32,16
    6,1908
6070 DATA 8,196,108,48,6,208,7,197,1
    07,16,3,76,150,225,32,249,
    1636
6080 DATA 234,32,106,221,160,0,32,11
    9,9,32,52,9,32,63,8,76,118
    5
6090 DATA 210,215,32,163,8,160,2,32,
    119,9,32,87,9,76,149,217,1
    520
6100 DATA 32,177,0,201,33,208,143,32
    ,177,0,32,227,223,133,0,13
    2,1750
6110 DATA 1,96,32,177,0,32,248,230,7
    2,201,176,240,13,201,171,2
    40,2130
6120 DATA 9,201,175,208,224,104,32,1
    77,0,72,198,161,208,4,104,
    76,1953
6130 DATA 31,8,32,177,0,32,227,223,3
    2,183,0,201,44,240,235,104
    ,1769
6140 DATA 104,104,96,32,163,8,141,16
    2,9,140,163,9,165,184,72,1
    65,1717
6150 DATA 185,72,160,2,32,129,9,32,5
    2,9,173,162,9,133,0,173,13
    32
6160 DATA 163,9,133,1,160,0,32,129,9
    ,32,87,9,104,133,185,104,1
    290
6170 DATA 133,184,32,177,0,76,107,21
    7,165,107,197,2,208,1,96,1
    33,1835
6180 DATA 2,169,7,208,2,169,5,24,101
    ,0,133,0,2,144,230,1,1197
6190 DATA 96,32,139,9,32,123,221,32,
    106,221,165,18,240,3,76,19
    8,1711
6200 DATA 8,32,27,9,166,0,164,1,32,4
    3,235,32,149,9,32,177,1116
6210 DATA 0,76,55,9,32,139,9,32,227,
    223,32,27,9,165,0,164,1199
6220 DATA 1,32,249,234,166,131,164,1
    32,32,43,235,32,149,9,32,1
    77,1818
6230 DATA 0,76,90,9,165,184,145,0,20
    0,165,185,145,0,96,177,0,1
    637
6240 DATA 133,184,200,177,0,133,185,
    96,165,107,133,2,32,40,9,7
    6,1672
6250 DATA 190,222,32,40,9,32,183,0,2
    01,33,208,2,104,104,96,0,1
    456
6260 DATA 104,104,96,0,0,0,0,0,0,0,0
    ,0,0,0,0,0,304
```

©

THE ATARI® GAZETTE

INSIGHT: Atari

Bill Wilkinson
Optimized Systems Software
Cupertino, CA

This month marks the end of my series on Atari I/O. That certainly doesn't mean that we won't continue to discuss assembly language I/O of related topics; it simply means that I feel I have finished my formal presentation of the material. Again, I strongly urge you to purchase the *Atari Technical User's Notes* (available from Customer Service, 1340 Bordeaux Ave., Sunnyvale, CA 94086, for $30, including shipping). There is a lot of detail in those "notes," including much that I have glossed over. I hope that my presentation, though, has served as a usable introduction to the subject.

Also this month, I give you a method for creating relocatable assembly language programs (and a method to then load them). We use the loader to implement our "M:" driver from last month, completely via BASIC (thus making it usable for those of you not yet into assembly language...and it *is* usable).

Finally, we continue our discussion of how BASIC works. *De Re Atari*, and the serialized version thereof which appears in this month's *BYTE*, does a good job of discussing the *how* of BASIC's syntaxer; we will delve into the *why*.

Atari I/O, Part 4: GRAPHICS

Errata! Before we get started on this month's topic, I must report an error I made in **COMPUTE!** #18. On page 100, in Table 1, under the "Note" pertaining to ICBLL/ICBLH, I stated that the length is decremented by one for each byte transferred. Actually, Atari's OS is smarter than that: upon return from GET/PUT RECORD (text or binary) ICBLL/ICBLH contain a count of the number of bytes successfully transferred. This result is eminently usable (e.g., in copying records or even whole files), and perhaps we will have a program here soon that demonstrates its use.

On with the new: this whole series started as a result of a comment that I read which said something like "Atari graphics from assembly language are hard to do – you have to know about display lists, vertical blank interrupts, etc." Knowing how BASIC does graphics for its users I said, "Nonsense! It's easy! Someone should show *how* easy!" And Richard Mansfield, of **COMPUTE!**, said, "Gee, I wonder who we could get..." Ahem.

If what you are trying to do is write an improved version of Eastern Front or Pacman or some other such pioneering project, then you need to know everything ever published and then some. *But,* if what you want is simply a way to transfer what you have learned or written using BASIC into a reasonably simple set of assembly language routines, read on.

Remember, BASIC does *all* its graphics and I/O via Atari's OS. BASIC knows nothing of graphics modes, display lists, character sets, color registers, etc. (True, BASIC A+ does its own thing with Player/Missile Graphics, but that's only because Atari's OS doesn't know about PMG.) So, anything done with standard BASIC statements can be duplicated *easily* in assembly language. To demonstrate the truth of this, Figure 1 contains a list of the seven BASIC graphics statements together with a note on how each is accomplished.

Accompanying this article is a listing of my proposal for a set of standard routines to be used by assembly language programmers when interfacing to OS graphics. These routines duplicate, as far as practicable, the statements used to do BASIC graphics. The listing clearly calls out ENTRY and EXIT parameters for each routine (i.e., register usage), so study it carefully.

As a very simple example of the routines' usage, I offer a program fragment that is written in both BASIC and assembly language:

```
GRAPHICS 3          LDA #3
                    JSR GRAPHICS
COLOR 3             LDA #3
                    JSR COLOR
PLOT 10,10          LDX #10
                    LDA #0
                    LDY #10
                    JSR PLOT
DRAWTO 25,15        LDX #25
                    LDA #0
                    LDY #15
                    JSR DRAWTO
SETCOLOR 2,0,14     LDX #2
                    LDA #0
                    LDY #14
                    JSR SETCOLOR
```

Before leaving this topic, some notes on the routines might be helpful: since the A-register will be zero upon entry to PLOT, DRAWTO, LOCATE, and POSITION for all graphics modes except GRAPHICS 8 (or 24), placing a LDA #0 in the beginning of POSITION would save code for anyone not using mode 8. Remember, Atari's "S:" driver can accomodate GRAPHICS 0 through 11 and 17 through 24. Adding 32 ($20) to any graphics mode (at the time of the call to GRAPHICS) will suppress the erasure of the screen. (I haven't figured out a use for this yet, but it's nice to know it's there.)

Obviously, one could save time (and sometimes space) by performing COLOR and SETCOLOR and POSITION via simple stores (e.g., STA), but there is a certain structuring and elegance that goes with the use of the routines. The graphics routines listed herein were assembled in the $600 page of memory, a much overworked location. I would hope that you would take the time to type them in to your assembler/editor and include them directly in future programs (EASMD users may .INCLUDE them indirectly). I really would appreciate hearing of your successes (or failures, if any) using these routines.

So far, no assembler available for the Atari produces relocatable, linkable object files (and, from what I have heard, neither will Atari's Macro Assembler). When we produced BASIC A+ and EASMD, we wanted them to move themselves to the top of memory, so we re-invented a scheme I have seen in several incarnations before: Assemble the program twice, setting the origin for any portion(s) to be relocated one page (256 bytes) higher for the second assembly, producing two object files. Write a program that compares the two objects and notes all locations that differ by one (differing by any other amount is an error). Produce a table (or bit map, or ...) of all these differences. At relocatable load time, read in the first object file (to where it is to be relocated) and use the table to change all the bytes which need to be relocated.

The system is a kludge, but a very effective one. It has a few limitations: you still don't have linkable object files, you must relocate in full page increments (i.e., multiples of 256 bytes), and you have to have some place safe to put the relocating loader. Are you willing to live with those limits? Then try this.

I present here three BASIC programs together with instructions for their use. The first program, MAKEREL (Program 1), seems to be to be perfectly adequate as is, written in BASIC. It's a little slow, but one only uses it when ready to create a new relocatable object file. The other two programs, LOADREL.A and LOADREL.B (Programs 2 and 3), could be advantageously rewritten in assembly language. They are presented here in BASIC because (1) this method fulfills the requirement for a "safe place" for the loader and (2) by presenting them in BASIC they can be used by those not yet ready to tackle assembly language and (3) it was easier for me.

The instructions below presume the use of the Atari Assembler/Editor or the OSS EASMD, but they can be easily adapted to most systems that produce Atari DOS-compatible object files.

How To Use The Relocator Programs

1) Write, assemble, and debug your code using some fixed address(es).

2) Ensure that your code is all in one piece (i.e., there is only one *=, at the beginning of the code segment).

3) Origin your code on an even page boundary (i.e., use ***=$hh00**, there 'hh' specifies any page from 02 through FE). Assemble the code into an object file on disk named "OBJECT1" (use **ASM ,,#D:OBJECT1**).

4) Change your origin to one page higher in memory (***=$nn00**, where 'nn' = 'hh' + 1). Assemble the code to "OBJECT2" (**ASM ,,#D:OBJECT2**).

5) Run the MAKEREL program. It will produce the file "DATA.REL".

6) Adjust the value of the variable NUMBEROFPAGES in both LOADREL.A and LOADREL.B (Programs 2 and 3) to reflect the number of 256-byte pages needed by your routine. **SAVE** the adjusted versions.

7) Anytime you want to load your routine, simply use **RUN "D:LOADREL.A"**.

Notes

A. Generally, it's a good idea to have your routine start execution at the origin (*=) point. Then you can invoke it from BASIC via **USR(PEEK(128) + 256 *(PEEK(129) - NUMBEROFPAGES))**

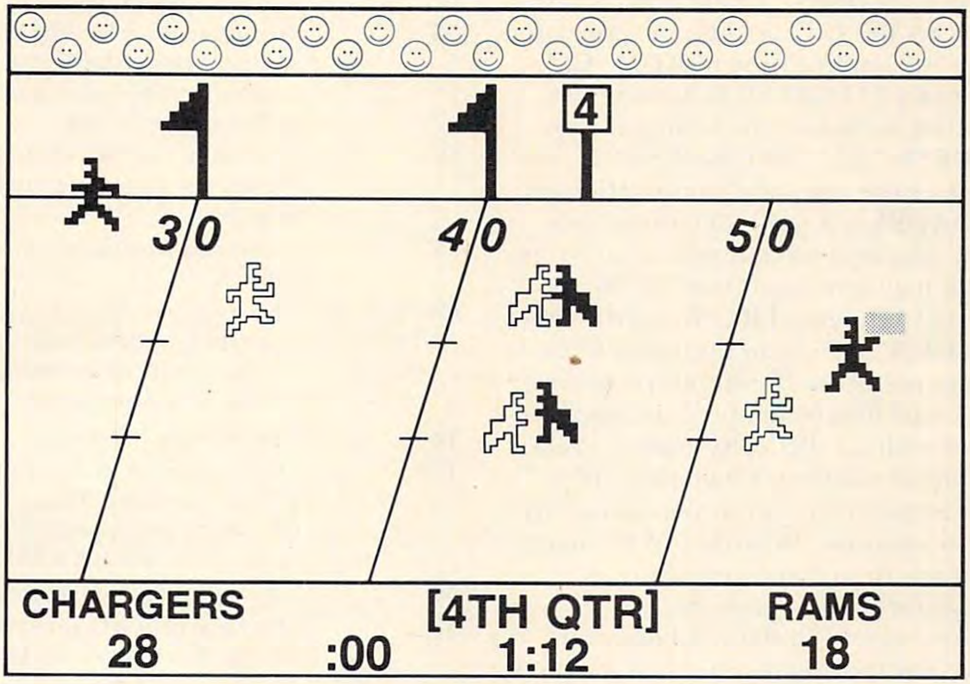
B. If you **RUN "D:LOADREL.A"** again without hitting RESET, it will load another copy above the first. Not too neat, *but* the advantages of being able to thus load several different modules should be obvious!

C. LOADREL.B performs an **ENTER "D: DATA.REL"**. Rather than waiting for the **ENTER** each time, you may **SAVE** the resultant program (after taking out the **ENTER** line) for a slightly faster load of a specific module.

Finally, we offer Program 4 which may be added to LOADREL.B to produce a relocatable load of last month's "M:" driver. (Again, be sure to delete the **ENTER** line from LOADREL.B.)

For once, I haven't forgotten you cassette

ARCADE PRO FOOTBALL™



An Armchair Quarterback's Dream Come True!

It's 4th down and 14 on your opponent's 44 yard line. You're out of time outs with just over a minute left in the game. You call it...sweep right, screen pass left or go for the bomb!

Arcade Pro Football™ puts you in the middle of 60 minutes of exciting gridiron play. You control the action *and* call all the shots. It's the most realistic computer sports game available and it's designed exclusively for the Atari 400/800™ personal computers by Arcade PLUS.

With features like these, Arcade Pro Football™ is in a league of its own!

- Play offense *and* defense head-to-head against the computer or against another player!
- Control two full-color, animated teams in 3-D perspective on a 100 yard, scrolling field!
- 4 game variations, including player handicapping!
- Over 25 offensive and defensive play possibilities...passing, *and* catching, running and kicking!
- All the fun and excitement of real Pro Football...penalties, interceptions, fumbles, bad snaps, 30 second clock—even 4-channel sound and a crowd to cheer you on!

Arcade Pro Football™ is available on cassette or disk for Atari® 400/800™ computers with 16K minimum memory from your local Atari computer dealer. Or send $29.95 (cassette)/$34.95 (disk) plus $2.50 postage and handling (California residents please add 6% sales tax).

arcade PLUS

5276 Hollister Avenue Suite 208 Santa Barbara, CA 93111 (805) 683-2305

users. If you enter LOADREL.A (carefully, please!) and **CSAVE** it (or **SAVE"C:"**) on a blank tape you need only change the last line to read **RUN "C:"**. Then **NEW** and enter LOADREL.B, leaving out the **ENTER** line, but including the listing of Program 4. Use **SAVE"C:"** (do NOT use CSAVE...it won't work!) to place the resultant combination on the tape after LOADREL.A (and, of course, you could then follow on the same tape with a program of your own). You may now enjoy the "M:" driver via this tape by **CLOAD**ing and **RUN**ning the first program (or use **RUN"C:"** if you used **SAVE"C:"**, my own preference for all but the largest programs).

MAKEREL could also be adapted to cassette usage, though not without difficulty and/or a relatively large amount of memory. Obviously, these programs can be improved upon tremendously by simply adding, for example, flexibility of file name. But my intention was to present something as simple and straightforward as possible, in the hopes that everyone would find it readable and useful. Obviously, my techniques could be adapted to other machines (does the PET have a relocating assembler?), so adapt away (and be sure to send **COMPUTE!** the results to share with the rest of us). On to lighter subjects.

Inside Basic, Part 2: The Why Of Syntaxing

Last month I presented a program to print out the keywords of BASIC. If you took the time to enter and run that program, you saw some strange things in the printout of the operators. But there was a method to our madness, as you will see.

Let us examine the tokenized (internal) form of the following line:

```
1025 PRINT "HI THERE", THIS * ( 3 + IS( FUN ) )
  : STOP
```

Assuming that we had just previously **NEW**ed, the tokenized form of that line is as follows (all numbers in decimal):

```
01 04 36 33 32 15 08 72 73 32 84 72 69 18 128
36 43 14 64 03 00 00 00 00 37 129 56 130 44
44 20 36 38 22
```

Now that isn't too terribly useful or readable, so let's examine the tokens one at a time:

01 04	This is the line number (4*256 + 1 = 1025) in standard 6502 form.
36	This is the line length, including the line number and this byte.
33	Statement length of the first statement. Actually, this is the displacement to the beginning of the next statement (from the beginning of the line).
32	The token for PRINT. Check the output of the keyword printing program from last month.
15	A special token that says a string constant follows.
08 72 73 32 74 72 69 82 69	The string constant consists of a byte that gives the length of the string followed by the characters of the string. Note that the quotes have disappeared.
18	The comma, tokenized.
128	Our first variable! Operator tokens over 127 are variables. The variable number (in the variable table) is 128 less than the token value. This variable is **THIS**.
36	The multiplication operator.
43	One variety of left parenthesis. This one is a normal or expression left parenthesis.
14	Another special token (actually, number 2

Figure 1.

BASIC Statement	Action performed
GRAPHICS g	If bit 4 ($10) of 'g' is on, this is the same as **OPEN #6, 12, g-16, "S:"** **If the bit is off, this is the same as OPEN #6, 16 + 12, g, "S:"** (Note: the fifth bit, $20, of 'g' should be copied into AUX1, the OPEN mode.)
COLOR c	Simply saves 'c' in a safe place.
POSITION h,v	Places 'h' in locations $55 and $56 (LSB,MSB) Places 'v' in location $54
PLOT h,v	Performs a **POSITION h,v** and then Performs a **PUT #6,c** (where 'c' is the color saved by **COLOR**)
LOCATE h,v,c	Performs a **POSITION h,v** and then Performs a **GET #6,c**
DRAWTO h,v	Performs a **POSITION h,v** and then Does a **POKE 763, c** ('c' is the COLOR saved, as above) and then Performs an **XIO 17, #6, 12, 0, "S:"**
SETCOLOR r,h,lu	Is equivalent to **POKE 708 + r, h*h16 + lu**

Note: FILL may be performed from assembly language by following exactly the same sequence specified in the *Basic Reference Manual*, using XIO 18, etc.

Program 1: MAKEREL

```
100 REM *** OPEN ALL 3 FILES ***
110 OPEN #1,4,0,"D:OBJECT1"
120 OPEN #2,4,0,"D:OBJECT2"
130 OPEN #3,8,0,"D:DATA.REL"
150 REM *** INITIALIZE VARIABLES ***
160 LINE=10000
```

	of 2), says a numeric constant follows.
64 03 00 00 00 00	The constant, in Atari BASIC internal floating point form. This is unique, as we shall see soon.
37	An addition operator.
129	The variable **IS** (already known to be an array, though it has not yet been DIMensioned).
56	Another left parenthesis. This one is called an "array left paren" in the BASIC source listing. We will later see why it is distinct.
130	Our last variable, **FUN**.
44 44	Two right parentheses. Strange, they are both the same.
20	Our End-Of-Statement token, otherwise known as a colon.
36	The statement end displacement for the second statement on this line.
38	The token for STOP. Again, refer to the keyword listing program.
22	An End-Of-Line token, otherwise known as a RETURN.

Wasn't that fun? For a masochist? Hopefully, you are asking questions that begin with "Why."

Why tokenize at all? For compactness: in our example we saved six bytes over a straight source line. For speed: it is much faster (at run-time) to discover that, for example, 32 means "PRINT" than it would be if we had to examine the letters "P", "R", "I", "N", "T" for a keyword match. Because tokenizing is almost an automatic byproduct of syntaxing.

Why syntax-check at entry? Because it is embarrasing to give a program to someone, have them run it, and get a SYNTAX ERROR message at line 23776 (the line that handles disk full conditions, which we never got to when we were testing). Because it makes program entry so much easier for be-

