Sound Synthesis And The Personal Computer — Past, Present, And Future

COMPUTE!

$2.50
January
1983
Issue 32
Vol. 5, No. 1
63379 £1.85 in UK

The Leading Magazine Of Home, Educational, And Recreational Computing

Music, Sound, And The Personal Computer

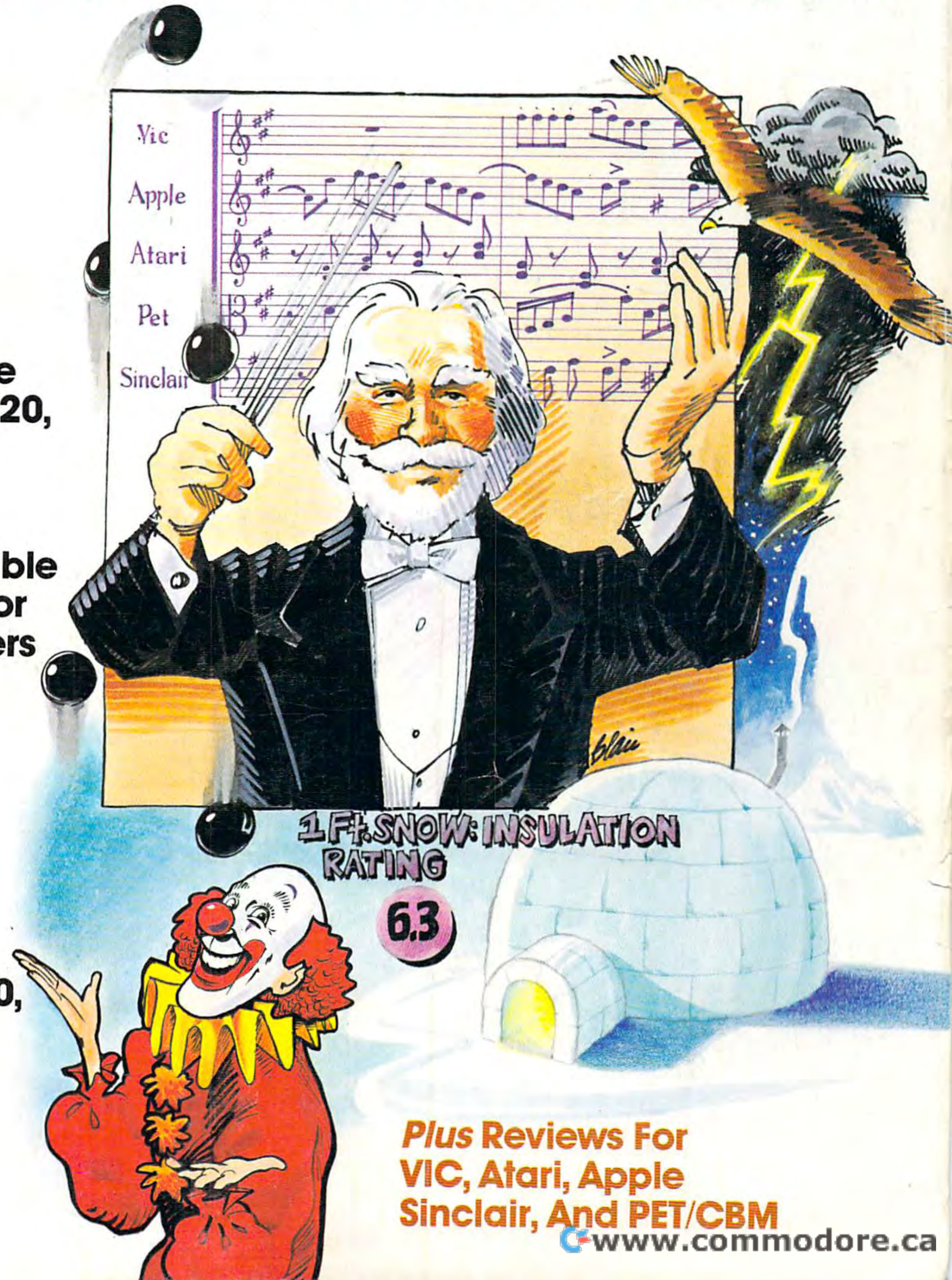
Music And Sound For VIC, Apple, Atari, PET/CBM, And The Sinclair/Timex

The Juggler, Thunderbird: **Two Exciting Game Programs For VIC-20, Atari, And TRS-80 Color Computer**

Writing Transportable BASIC Programs For Personal Computers

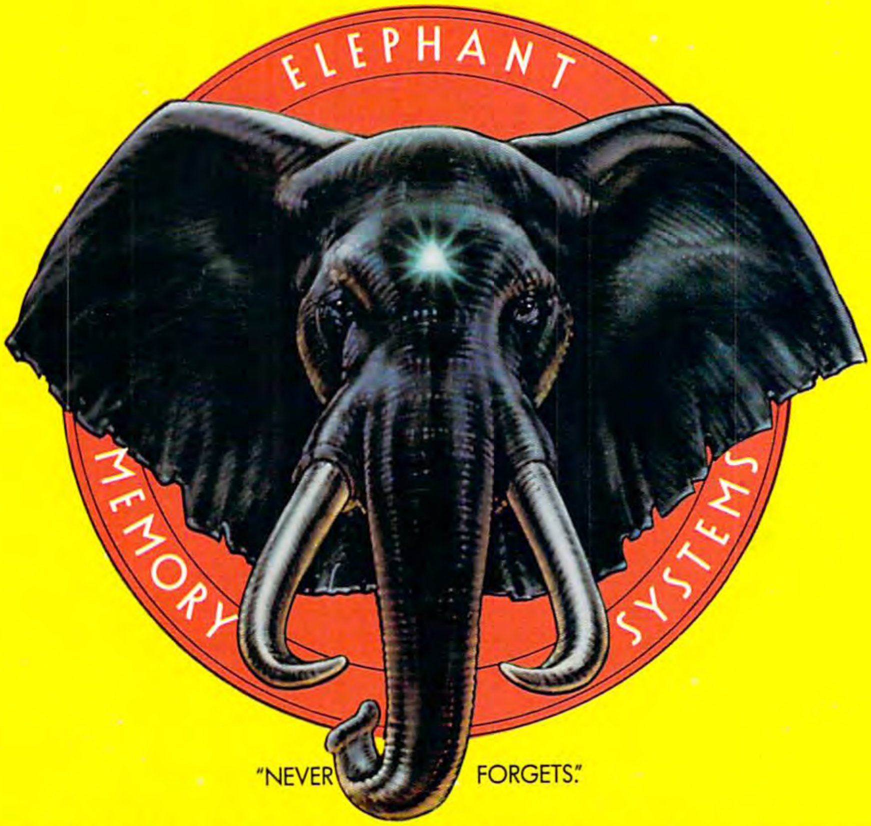
A New Monthly Column: **Programming The TI-99/4A**

A Home Energy Calculator For Atari, Apple, VIC-20, PET/CBM, And Commodore 64



Plus **Reviews For VIC, Atari, Apple Sinclair, And PET/CBM**

REMEMBER:



MORE THAN JUST ANOTHER PRETTY FACE.

Says who? Says ANSI.

Specifically, subcommittee X3B8 of the American National Standards Institute (ANSI) says so. The fact is all Elephant™ floppies meet or exceed the specs required to meet or exceed all their standards.

But just who is "subcommittee X3B8" to issue such pronouncements?

They're a group of people representing a large, well-balanced cross section of disciplines—from academia, government agencies, and the computer industry. People from places like IBM, Hewlett-Packard, 3M, Lawrence Livermore Labs, The U.S. Department of Defense, Honeywell and The Association of Computer Programmers and Analysts. In short, it's a bunch of high-caliber nitpickers whose mission, it seems, in order to make better disks for consumers, is also to make life miserable for everyone in the disk-making business.

How? By gathering together periodically (often, one suspects, under the full moon) to concoct more and more rules to increase the quality of flexible disks. Their most recent rule book runs over 20 single-spaced pages—listing, and insisting upon—hundreds upon hundreds of standards a disk must meet in order to be blessed by ANSI. (And thereby be taken seriously by people who take disks seriously.)

In fact, if you'd like a copy of this formidable document, for free, just let us know and we'll send you one. Because once you know what it takes to make an Elephant for ANSI . . .