```
170 DCNT=0
200 REM *** STRIP HEADER ($FFFF) WORD ***
220 GET #1,FF:GET #1,FF
230 REM STRIP HEADER AND ADDRESSES FROM FILE2
240 GET #2,FF:GET #2,FF:REM HEADER
250 GET #2,FF:GET #2,FF:REM START ADDRESS
260 GET #2,FF:GET #2,FF:REM END ADDRESS
300 REM *** PROCESS ADDRESSES ***
310 GET #1,LOW:GET #1,FIRSTHIGH:FIRST=LOW+256*FIRSTHIGH
320 GET #1,LOW:GET #1,HIGH:LAST=LOW+256*HIGH
400 REM *** READY TO PRODUCE OUTPUT ***
410 FOR ADDR=FIRST TO LAST
420   IF DCNT=0 THEN PRINT #3;LINE;" DATA ";:LINE=LINE+10
430   GET #1,B1:GET #2,B2
440   IF B1=B2 THEN 480
450   IF B2<>B1+1 THEN PRINT "BAD RELOCATION":STOP
460   B1=B1-FIRSTHIGH:REM THE RELOCATION FACTOR
470   PRINT #3:"*";:REM AND FLAG THIS BYTE
480   PRINT #3;B1;
490   DCNT=DCNT+1
500   IF DCNT<=9 THEN PRINT #3;",";
510   IF DCNT>9 THEN DCNT=0:PRINT #3
520   NEXT ADDR
530 REM *** CLEAN UP ***
540 IF DCNT=0 THEN PRINT #3;LINE;" DATA ";
550 PRINT #3;"="
560 PRINT #3;"GOTO 500"
580 CLOSE #1:CLOSE #2:CLOSE #3
590 END
```

Program 2: LOADREL.A

```
10 REM *** THIS IS LOADREL.A ***
20 REM (THIS SIMPLY SETS UP MEMORY FOR LOADREL.B)
30 NUMBEROFPAGES=1:REM CHANGE THIS AS NEEDED
40 SIZE=256*NUMBEROFPAGES
100 REM *** SEE COMPUTE! #19 ***
110 LET LOMEM=743:MEMLOW=128
120 LADDR=PEEK(LOMEM):HADDR=PEEK(LOMEM+1)
129 REM -- LINE 130 ENSURES THAT 1K BYTES STARTS ON PAGE BOUNDARY --
130 IF LADDR<>0 THEN LADDR=0:HADDR=HADDR+1
140 ADDR=LADDR+256*HADDR
150 ADDR=ADDR+SIZE
160 HADDR=INT(ADDR/256):LADDR=ADDR-256*HADDR
170 POKE LOMEM,LADDR:POKE LOMEM+1,HADDR
180 POKE MEMLOW,LADDR:POKE MEMLOW+1,HADDR:RUN "D:LOADREL.B"
```

Program 3: LOADREL.B

```
100 REM *** THIS IS LOADREL.B ***
110 REM
120 REM THIS PROGRAM DOES THE ACTUAL RELOCATABLE LOAD
130 REM
140 DIM TEMP$(10)
150 NUMBEROFPAGES=1:REM ADJUST TO SAME AS LOADREL.A
200 REM AGAIN, SEE COMPUTE! #19
210 LET LOMEM=743:MEMLOW=128
220 POKE LOMEM,PEEK(MEMLOW):POKE LOMEM+1,PEEK(MEMLOW+1)
300 REM RPAGE IS THE MEMORY PAGE WHERE WE RELOCATE TO
310 RPAGE=PEEK(MEMLOW+1)-NUMBEROFPAGES
330 REM OBVIOUSLY, THIS VALUE SHOULD MATCH THE MEMORY
340 REM RESERVED IN 'LOADREL1.SAV'
350 ADDR=RPAGE*256:REM STARTING ADDR OF LOAD
```

ginners, particularly kids. Because I like it.

Why one-byte variable numbers? Again, for speed and compactness. Use variable names as long as you like: only the first usage eats up any more memory than a single-character, undecipherable variable name. There are disadvantages: a maximum of 128 different variables, a mispelled variable name can't be purged from the variable table without LISTing and reENTERing. On the whole, a very wise choice (*I* can say that, it's one part of Atari BASIC I *didn't* design into the specs).

Why internalized numeric constants? For speed. Period. Well, maybe for simplicity at runtime, but that's only a maybe. Did you know that numeric constants in Atari BASIC actually execute faster than variables? Write a timing loop and prove it to yourself.

Why line length bytes? Do you need them if you have statement length bytes? We don't *need* them, but they make line skipping (as when we are executing a GOTO) faster than it would be if we had to skip individual statements.

Why statement length bytes? Given that you have line length bytes? This one is harder to answer, because it has to do with how we execute GOSUB/RETURN, etc. I will leave that for a later article, but I will note that these bytes were extremely helpful when it came to implementing the **IF...ELSE...ENDIF** structure in BASIC A+.

Why decimal floating point? Because it is easier for beginners to understand (try **PRINT 123.123-123** using Applesoft) and is obviously preferable for money applications. Actually, our decimal add and subtract are faster than the corresponding binary routines. Admittedly, multiply suffers a little and divide suffers a lot.

Why different kinds of left parentheses? *Why* several kinds of equal sign? Because it's easy for the syntaxer to see the different

```
400 REM ****************************
410 REM *  GET THE RELOCATION DATA *
420 REM ****************************
450 ENTER "D:DATA.REL"
500 REM *** THE ENTER BRINGS US HERE ***
510 READ TEMP$
520 IF TEMP$(1,1)="=" THEN END
530 IF TEMP$(1,1)<>"*" THEN POKE ADDR,VAL(TEMP$):GOTO 550
540 POKE ADDR,VAL(TEMP$(2))+RPAGE:REM RELOCATION
550 ADDR=ADDR+1:GOTO 510
```

Program 4: DATA.REL

```
520 IF TEMP$(1,1)="=" THEN 1000
1000 REM LINE 1010 IS USED TO INITIALIZE THE M: DRIVER
1010 JUNK=USR(RPAGE*256+48)
1020 END
10000 DATA 162,0,189,26,3,240,10,201,77,240
10010 DATA 26,232,232,232,208,242,96,169,77,157
10020 DATA 26,3,169,59,157,27,3,169,*0,157
10030 DATA 28,3,169,0,157,29,3,169,0,141
10040 DATA 231,2,169,*1,141,232,2,96,104,240
10050 DATA 205,168,104,104,136,208,251,240,197,76
10060 DATA *0,111,*0,146,*0,133,*0,159,*0,73
10070 DATA *0,76,74,*0,160,1,96,189,74,3
10080 DATA 41,8,240,13,173,229,2,141,210,*0
10090 DATA 172,230,2,136,140,211,*0,173,210,*0
10100 DATA 141,206,*0,173,211,*0,141,207,*0,160
10110 DATA 1,96,189,74,3,41,8,240,12,173
10120 DATA 206,*0,141,208,*0,173,207,*0,141,209
10130 DATA *0,160,1,96,72,32,181,*0,104,160
10140 DATA 0,145,224,32,192,*0,96,32,160,*0
10150 DATA 176,7,160,0,177,224,32,192,*0,96
10160 DATA 32,181,*0,205,208,*0,208,9,204,209
10170 DATA *0,208,4,160,136,56,96,160,1,24
10180 DATA 96,173,206,*0,133,224,172,207,*0,132
10190 DATA 225,96,172,206,*0,208,3,206,207,*0
10200 DATA 206,206,*0,160,1,96,0,0,0,0
10210 DATA 0,0,=
```

Program 5: Graphics Routines, Equates

```
0000        1010         .PAGE "Equates, etc."
            1020 ;
            1030 ; CIO EQUATES
            1040 ;
E456        1050 CIO    =    $E456     ; Call OS thru here
0342        1060 ICCOM  =    $342      ; COMmand to CIO in IoCb
0344        1070 ICBADR =    $344      ; Buffer or filename ADdRess
0348        1080 ICBLEN =    $348      ; Buffer LENgth
034A        1090 ICAUX1 =    $34A      ; AUXilliary byte # 1
034B        1100 ICAUX2 =    $34B      ; AUXilliary byte # 2
            1110 ;
0003        1120 COPN   =    3         ; Command OPeN
000C        1130 CCLOSE =    12        ; Command CLOSE
0007        1140 CGBINR =    7         ; Command Get BINary Record
000B        1150 CPBINR =    11        ; Command Put BINary Record
0011        1160 CDRAW  =    17        ; Command DRAWto
0012        1170 CFILL  =    18        ; Command FILL (not used in this demo)
            1180 ;
0004        1190 OPIN   =    4         ; OPen for INput
0008        1200 OPOUT  =    8         ; OPen for OUTput
            1210 ;
            1220 ;
            1230 ; EQUATES used by the S: driver and
```

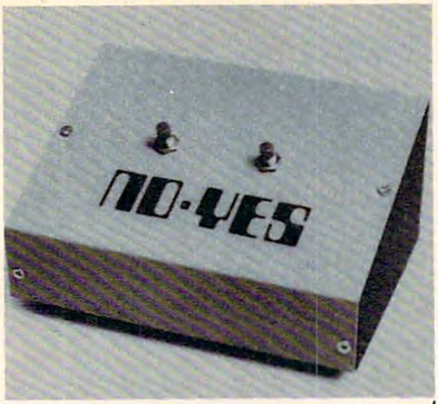
kinds of equal signs in, for example, **LET A = B = C + D$ = E$**. Sure, we could tell the difference at run time from context, but why should we when it's so easy to distinguish between a 45 and a 34 and a 52?

Why doesn't Atari BASIC have string arrays? I really didn't want to put this question in, but I wanted to save myself the letters and threatening phone calls. The best reason is that it was a choice of string arrays or syntax checking. (Obviously, I like the choice.) Other rationales include the fact that Atari was aiming for the educational market, where the HP2000 (with 72-character, Atari-style strings) was the *de facto* standard.

My personal favorite reasons are twofold: (1) anything you can do with string arrays you can also do with long strings (admittedly, sometimes with a little more difficulty) though the reverse is definitely not true; and (2) string arrays are unique to DEC/ Microsoft/??? BASIC and do not appear in that form in any other of the more popular languages (e.g., FORTRAN, COBOL, PASCAL, C, FORTH, etc.). Techniques learned with long strings are portable to these other languages: techniques involving string arrays are, at best, difficult to transfer. Finally, long strings as implemented on the Atari have some unique advantages not immediately obvious. I hope to explore some of these advantages in future columns.