We think you'll want us to make some Elephants for you.

ELEPHANT.™ HEAVY DUTY DISKS.

For a free poster-size portrait of our powerful pachyderm, please write us.
Distributed Exclusively by **Leading Edge Products, Inc.**, 225 Turnpike Street, Canton, Massachusetts 02021
Call: toll-free 1-800-343-6833; or in Massachusetts call collect (617) 828-8150. Telex 951-624.

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!

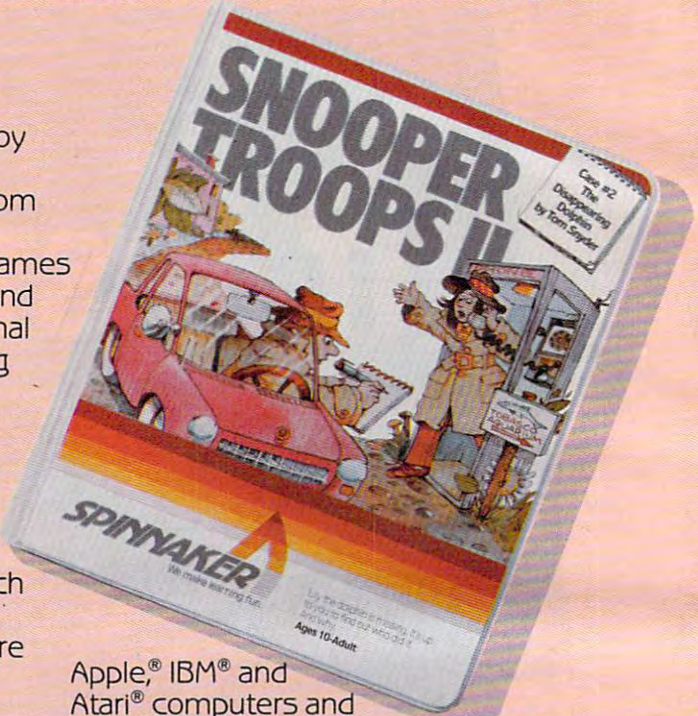
Our nationwide team of professional dealers will show you how your office will benefit by using WordPro PLUS. At a price far less than you realize.

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.
51 Fremont Street
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.
All specifications subject to change without notice.

Introducing Snooper Troops™ detective series. Educational games that turn ordinary homes into Sherlock homes.



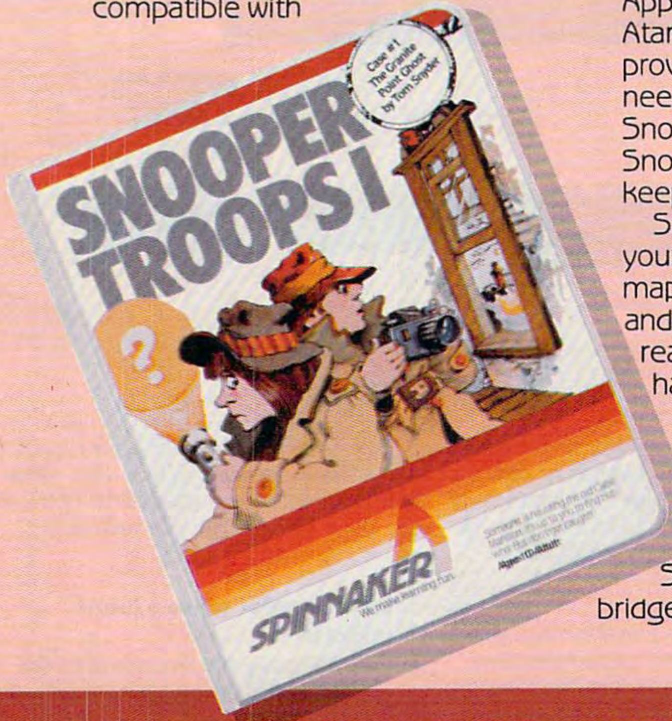
Where can you find educational games that your kids will really enjoy playing?

Elementary, my dear Watson. From Spinnaker.

Our Snooper Troops detective games are fun, exciting and challenging. And best of all, they have real educational value. So while your kids are having fun, they're learning.