```
               ;    the VBLANK routines
               ;
0055           HORIZONTAL = $55
0054           VERTICAL =   $54
02FB           DRAWCOLOR =  $2FB
02C4           COLOR0 =     $2C4
               ;
               ; miscellany
               ;
00FF           LOW     =     $FF
0100           HIGH    =     $100
               ;
Graphics routines for COMPUTE! #21
The actual routines

0000                  .PAGE "The actual routines"
               ;
               ; First, set the location and some miscellaneous
               ;      RAM usage
               ;
0000                  *=    $660
               ;
0660 00        SAVECOLOR .BYTE 0      ; where COLOR is saved
               ;
0661 53        SNAME   .BYTE "S:",0   ; the filename for open
0662 3A
0663 00
               ;
               ;
               ; GRAPHICS g
               ;
               ; ENTRY: A-reg contains graphics mode 'g'
               ;  EXIT: Y-reg has completion status
               ;
               GRAPHICS
0664 48               PHA             ; save 'g'
0665 A260             LDX  #6*$10     ; file 6
0667 A90C             LDA  #CCLOSE
0669 9D4203           STA  ICCOM,X
066C 2056E4           JSR  CIO        ; First, we must close file #6
               ; (we ignore any errors from the close)
               ;
066F A260             LDX  #6*$10     ; again, file 6
0671 A903             LDA  #COPN      ; we will open this 'file'
0673 9D4203           STA  ICCOM,X
0676 A961             LDA  #SNAME&LOW
0678 9D4403           STA  ICBADR,X   ; we use the file name "S:"
067B A906             LDA  #SNAME/HIGH
067D 9D4503           STA  ICBADR+1,X ; by pointing to it
               ;
               ; all is set up for OPEN, now
               ; we tell CIO (and S:) what kind of open
               ;
0680 68               PLA             ; our saved 'g' graphics mode
0681 9D4B03           STA  ICAUX2,X   ; is given to S:
               ; (note that S: ignores the upper bits of AUX2)
0684 29F0             AND  #$F0       ; now we get just the upper bits
0686 4910             EOR  #$10       ; and flip bit 4
               ; (Read the text.  S: expects this bit inverted
               ;    from what normal BASIC usage is.)
0688 090C             ORA  #$0C       ; allow read and write access (for CIO)
068A 9D4A03           STA  ICAUX1,X   ; make CIO and S: happy
068D 2056E4           JSR  CIO        ; and do the OPEN of S:
```



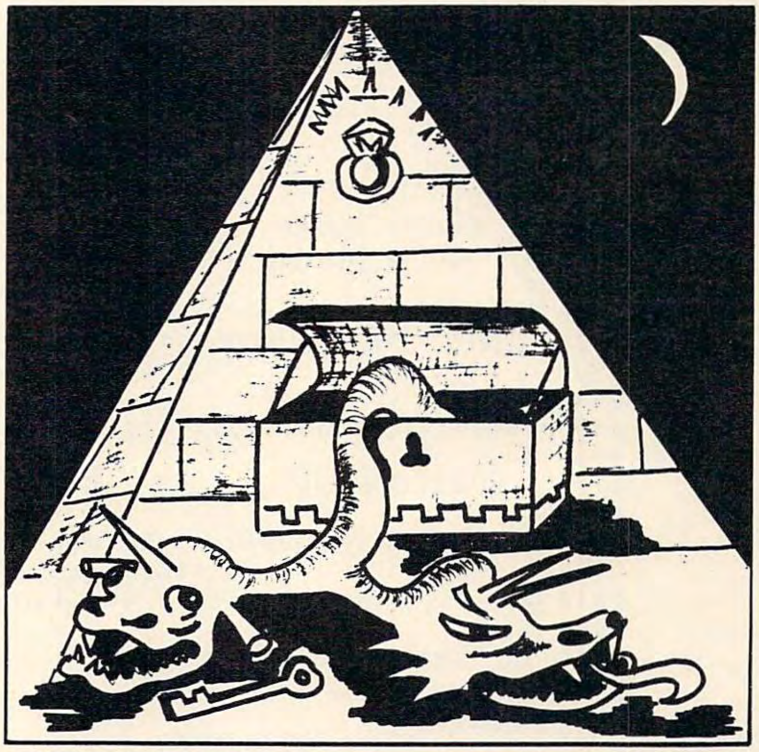
ANSWER-MAN

FOR USE WITH YOUR ATARI 400/800. PLUGS INTO THE GAME PORTS. GREAT FOR EDUCATION OR FOR FUN. COMES WITH SOFTWARE. GUN METAL GRAY CASE.

$29.95

LIFELINE COMM. INC.
3107 PICO BL. #1.
SANTA MONICA, CA.
90405. (213)828-0489.

Dealers – Reserve your copies of **COMPUTE**'s first Atari and PET/CBM books today. Call 919-275-9809 for ordering information.



CRYPTS OF TERROR

Beware as you enter the Crypts Of Terror. No one has survived this horror. Only your unrelenting nerve and determination will drive you deeper into the unknown.

Find what lurks in these ancient crypts!!

At last we have found an adventure with full graphics, sound and intrigue for your ATARI 400/800 computer.

• CRYPTS OF TERROR is the first adventure game that was completely designed for the Atari computers only. The graphics are the finest available using the full potential of the Atari.

ATARI ®

Atari 800/400 16K requires joysticks.
Payment: Personal Checks – allow three weeks for check to clear.
American Express, VISA, MasterCard – include all numbers on card. Please include phone number with all orders.
Orders from USA $29.95 (US funds)
Orders from Canada $39.95 (Canadian funds)
Plus $2.00 for shipping.

Ontario residents add 7% R.S.T.
Check your local computer dealer for Crypts Of Terror.
Dealer inquiries encouraged.

INHOME SOFTWARE

PH. 1-416-961-2760

1560 Yonge St.
P.O. Box 10
Toronto
Ontario Canada
M4T 1Z7

```
0690 60      1820         RTS
             1830 ;
             1840 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
             1850 ;
             1860 ; COLOR c
             1870 ;
             1880 ;  ENTER: Color 'c' in A-register
             1890 ;   EXIT: Unchanged
             1900 ;
             1910 COLOR
0691 8D6006  1920         STA  SAVECOLOR
0694 60      1930         RTS              ; exciting, wasn't it?
             1940 ;
             1950 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
             1960 ;
             1970 ; POSITION h,v
             1980 ;
             1990 ;  ENTER: h (horizontal) position in X,A
             2000 ;             registers (LSB,MSB)
             2010 ;         v (vertical) position in Y-register
             2020 ;
             2030 ;   EXIT: unchanged
             2040 ;
             2050 POSITION
0695 8655    2060         STX  HORIZONTAL
0697 8556    2070         STA  HORIZONTAL+1 ; read the text
0699 8454    2080         STY  VERTICAL   ; too simple, right?
069B 60      2090         RTS
             2100 ;
             2110 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
             2120 ;
             2130 ; PLOT h,v
             2140 ;
             2150 ;  ENTER: must have done a previous COLOR call
             2160 ;         X,A,and Y registers set as in POSITION
             2170 ;
             2180 ;   EXIT: Y-register has completion status
             2190 ;
             2200 PLOT
069C 209506  2210         JSR  POSITION
069F A260    2220         LDX  #6*$10     ; file 6, again
06A1 A90B    2230         LDA  #CPBINR    ; Command Put BINary Record
06A3 9D4203  2240         STA  ICCOM,X
06A6 A900    2250         LDA  #0
06A8 9D4803  2260         STA  ICBLEN,X
06AB 9D4903  2270         STA  ICBLEN+1,X ; if buffer length is zero...
06AE AD6006  2280         LDA  SAVECOLOR ; then CPBINR puts one char from A-reg
06B1 2056E4  2290         JSR  CIO        ; and this is how we PLOT
06B4 60      2300         RTS
             2310 ;
             2320 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
             2330 ;
             2340 ; LOCATE h,v,c
             2350 ;
             2360 ;  ENTER: X,A,and Y registers set up as in POSITION
             2370 ;   EXIT: A-register has the LOCATEd color
             2380 ;         Y-register has the completion code
             2390 ;
             2400 LOCATE
06B5 209506  2410         JSR  POSITION
06B8 A260    2420         LDX  #6*$10     ; file 6
06BA A907    2430         LDA  #CGBINR    ; Command Get BINary Record
06BC 9D4203  2440         STA  ICCOM,X
```