As a Snooper Trooper, your child will have a great time solving the mysteries. But it will take some daring detective work. They'll have to question suspects, talk to mysterious agents, and even search dark houses to uncover clues.

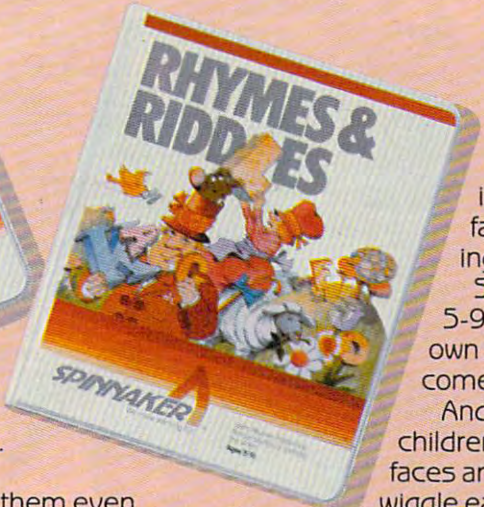
The Snooper Troops programs are compatible with Apple,® IBM® and Atari® computers and provide your kids with everything they need: a SnoopMobile, a wrist radio, a SnoopNet computer, a camera for taking Snoopshots and even a notebook for keeping track of information.



Snooper Troops detective games help your children learn to take notes, draw maps, organize and classify information and they help develop vocabulary and reasoning skills. All while your kids are having a good time.

So if you want to find educational games that are really fun, here's a clue: Snooper Troops games are available at your local software store, or by writing to: Spinnaker Software, 215 First Street, Cambridge, MA 02142.

Spinnaker's early learning games will help make your children as smart as you tell everyone they are.



Your kids are pretty smart. After all, they're *your* kids.

Spinnaker can help make them even smarter. With a line of educational software that kids love to play.

Spinnaker games make the computer screen come to life with full color graphics and sound. And they're fun. Lots of fun. But they also have real educational value.

Some of our games help exercise your child's creativity. Others improve memory and concentration. While others help to improve your child's writing, vocabulary, and spelling skills.

And every Spinnaker game provides familiarity with the computer and helps your children feel friendly with the computer. Even if they've never used a computer before.

And Spinnaker games are compatible with the most popular computers: Apple,® Atari® and IBM.®

Our newest game, KinderComp™ (Ages 3-8) is a collection of learning exercises presented in a fun and exciting manner.

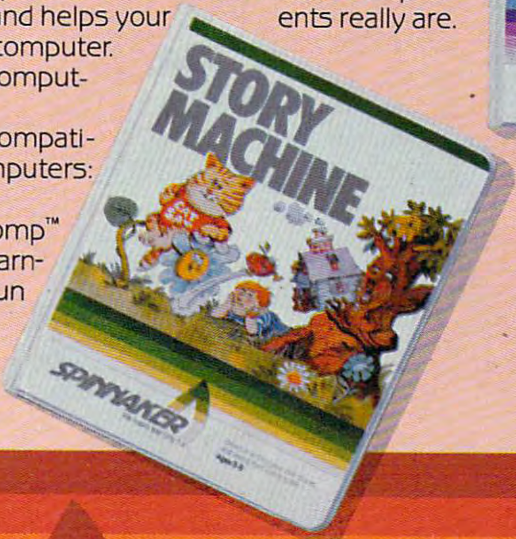
Rhymes and Riddles™ (Ages 4-9) is a letter guessing game featuring kids' favorite riddles, famous sayings and nursery rhymes.

Story Machine™ (Ages 5-9) lets children write their own stories and see them come to life on the screen.

And FACEMAKER™ lets your children create their own funny faces and make them wink, smile, wiggle ears (not your kids' ears, the ears on the screen), etc.

And we're introducing new games all the time.

So look for Spinnaker games at your local software retailer, or by writing to: Spinnaker Software, 215 First St., Cambridge, MA 02142. And show your kids how smart their parents really are.



SPINNAKER™
We make learning fun.

January 1983 Vol. 5, No. 1

FEATURES

EDUCATION AND RECREATION

REVIEWS

COLUMNS AND DEPARTMENTS

THE JOURNAL

NOTE: See page 222 before typing in programs.