```
06BF A900   2450          LDA  #0
06C1 9D4803 2460          STA  ICBLEN,X
06C4 9D4903 2470          STA  ICBLEN+1,X ; if Buffer LENgth is zero,
06C7 2056E4 2480          JSR  CIO        ; then the character is returned in A
06CA 60     2490          RTS
            2500 ;
            2510 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            2520 ;
            2530 ; DRAWTO h,v
            2540 ;
            2550 ;  ENTER: must have done a previous PLOT
            2560 ;         X,A,and Y registers as in POSITION
            2570 ;
            2580 ;   EXIT: Y-register has completion code
            2590 ;
            2600 DRAWTO
06CB 209506 2610          JSR  POSITION
06CE AD6006 2620          LDA  SAVECOLOR
06D1 8DFB02 2630          STA  DRAWCOLOR ; where DRAWTO expects its color
06D4 A260   2640          LDX  #6*$10     ; file 6...once more
06D6 A911   2650          LDA  #CDRAW     ; just a command to "S:"
06D8 9D4203 2660          STA  ICCOM,X
06DB A90C   2670          LDA  #$0C
06DD 9D4A03 2680          STA  ICAUX1,X   ; insurance
06E0 A900   2690          LDA  #0
06E2 9D4B03 2700          STA  ICAUX2,X   ; ...guaranteed to work
06E5 2056E4 2710          JSR  CIO        ; do the actual DRAWTO
06E8 60     2720          RTS
            2730 ;
            2740 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            2750 ;
            2760 ; SETCOLOR r,hue,lum
            2770 ;
            2780 ;  ENTER: X-register has color register 'r'
            2790 ;         A-register has hue
            2800 ;         Y-register has luminance
            2810 ;   EXIT: (undefined)
            2820 ;
            2830 SETCOLOR
06E9 0A     2840          ASL  A
06EA 0A     2850          ASL  A
06EB 0A     2860          ASL  A
06EC 0A     2870          ASL  A          ; we need hue * 16
06ED 9DC402 2880          STA  COLOR0,X   ; save it here for a nonce
06F0 98     2890          TYA
06F1 290E   2900          AND  #$0E       ; only luminance bits that matter
06F3 18     2910          CLC
06F4 7DC402 2920          ADC  COLOR0,X   ; end of the nonce
06F7 9DC402 2930          STA  COLOR0,X   ; and VBLANK will move this to hardware
06FA 60     2940          RTS
            2950 ;
06FB        2960          .END

Graphics routines for COMPUTE! #21
The actual routines

=E456 CIO         =0342 ICCOM      =0344 ICBADR       =0348 ICBLEN
=034A ICAUX1      =034B ICAUX2     =0003 COPN         =000C CCLOSE
=0007 CGBINR      =000B CPBINR     =0011 CDRAW        =0012 CFILL
=0004 OPIN        =0008 OPOUT      =0055 HORIZONTAL   =0054 VERTICAL
=02FB DRAWCOLOR   =02C4 COLOR0     =00FF LOW          =0100 HIGH
 0660 SAVECOLOR    0661 SNAME       0664 GRAPHICS      0691 COLOR
 0695 POSITION     069C PLOT        06B5 LOCATE        06CB DRAWTO
 06E9 SETCOLOR
```

©

P/M Graphics Made Easy

T. Sak, S. Meier
Baltimore, MD

Many people have called the Atari's graphics capabilities its best feature, especially the player-missile graphics. We won't argue, but how many of you have backed away because it looks too difficult to handle in BASIC or you simply are not satisified with the execution speeds which you are able to achieve?

Well, no more excuses! We've got a machine language subroutine that you can use with BASIC to achieve exciting graphics performance without a lot of muss and fuss. As a matter of fact, you make only one setup call to the subroutine and then forget it! And we promise you need know nothing about machine language. Just a few POKEs and you'll have your players dancing around the television screen.

You Don't Need To Know Machine Language

There have been a number of very helpful articles published describing the essential player-missile graphic information. Chris Crawford's description in **COMPUTE!** #8 is particularly noteworthy. We're going to assume that you are familiar with the fundamentals, but we'll review highlights as they're required.

A feature of the Atari with which you may not be familiar is its "interrupt" mechanism and how you can let it move your players for you at machine language speed — without the overhead of calling it from your BASIC program. Before we explore this useful feature, let's take a quick refresher course on interrupts.

As you know, the Atari keeps itself pretty busy doing its "housekeeping" chores even while it is interpreting your BASIC program. Among other things, the Atari must maintain the steady delivery of information to your television set, allowing it to paint a constantly up-to-date picture of the display data. Multiple, concurrent activities are performed by allowing one particular activity to periodically interrupt another.

The traditional analogy is that of a busy business executive who, while engaged in a meeting with an associate, is interrupted by a telephone call. The ringing phone signals the interrupt; the executive "checkpoints" his meeting and answers the phone. After disposing of the call, the executive resumes his meeting at the point of interruption.

A similar circumstance occurs each time a complete picture is painted by your television set. The television's electron beam paints the picture by sweeping horizontal rows across the picture tube beginning in the upper left hand corner and ending in the lower right. The beam is turned off when it reaches the lower right corner and is returned to its upper left starting position. This return trip is essentially a vertical positioning movement so this period when the beam is turned off is known as the *vertical blank time.*

Move During Vertical Blanks

The onset of the vertical blank cycle serves as an opportunity for the Atari's antic chip to signal an interrupt, the vertical blank or VBLANK interrupt. The operating system uses this occasion to perform some of its "housekeeping" duties. Fortunately, the operating system designers allow us to include a machine language subroutine which can be executed as one of these tasks.

The machine language vertical blank interrupt player movement subroutine described here is called VBLANK PM and it allows you to simply POKE the next x and y coordinate at which your player is to be displayed. There is no need to repeatedly call the subroutine from BASIC via the USR function. The subroutine will be automatically executed during the next vertical blank period. It is possible to move the players every time a new screen is painted on the television – and that's 60 times a second!

You may recall from other articles that an appropriate POKE to location 53248 (and the three memory locations following) permits you to position players zero through three horizontally along the x-axis. It's not quite as easy to position the players vertically along the y-axis. Not until now!

The VBLANK PM subroutine takes care to move the players in both directions. Movements along the vertical axis involve "erasing" and rewriting the player in the new position. VBLANK PM does this for you, automatically. There are a few things which you must do for VBLANK PM however.

First, you must get the VBLANK PM machine language subroutine into memory and notify the operating system that is to be included as one of the "housekeeping" tasks to be performed as a part of servicing the vertical blank interrupt. Next, it's up to you to draw your players and tell VBLANK PM how tall they are. After initialization, VBLANK PM looks after the positioning of your players until either a warm start (pressing SYSTEM RESET) or a cold start (power-off, power-on sequence) is performed.

Santa Cruz Educational Software * Tricky Tutorials * Santa Cruz Educational Software * Tricky Tutorials * Santa Cruz Educational Software * Tricky Tutorials

ATARI (tm) GRAPHICS AND SOUND MADE EASY!

SANTA CRUZ EDUCATIONAL SOFTWARE HAS WRITTEN A SERIES OF AFFORDABLE PROGRAMS THAT DEMONSTRATES MANY OF THE SPECIAL "TRICKS" THAT ONLY THE ATARI 400/800 COMPUTER IS CAPABLE OF DOING. WE OFFER EVERYTHING FROM A PROGRAM THAT DOES FANTASTIC HIGH RESOLUTION GRAPHICS TO ONE THAT ACTUALLY DIALS YOUR PHONE.....

****

MASTER MEMORY MAP(tm) — This is really the key to using the ATARI'S capabilities. We start out by explaining how to PEEK and POKE values into memory so that even new programmers can use this. Then: we give you over 15 pages of the memory locations that are the most useful. The information is condensed from both the ATARI'S Operating System Manual and various articles and programs. It is, of course, useful even for experienced programmers as a reference. Also, we highly suggest that dealers offer this Memory Map to customers who request to be told how to use the power of the machine. We guarantee it will answer many of the questions you have about the machine. **$5.95**

TRICKY TUTORIALS(tm)

#1: DISPLAY LISTS — This program teaches you how to alter the program in the ATARI that controls the format of the screen. For example: when you say graphics 8 the machine responds with a large graphics 8 area at the top of the screen and a small text area at the bottom. Now, you will be able to mix the various modes on the screen at the same time. Just think how nice your programs could look with a mix of large and small text, and both high and low resolution graphics. this program has many examples plus does all of the difficult caluculations! **$14.95**

#2: HORIZONTAL/VERITCAL SCROLLING — The information you put on the screen, either graphics or text, can be moved up, down or sideways. This can make some nice effects. You could move only the text on the bottom half of the screen or perhaps create a map and then move smoothly over it by using the joystick. **$14.95**

#3: PAGE FLIPPING — Normally you have to redraw the screen every time you change the picture or text. Now you can learn how to have the computer draw the next page you want to see while you are still looking at the previous page, then flip to it instantly. You won't see it being drawn, so a complicated picture can seem to just appear. Depending on your memory size and how complicated the picture, you could flip between many pages, thus allowing animation or other special effects with your text. **$14.95**

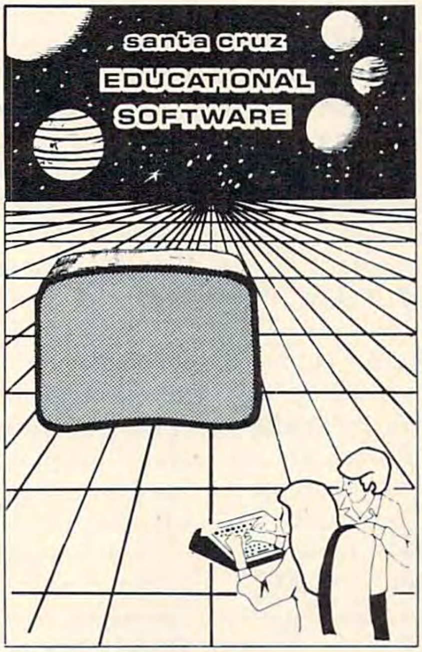
#4: BASICS OF ANIMATION — Shows you how to animate simple shapes using the PRINT and PLOT commands, and also has nice little PLAYER/MISSILE Graphics demo to learn. This would be an excellent way to start making your programs come alive on the screen. Recommended for new owners. **$14.95**

#5 PLAYER MISSILE GRAPHICS — This complex subject is demonstrated by starting with simple examples, and building up to a complete game and also an animated business chart on multiple pages! As always, the computer does most of the calculations. Requires 32K disk or tape and costs **$29.95**

#6: SOUND — From explaining how to create single notes, to demonstrating complex four channel sound effects, this newest tutorial is great. Even those experienced with ATARI's sound capabilities will find the menu of sound effects a needed reference that can be used whenever you are in the need of a special sound for your programs. Everyone will learn something new! Written by Jerry White. **$14.95**

Tricky Tutorials (except #5) require 16K memory for cassette orders and 24K for disk. The price is $14.95 each. You may order 1,2,3, & 4 for $49.95! All six in a colorful binder cost $89.95.

THE GRAPHICS MACHINE!! — Turn your computer into an incredible graphics tool with advanced commands like circle, box, fill, polygon, line, help, etc. 3 colors in graphics 8 with instant text!!! Create colorful business charts or beautiful drawings and then save or retrieve them from disk in 5 SECONDS. YES, it's that fast. Needs all 48K disk, and costs **$19.95**



MINI-WORD PROCESSOR — This is for those of you who have a printer, but don't want to spend $100 or more for a fancy word processor. It is suitable for simple editing of text, accepts most control characters for your printer, and text is stored on disk for easy retrieval. Holds 2½ typed pages at a time. Requires 32K, disk or tape. **$14.95**

BOB'S BUSINESS — 14 small business type programs for home or office, all chosen from a nice menu. Supports printed output. 169 sectors of output require 16K tape, or 32K disk. **$14.95**

KID'S #1 — Includes the following: 1) TREASURE — search for the lost treasure while trying to keep from falling into the sea. Nice graphics if you find it!; 2)DIALOGUE — talk back to the computer about four subjects; 3) MATH QUIZ — Nice musical and graphical rewards for good scores. Parents input the level of difficulty.

KID'S #2 — A spelling quiz, a "scrabble" type game, and a version of Touch with the computer giving all the directions! Both Kid's programs require 16K tape or 24K disk and cost **$14.95** each.

MINI-DATABASE/DIALER — This unique new program stores and edits up to 8 lines of information such as name, address, and phone numbers, or messages, inventories or anything you want. It has the usual sort, search, and print options, but it also has an unusual feature. If your files include phone numbers and you have a touch-tone phone, the program will DIAL THE PHONE NUMBERS FOR YOU! This is perfect for those who make a lot of calls like salesmen, teens, or those trying to get through to busy numbers (acts as an auto-redialer). It is also a lot of fun to use. Requires 16K cassette or 24K disk and costs **$24.95**

FONETONE — For those who only want to store name and phone numbers and have the dialer feature as above, we offer this reduced version. Same memory requirements, but only costs **$14.95.** Don't forget you must have a touch-tone phone.

PLAYER PIANO — Turns your keyboard into a mini-piano and more. Multiple menu options provide the ability to create your own songs, save or load data files using cassette or diskette, fix or change any of up to 400 notes in memory, and play all or part of a song. The screen displays the keyboard and indicates each key as it is played from a data file or the notes you type. You don't have to be a musician to enjoy this educational and entertaining program. Requires 24K cassette or 32K disk. **$14.95**

BOWLERS DATABASE — Provides the league bowler with the ability to record and retrieve bowling scores providing permanent records. The data may then be analyzed by the program and displayed or printed in summary or detail form. Data may be stored on cassette or diskette and updated quickly and efficiently. The program proivdes such information as highest and lowest scores by individual game, (first, second, and third games throughout the season), high and low series, current average, and more. The program listing and documentation provided are a tutorial on ATARI basic and record keeeping. Requires 16K for cassette or 24K for disk. **$14.95**

By the time you read this all computers (400/800) being produced should have the fabled GTIA chips included. ATARI service may upgrade older computers... call and ask (it's easy to do yourself). We have one and the improvements that graphics modes 9,10, and 11 offer are great!! To help you figure out what to do with the new modes a new Tricky Tutorial will be offered in March on Modes 9 to 11. Either give us a call or write around that time.

****

SEE YOUR LOCAL DEALER
OR
ORDER FROM:

SANTA CRUZ EDUCATIONAL SOFTWARE
5425 Jigger Dr., Soquel, CA 95073
C.O.D. Orders call (408) 476-4901
*Include $2.00 postage ($.75 for Memory Map)
*In California, include 6.5% tax

Santa Cruz Educational Software * Tricky Tutorials * Santa Cruz Educational Software * Tricky Tutorials * Santa Cruz Educational Software * Tricky

Program 1 is an example of the initialization and use of the VBLANK PM subroutine. This program causes VBLANK PM to be loaded and initialized and players zero and one to be drawn and then moved about the television screen in a random pattern. The players are male and female gender symbols which the program "dances" around the screen.

Lines 100 through 200 are the main program; we'll save an explanation of these lines until after you've gained some insight into the initialization subprogram contained in lines 1000 through 1110. The VBLANK PM machine language subroutine is expressed in the DATA statements numbered 2000 through 2100. Finally, lines 3000 through 3020 supply a description of the two players used in this example.

The first task is to load VBLANK PM into page six of memory. Page six is locations 1536 through 1791 (hexadecimal 600 through 6FF) and has been left available by Atari's software designers for applications such as this one. These 256 bytes of memory are not disturbed by BASIC or DOS; however, a cold start does cause page six to be cleared to zeroes. Line 1010 causes the VBLANK PM to be read and POKEd into memory. Line 1020 clears a few locations used by the subroutine; this statement can be omitted if you are sure that page six has not been altered since the last cold start.

We're going to employ the Atari's antic chip direct memory access (DMA) facility to transfer graphics information from memory to the television using single line resolution. (You might want to reread Chris Crawford's article or just "trust us on this one!") This means that we must allocate 2K (2048) bytes of memory for the storage of players. In line 1030 we obtain the page number of RAMTOP, deduct 16 pages, and call the result the base of the required 2K byte allocation.

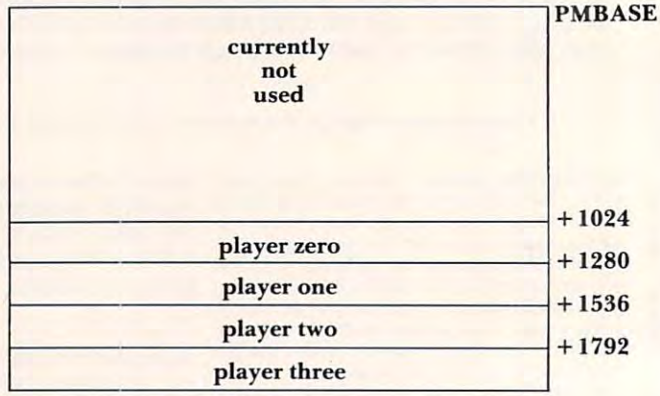
Memory Allocation

Why 16 pages? Well, first consider that 2K bytes are eight pages (a page contains 256 bytes) and that, depending on the graphics mode (i.e., GRAPHICS 0 through GRAPHICS 8), you must allow sufficient space at the top of RAM to contain the display list and screen data. Incidentally, the player-missile 2K byte allocation must begin at an address which is a multiple of 2048; we call this starting address PMBASE.

One more cautionary note: you will have to allow more than 16 pages between PMBASE and RAMTOP if you are using graphics modes six through eight. Fred Pinho's article in **COMPUTE!** #16 provides greater detail in this area.

Figure 1 depicts the 2K byte memory allocation.

Figure 1.



Remember, we didn't design this scheme, Atari did, and we're not sure why but there is a considerable amount of unused space involved. You can use the lower, unused bytes for your own purposes without disturbing anything, if you like. We're only going to use the upper 1K bytes.

Player zero occupies PMBASE+1024 through PMBASE+1279; player one is situated in locations PMBASE+1280 through PMBASE+1535, and so on for players two and three. Line 1040 clears any residual data – if you're in a hurry and are sure that this area is already clear (i.e., following a cold start), you won't need line 1040.

Lines 1050 and 1060 are used to draw players zero and one. VBLANK PM expects the players to be drawn such that their top line is initially placed at the beginning of the individual player's storage area. The player can be as tall as you like up to 255 lines; of course, you will never see all of a player which is that tall on the screen at the same time!

Next you can see that we've taken advantage of the Atari's special memory locations for some functions. You establish the players' colors with a POKE into locations 704 through 707 for players zero through three, respectively. Line 1070 is used to set the colors and assumes that you've set the variables PCOL0, PCOL1, PCOL2, and PCOL3 already.

Line 1080 establishes the positioning addresses which you will be using later to signal player movements using only POKEs. PLX and PLY are the locations POKEd to establish the next x and y position of player zero. A POKE into location PLX+1 and PLY+1 accomplishes the same thing for player one, and so forth for players two and three. PLL (and PLL+1, PLL+2, and PLL+3) are POKEd to inform VBLANK PM of the length (or height) of each player.

Line 1090 initializes the remaining control parameters. A 62 is POKEd into location 559 to set the single line player-missile resolution graphics; a one placed into location 623 establishes the player/playfield priorities giving the players priority over the playfield. (You can change this to suit your purposes, if you wish.) Location 1788 is in VBLANK PM and is POKEd with the number of the first page containing player-missile data. Locations 53277 and 54279 are used to switch on the DMA graphics data transfer facility and to tell the ANTIC chip where in memory to find the player graphics data.

Wrapping Up The Loose Ends

You're almost ready to go! A subroutine call to VBLANK PM from line 1100 allows VBLANK PM to notify the operating system of both his presence and his desire to be automatically invoked as a part of the vertical blank interrupt process. This is the only time in which your BASIC program must explicitly call VBLANK PM.

Okay, to wrap up loose ends, let's take a quick look at the main program – lines 100 through 200. Line 100 turns off the cursor, clears the screen, and provides a black background so that we can readily see the players.

Line 110 sets the players' colors before the VBLANK PM initialization subprogram is executed. You know how to set the colors, right? Multiply the color number by 16 and add the desired intensity – the color and intensity numbers are the same as those used in the SETCOLOR command. Line 120 assures the VBLANK PM is launched.

Line 130 illustrates the manner in which you pass instructions to VBLANK PM. Here we are telling VBLANK PM that both players are eight lines tall. You can change this parameter at any time – we have a little surprise for you later about why you might want to change this parameter.

Lines 140 and 150 establish the initial television screen positions of players zero and one, respectively. A word about the available values for the x and y coordinates might be helpful as not all x and y values will result in the player being displayed. There are 255 x positions with only 160 of these appearing across the television screen beginning with an x value of 48.

Similarly, there are 255 y positions with 192 of these visible on the screen beginning with 32 at the top. (These x and y values may vary slightly depending on the adjustment of your television receiver.) VBLANK PM assumes that you are referring to the upper left hand corner of your player whenever you POKE new x and y coordinate values.

Lines 170 and 180 illustrate the use of the pseudo-random number function to determine the next set of x and y coordinates. Line 190 provides a small delay between player movements. Delete the FOR and NEXT statements if you want to see how fast – and easy – it is to move players.

Well who said player-missile graphics had to be anything but fun?! Give VBLANK PM a try in one of your current programs to add a little zip; or try it in your next graphics project.

Oh, we almost forgot that we promised you a surprise regarding why you might want to change the height of a player. VBLANK PM has a few more features which allow you to animate the movements of your players – but more about this next time!