GUIDE TO ARTICLES AND PROGRAMS

AP Apple, **AT** Atari, **P** PET/CBM, **V** VIC-20, **O** OSI, **C** Radio Shack Color Computer, **64** Commodore 64, **ZX** Sinclair ZX-81, ***** All or several of the above.

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, P.O. Box 5406, Greensboro, NC 27403. Second class postage paid at Greensboro, NC 27403 and additional mailing offices. Entire contents copyright © 1982 by Small System Services, Inc. All rights reserved. ISSN 0194-357X.

EDITOR'S NOTES

Announcing A Significant New Magazine From COMPUTE! Publications.

We've promised that 1983 would be an exciting year and are now willing to divulge one of the reasons why. *The Commodore Gazette*™ will premiere as a monthly in the spring of 1983. The Gazette will not impact **COMPUTE!** editorially or alter the current scope of **COMPUTE!**. The Gazette is planned as a laypersons's guide to consumer computing. It will be written for beginning and intermediate level owners and users of the VIC-20, 64, and Ultimax computers. Regular features will include best seller lists for recreational and educational software, reviews, new products, tutorials on home and educational applications, and much more. Written for entertainment as well as education, *The Commodore Gazette*, while appealing to users wishing to learn more about programming and computers, will also have continuing appeal for those who simply want to obtain maximum use from their computers in a non-technical way. Next issue we'll give you full details on the new magazine. **COMPUTE!** will continue to present its normal excellent range of information for the VIC-20 and Commodore 64.

A Call For Editors

COMPUTE! Publications, both our magazine and book publishing divisions, is looking for experienced staff members for our growing editorial needs. If you've been writing for **COMPUTE!**, or if you have meaningful editorial experience, we'd like to see a resumé as soon as possible. We're specifically interested in writers with experience using Atari, VIC-20, and related computer hardware. We are a progressive and growing company, with an excellent working environment and benefits, located in the attractive Piedmont area of central North Carolina. If you're interested, please send a resumé along with work history, salary expectations, and other pertinent information to Kathleen Martinek, Managing Editor, **COMPUTE!** Publications, Post Office Box 5406, Greensboro, NC 27403. Your inquiry will be treated with complete confidence. Mark the envelope "Personal and Confidential," please. Remember to include samples of your writing.

We cannot accept any telephone calls prior to submission of a resume, and will deal only with the individual interested in the position. We do not wish to work with personnel agencies.

Random Asides

You'll notice several enhancements in this issue as part of our continuing quest to better serve our readers. Among these are additions aimed at beginners. "Questions Beginners Ask" and a revised section on using our program listing conventions will become regular features of **COMPUTE!**.... We're setting new records again. Press run for this issue is an astonishing 156,000 magazines. We had to declare October, November, and December *sold-out* within weeks of publication date. It was only a few months ago we were collectively applauding the 100,000 mark... New personal computers are on the way from Mattel and NEC, among others... Atari may be dropping the rumored 600 given recent changes in competitor pricing. Look for a new competitor to the Commodore 64... Magnum publishes a monthly list of the 100 best-selling computer books in the US. It's compiled from industry sources. **COMPUTE! Books** has five titles in the top 100, and *COMPUTE!'s First Book of VIC* is number one for the second month in a row.

Reader Service Cards

A first for **COMPUTE!**. In the back of this issue, you'll find reader service cards for the very first time. Use them to request additional information from advertisers. Simply look up the advertiser in the advertising index and circle the appropriate number. Remember though, if you're in a hurry to contact a particular advertiser, it's probably best to write or call directly.

COMPUTE!'s New Look

As you'll notice as you explore this issue, we've made some subtle and significant changes in the overall design of the magazine. We think you'll find **COMPUTE!** even easier to read and enjoy. Thanks to everyone here for helping implement these changes.

Robert Lock

The Home Accountant.™ The #1 best-seller.

Any home finance package will balance your checkbook. But to become the #1 best-seller you've got to be something special.

The Home Accountant™ is.

It's the only one that prints a net worth statement and a personal finance statement. So you know exactly where you stand financially every day of the year. It will even print your checks, automatically.

Not only that, The Home Accountant™ lets you label every transaction. Just imagine sitting down to do your taxes and having every penny you've spent and earned neatly listed by category—and available at the touch of a button. It's an incredible time-saver.