```
100 POKE 752,1:PRINT CHR$(125):SETCOLOR 2,0,0
110 PCOL0=216:PCOL1=56:REM color of players
120 GOSUB 1000:REM initialize vb routine
130 POKE PLL,8:POKE PLL+1,8:REM player's height
140 POKE PLX,108:POKE PLY,102:REM player
 0's initial position
150 POKE PLX+1,108:POKE PLY+1,72:REM ditto player 1
160 REM let players dance!
170 POKE PLX,RND(0)*159+48:POKE PLY,RND(0)*191+32
180 POKE PLX+1,RND(0)*159+48:POKE PLY+1,
RND(0)*191+32
190 FOR I=1 TO 75:NEXT I:GOTO 170
200 END
1000 REM INITIALIZE VBLANK PM SUBR
1010 FOR I=1536 TO 1706:READ A:POKE I,A:NEXT I
1020 FOR I=1774 TO 1787:POKE I,0:NEXT I
1030 PM=PEEK(106)-16:PMBASE=256*PM
1040 FOR I=PMBASE+1023 TO PMBASE+2047:PO
KE I,0:NEXT I
1050 FOR I=PMBASE+1025 TO PMBASE+1032:RE
AD A:POKE I,A:NEXT I
1060 FOR I=PMBASE+1281 TO PMBASE+1288:RE
AD A:POKE I,A:NEXT I
1070 POKE 704,PCOL0:POKE 705,PCOL1:POKE
706,PCOL2:POKE 707,PCOL3
1080 PLX=53248:PLY=1780:PLL=1784
1090 POKE 559,62:POKE 623,1:POKE 1788,PM
+4:POKE 53277,3:POKE 54279,PM
1100 X=USR(1696)
1110 RETURN
2000 REM vblank interupt routine
2010 DATA 162,3,189,244,6,240,89,56,221,
240,6,240,83,141,254,6,106,141
2020 DATA 255,6,142,253,6,24,169,0,109,2
53,6,24,109,252,6,133,204,133
2030 DATA 206,189,240,6,133,203,173,254,
6,133,205,189,248,6,170,232,46,255
2040 DATA 6,144,16,168,177,203,145,205,1
69,0,145,203,136,202,208,244,76,87
2050 DATA 6,160,0,177,203,145,205,169,0,
145,203,200,202,208,244,174,253,6
2060 DATA 173,254,6,157,240,6,189,236,6,
240,48,133,203,24,138,141,253,6
2070 DATA 109,235,6,133,204,24,173,253,6
,109,252,6,133,206,189,240,6,133
2080 DATA 205,189,248,6,170,160,0,177,20
3,145,205,200,202,208,248,174,253,6
2090 DATA 169,0,157,236,6,202,48,3,76,2,
6,76,98,228,0,0,104,169
2100 DATA 7,162,6,160,0,32,92,228,96
3000 REM players 0 & 1
3010 DATA 6,6,8,126,195,195,195,126
3020 DATA 126,195,195,126,24,126,126,24
```

©

Review:

Eastern Front (1941)

Edward P. McMahon
Potomac, MD

Eastern Front (1941) by Chris Crawford of the Atari Staff is a paradigm for computer war games. Not a shoot-em-up type arcade game, it is a corps-level historical simulation. The subject of this excellent simulation is the first 41 weeks of Operation Barbarossa, Hitler's massive attack on Russia which began on June 22, 1941.

Eastern Front has many features of a well-done historical simulation wargame: simultaneous movement of both players, supply rules, reinforcements and resupply effects, and effects of terrain. There is some time pressure also, which is not usually found in simulation wargames of the board-and-counter variety. The computer (a worthy opponent playing the Russian side) thinks out its move during the vertical blank periods when you are planning your moves. The more time you take, the better will be the computer's move. More on this later.

The game starts immediately after booting in (it is an AUTORUN.SYS file on the disk version), but first-time players don't immediately respond. They are entranced by the graphics presentation. The playfield is 2⅓ screens horizontally and 4⅓ in the vertical dimension and is filled with excellent redefined character sets – mountains, rivers, forests, marshes, cities and coastal areas. As you move your hollow square cursor to any edge of the screen window, the map smoothly fine-scrolls to display the correct part of the playfield. The attention to detail is admirable. The trees in the forest areas are different sizes; the rivers and coasts are displayed to the highest possible resolution. The colors have been carefully chosen – I have not noticed any "bleeding" between adjacent colors – and dramatically indicate the change of seasons. The autumn season begins on October 5, 1941, when the green land changes to a purple-brown mud color. (Remember that date. If you haven't captured your objectives, destroyed most of the original Red forces, and established a strong defensive position by then, you are in trouble.) The ground changes again to white in winter, and the rivers and marshes freeze (blue to white) from north to south as the weeks progress. The process reverses in the spring. Another very nice detail.

A few words on the history (*History of the Second World War,* Sir Basil Liddell Hart (ed), Marshall Cavendish USA, Ltd., 1973-1974). Hitler began open plans to invade Russia with discussions in June, 1940. A late spring offensive was planned, and the first strategy (by Maj. Gen. Marcks) was

...it is a corps-level historical simulation.

two thrusts – the largest to Moscow through Smolensk, the second to Kiev. These would join in a pincer movement, trapping most of the Red Army. General Halder and the German High Command modified the Marcks plan by weakening the Kiev thrust to strengthen the push to Moscow, and added a third line of attack to Leningrad. Three Army Groups were defined: Army Group North (von Leeb), Army Group Center (von Beck), and Army Group South (von Rundstedt). Von Kleist's I Panzergruppe and Guderain's II Panzergruppe were aimed north and south of the Pripet marshes respectively. The General Staff and probably Army Group leaders played out major war games in late 1940, taking both sides of the campaign. But early in December, Hitler made what the German Army War Diary calls "a substantial alteration." Leningrad became the principal military target and Moscow was to be taken afterward.

The aim was still rapid advance and encirclement to prevent the Red Army from escaping into the interior, and the destruction of Russia's industrial power in the Ukraine, in Leningrad, and in Moscow. But Hitler's modification had the Army Group Center waiting until Army Group North achieved its more difficult, more distant objective before going on to Moscow. The High Command did not argue successfully with Hitler, and the directive for Operation Barbarossa was signed on December 18, 1940.

On June 22, the longest day, the largest invasion in the world began against an army which had suffered Stalin's 1937-1939 purges: three of five Marshals, 13 of 15 Army Commanders, 57 of 85 Corps Commanders and more had been shot or disappeared without a trace. The German attacks were devastating in the North and Center (Smolensk fell on July 15, but Kiev held out as a pocket of resistance until late September). Nearly two-thirds of the Red Army's strength at the outbreak of the war was destroyed. The Germans occupied Russia up to a line from Leningrad to the Crimea.

Estimated losses by the end of 1941 for the Red Army were 5-7 million killed or wounded, 3-5 million P.O.W., 21,000 tanks and 33,000 guns destroyed. Russia fought back with extraordinary national effort, calling on all its resources and extensive Allied help. The Germans achieved some additional victories, but the Blitzkrieg was blunted by the vastness of Russia, the mud and the cold.

Your only hope of winning the simulation is to follow the suggestions of the author, Chris Crawford, in the excellent user's manual which comes with the game: break through and use the mobility of the armored units to encircle the Russian corps from behind, and concentrate forces by pushing your infantry as fast as possible to attack and eliminate pockets of enemy units. These are the classical Blitzkrieg tactics. But, before the autumn mud stops your panzers, form a defensive line using terrain (rivers and cities) to your advantage. Fall back in order during the winter counterattacks.

The game is well documented, but the mathematical rules of combat and supply are not given. The user interface is well designed, so the game is almost entirely playable from the joystick alone (three keys are needed: START, OPTION, and SPACE BAR). The only feature which I find seriously lacking is the ability to save a game and restart it. The game takes two or three hours to play (more, if you want to keep a record of what you are doing) and I find it difficult to come up with an uninterrupted block of time like that. Moreover, I can't study different moves for a given situation, but perhaps that's good. The unknowns can't be resolved, so the game keeps my interest. There is randomness in the combat and supply rules (a good feature) so a tactic which works today may not work tomorrow. Another reason why replaying a tactic may not give meaningful results was mentioned earlier: the computer works on its move while you are entering yours. The computer selects a move for each of its units and, as time is available, iteratively improves each move. This is the feature of the game in which Crawford takes the most pride.

Crawford states that he uses only 75% of the Atari's graphics capabilities. He should know. He is one of Atari's most creative staff members, and certainly understands the machine. The way he uses that 75% makes *Eastern Front (1941)* a showpiece and a challenge to other program designers. (If he ever uses 100%, I think I'll sit and stare for a week or so.)

I am still experimenting with small, local tactics, as hopeless as that may be. If I ever get to March '42 again (before 2 a.m.). perhaps a late winter thrust to push some muster points farther west will add some victory points. Hmmm... ©

Odds & Ends

Clearing Memory

Charles Brannon
Editorial Assistant

Before using an area of memory for storage, it is often necessary to clear it out. For example, a GRAPHICS command clears the screen by writing zeros to all the screen memory. Since there are no BASIC statements that directly support player/missile graphics, the memory used by this facility has to be cleared by the programmer, usually with a FOR/NEXT loop.

ATARI BASIC does not clear out the old values of an array or string when a program is RUN, even if there is garbage in the memory used by these variables. This also necessitates some kind of loop to clear out this memory.

The problem with this is that an array of any substantial size requires a long time to clear out. For strings, there is a shortcut:

```
10 DIM A$(100)
20 A$(1)=" ":A$(100)=" ":A$(2)=A$
```

Line 20 will "instantly" fill A$ with spaces. The space in quotes can be changed in order to fill a string with a desired character.

There is a "quick and dirty" way to clear out memory. This relies on the previously mentioned GRAPHICS command. GRAPHICS 8+16:GRAPHICS 0 will clear out about 8K of high RAM. If executed before a DIM statement, this will usually suffice. Since most Player/Missile memory is in the top of memory, the GRAPHICS command is definitely satisfactory. If you don't have 8K of free memory, you'll get an ERROR-147 (Insufficient RAM for GRAPHICS mode), in which case you'll have to use GRAPHICS 7+16 (or lower), or resort to the BASIC clear loop. ©

COMPUTE! Overview:

Textwizard

Textwizard consists of a copy- and write-protected disk containing the program, an instruction manual, and a reference card which summarizes all commands. All of this is in a luxuriously padded binder.

Original purchasers may request a back-up disk when the warranty card is returned with a $5.00 check. After that, clobbered discs can be exchanged at the factory for a $30.00 fee.

Forewarned, we treated the disk with great respect. What we found was a carefully human-engineered word processing system for a personal computer. It will not turn your Atari into a $15,000 professional word processor, but many of the important differences between Textwizard and a professional system are a function of the 40-character display. Thus, it is not reasonable to consider constructing large organizational charts, flow charts, or even moderately complex graphs. Tables of data which span more than 40 columns are tortuous to create. Other features usually seen in professional systems, including automatic hyphenation, positioning of footnotes at the bottom of a page, tabular sorts, arithmetic functions and representation of mathematical symbols are not available on Textwizard.

The Manual

First impressions of the program come from the well made and finished looseleaf text which accompanies the program. Filled with 56 pages of instructions, including an index, the manual gives the feeling that the producers of this program care about quality.

The *Instruction Guide* is, however, for one panelist, the weakest element in the package. It was not carefully proofread nor carefully tested with naive users. There are simply too many instances where the user who methodically follows the step-by-step instructions is left hanging with a feeling of "What do I do now?"

In contrast to the above, one panelist was pleased with the manual: "The user's manual is a very good example of what in some educational circles is known as the "KISS" of knowledge (Keep It Short and Simple). The manual walks the user through each function in logical, explicit language, and is itself an excellent example of programmed learning."

"...the manual gives the feeling that the producers of this program care about quality."

Ease Of Use

The program uses the Atari DOS system and diskettes must be formatted before you can store data on them. Data saved to disk appears to be compacted prior to saving, thus increasing the amount of data which can be stored on each disk.

The program is written completely in machine language and is loaded without the left cartridge in place. This increases the amount of available memory for storage of the text. With 48K of memory installed in the computer, there is just over 30K left for text. Eighteen K Bytes of program seems somewhat large, considering the stated capabilities of the program; however, this size would certainly seem to take it out of the "kid's toy" category.

The program boots quickly and, unless the amount of text being held is substantial, no loss of speed is noticed. There is one quirk which appears when the amount of memory in use begins to exceed 5K. There is a delay in the text's appearance on the screen when the inserting command is being used. This can be somewhat disconcerting if the operator is a touch typist and is watching the screen. The nice part is that the letters are all picked up and, if the typing is done accurately, it will all eventually show up.

When first used, however, the program appears to lack features found in other word processors. The program is not menu driven. Consequently, a number of functions such as loading, saving, disk formatting, and drive # setting, require that the operator remember specific keyboard sequences. Some of these are quite lengthy and would be more easily used if they were contained on a menu. The Atari full cursor movement feature is well used; however, there are no provisions for fast single stroke movement from the center of a line to either end of the line. These are not essential, but

would speed editing. There is no scrolling ability which tends to limit the speed with which text can be edited. Offsetting this, to some degree, is the ability to move through the text in either direction using a combination of the OPTION key and the cursor arrow keys. The system is friendly. It may be conceptualized as two major programs: one is for creating and editing text, and one is for printing documents. With very few exceptions, commands to the editing component involve only two keystrokes.

Text formatting and printing functions utilize Atari's special keyboard controls (i.e., OPTION, SELECT, START) in conjunction with other keyboard characters which, when possible, relate to the particular function or mode required. For example, CTRL+T sends the cursor to the top of the page, CTRL+B sends the cursor to the bottom of the page; CTRL+M is used to Move text, CTRL+D to Duplicate text, CTRL+S for Search, etc. The use of characters such as BACKSP, INSERT and editing ARROWS will be familiar to Atari users and greatly facilitate operation – especially for the word processing beginner.

One particular feature proved extremely versatile. Margin control is set, not in inches or character widths, but in *dot* widths. Although this necessitates some computation by the user (150 dots=approx. 1 in.), it allows "fine tuning" when formatting a document; some very creative copy can be produced with such a tool. It would be a nice touch, perhaps, if Datasoft supplied a margin guage or ruler which translated dots into inches and/or character widths as a useful addition to the Command Reference Card – which, in itself, is very clear and precise.

One severe problem occurs with the buffer operation. There is no protection for information stored in the buffer. Thus, inadvertent loss of text can and does happen unless the operator takes careful note of buffer use. It seems best not to store text in the buffer for any length of time, but rather to use the buffer for simple movement operations only. The lack of a screen display for print formatting requires that the printer be used each time the operator wishes to see the actual formatting of the text. This can account for a significant loss of time and reams of paper being generated, when the actual formatting on the page must be seen. One panelist felt that this, combined with an inability to print single pages of a long text file, proved to be the most serious deficiency of the program.

Search, Merge, Disk Functions

The search feature does not work reliably, particularly when the string involves as few as three characters. For example, on two Model 800 systems a search for "he" (the string, h+e+space) not only correctly identified all occurrences of "he," but also incorrectly located embedded instances of "he" (as in "wherever") and, worse yet, totally inappropriate strings (e.g., "Ruth."). Search/replace operations were similarly plagued. Even when it behaves properly, global replacement is only semi-automatic: each "old phrase" must first be located; replacement with "new phrase" must then be manually verified.

Tab stops are preset to five spaces, and cannot be altered. This makes the construction of even 4- or 5-column tables of numbers overly cumbersome.

Generally Good Fatal Error Protection

It is generally very difficult for a user to make fatal errors. The exception to this rule is, however, important. File deletion is accomplished by simultaneously pressing OPTION D, followed by a file name. Even though the contents of the file still exist on the disk, they are thereafter forever inaccessible to the user. For an operation with such important consequences, it is reasonable to expect the system to help prevent the user from making devastating mistakes. Atari DOS requires an affirmative acknowledge prior to a file deletion. This feature probably should have been included.

Text requiring no special formatting features (pagination; underlining; centering; use of superscripts, subscripts, page numbers, boldface, etc.) may be printed to an Atari 825, Epson MX80, or Centronics 737 printers by issuing the command sequence OPTION P, followed by the filename. The print routine incorporates defaults for left, right, top, and bottom margins, proportional spacing, and right margin justification.

Overriding any of these defaults, or incorporating any of the many special features, requires the user to embed a command string within the text file. Features common to the entire file (e.g., placement of page numbers) are indicated on the first line in the text. Other commands (e.g., for centering) are embedded as they are needed. Unlike some other word processors, Textwizard does not permit underlining on the Epson MX-80. An approximation to boldface type may be made on the Epson, but not other printers; this restriction, too, is odd since Letter Perfect can produce boldface on all three printers. Generation of superscripts and subscripts is easy, but the instructions fail to mention that it cannot be accomplished on the Epson.

Pagination Is Especially Flexible

Arabic numerals may be placed automatically anywhere on the top or bottom lines of a manuscript. Since pagination may begin with any value, sections or chapters of a manuscript may be independently prepared.

A nice feature is the option to print text in