You can also create bar, line and trend analysis graphs for every category—in color. It's great for realistic budgeting.

Sound amazing? Wait, there's more.

Let's say you write a check to pay your Visa. The Home Accountant™ automatically debits your checking account and credits your Visa account.

And it does this with every one of the two hundred* budget categories: credit cards, checking accounts, money markets, cash, rent checks, insurance payments—you customize your own financial package.

Check out The Home Accountant™ soon. You'll find it does a lot more than simply manage your money.

It manages your money simply.

*The Home Accountant™ is available for the Apple II/IBM Personal Computer/Atari 400/800 Computers/Osborne/TRS 80 Model III/Commodore VIC 64. The actual budget capacities will vary with each computer.

Continental Software
A Division of Arrays, Inc.

It sells the most, because it does the most!

Continental Software Co., 11223 South Hindry Avenue, Los Angeles, California 90045 Telephone (213) 417-3003 • (213) 417-8031

Publisher/Editor-In-Chief	Robert C. Lock
Publisher's Assistant	Alice S. Wolfe
Senior Editor	Richard Mansfield
Managing Editor	Kathleen E. Martinek
Features Editor	Tom R. Halfhill
Technical Editor	Ottis R. Cowper
Editorial Assistant	Charles Brannon
Programming Assistant	Patrick Parrish
Administrative Assistant	Vicki Jennings
Copy Assistants	Juanita Lewis Mary Parker
Associate Editors	Jim Butterfield, Toronto, Canada Harvey Herman, Greensboro, NC Fred D'Ignazio, c/o COMPUTE!, P.O. Box 5406 Greensboro, NC 27403 David Thornburg P.O. Box 1317, Los Altos, CA 94022
Contributing Editors	Marvin DeJong Bill Wilkinson Gene Zumchak
Art Director/ Production Manager	Georgia Papadopoulos
Assistant	Irma Swain
Artists	De Potter Jean Hendrix
Typesetting	Terry Cash
Illustrator	Harry Blair
Production Assistant	Dai Rees
Associate Publisher/ National Advertising Sales Manager	Andy Meehan
Advertising Coordinator	Alice S. Wolfe
Advertising Accounts	Bonnie Valentino
Sales Assistant	Rosemarie Davis
Operations/Customer Service Manager	Carol Lock
Coordinator	Fran Lyons
Assistants	Christine Gordon Dorothy Bogan Gail Jones Chris Patty Patty Jones
Shipping & Receiving	Jim Coward Larry O'Connor
Accounting Manager	W. Jerry Day
Bookkeeper	Ellen Day
Accounting Assistant	Linda Roquemore
Assistants	Doris Hall Ruth Granger Anna Harris

Small System Services, Inc. publishes:

COMPUTE! The Journal For Progressive Computing

COMPUTE! Books

Corporate office:
625 Fulton Street,
Greensboro, NC 27403 USA

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

Telephone: 919-275-9809

Robert C. Lock, President
W. Jerry Day, Vice-President and Comptroller
Kathleen E. Martinek, Assistant To The President
Sonja Whitesell, Executive Assistant

Coming In February

Special Games Issue:

How The Professionals Program Games

Four Exciting Games: *Mastermaze, Slalom, Copycat, And A Day At The Races*

VIC And PET High Resolution Plotters

Atari SuperFont Plus And Left-handed Joysticks

Apple High Resolution Painter

Color Computer Relocation Calculator

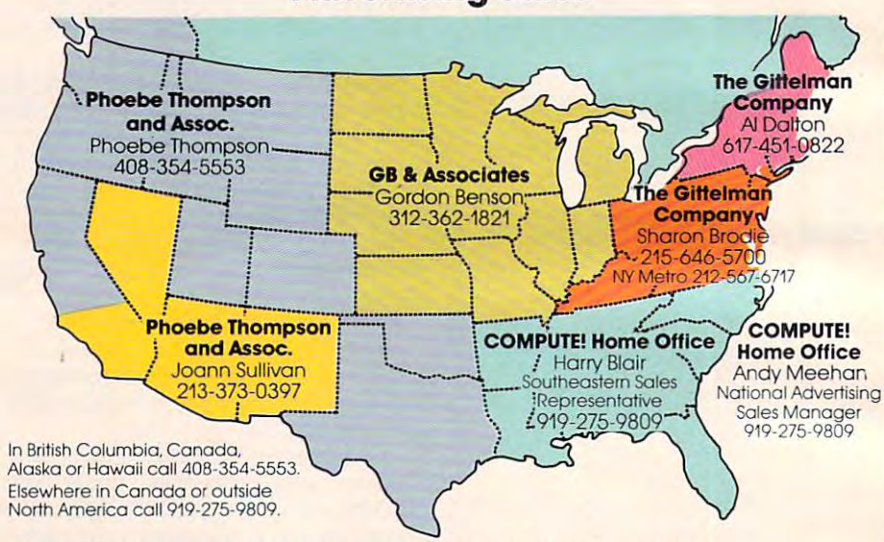
Subscription Information
COMPUTE! Circulation Dept.
P.O. Box 5406
Greensboro, NC 27403

TOLL FREE
Subscription Order Line
800-334-0868
In NC 919-275-9809

COMPUTE! Subscription Rates (12 Issue Year):

US	(one yr.)	$20
	(two yrs.)	$36
	(three yrs.)	$54
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

Advertising Sales



In British Columbia, Canada, Alaska or Hawaii call 408-354-5553.
Elsewhere in Canada or outside North America call 919-275-9809.

Phoebe Thompson and Associates
101 Church Street
Suite 13
Los Gatos, CA 95030
PHOEBE THOMPSON

Phoebe Thompson and Associates
2556 Via Tejon
Palos Verdes Estates, CA 90274
JOANN SULLIVAN

GB & Associates
P.O. Box 335
Libertyville, IL 60048
GORDON BENSON

COMPUTE! Home Office
625 Fulton Street
Greensboro, NC 27403
HARRY BLAIR
Southeastern Sales Representative

The Gittelman Company
Statler Office Building
Suite 518
20 Providence Street
Boston, MA 02116
AL DALTON

The Gittelman Company
Summit Office Center
7266 Summit Avenue
Fort Washington, PA 19034
SHARON BRODIE

Address all advertising materials to:
Alice S. Wolfe
Advertising Production Coordinator
COMPUTE! Magazine
625 Fulton Street
Greensboro, NC 27403 USA

AN INFORMATION MANAGEMENT SYSTEM FOR YOUR COMMODORE COMPUTER

InfoPro is a menu driven and interactive "information management" system for the Commodore 8032 computer. **InfoPro** uses "friendly" screen prompts that "guide" you from function to function. This makes **InfoPro** unusually easy to learn and just as easy to operate.

For Mailing List applications **InfoPro** can print up to 8 labels across and even has a built in "structure" with fields already pre-set. This structure can easily be changed to fit many other types of office jobs.

Another extremely powerful feature of **InfoPro** is Super Scan. The Super Scan feature acts like an "electronic filing cabinet" and provides the user with almost instantaneous access to the data stored in a file. The powerful Report Generator allows you to "select" information for printing based on up to 5 different parameters or criteria and to perform various math functions.

Another powerful and indispensable feature is **InfoPro**'s ability to interact with the WordPro family of word processing programs. This provides the user with a "link" from the area of data information management to the area of word processing, allowing the user to manipulate, sort, and select data by certain criteria, which can then be inserted into "personalized" letters, documents, overdue notices, etc. **InfoPro** will also allow you to ADD, DELETE or CHANGE your information "fields" any time you wish. This means that as your business changes, **InfoPro** has the flexibility to change with it.

As with all Professional Software products, **InfoPro** comes complete with a professionally written and fully-tested user oriented manual. **InfoPro** also includes a program ROM, and **InfoPro** System Diskette.

Start managing your information today.

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

Professional Software Inc.

51 Fremont Street
Needham, MA 02194
Tel: (617) 444-5224
Telex: 951579

WordPro™ and InfoPro™ are registered trademarks of Professional Software

ASK THE READERS

The Editors and Readers of COMPUTE!

High Vs. Low Resolution

Could you explain the difference between hi-res and low-res graphics?

Mike Porter

The essential difference is that when a computer does not have the high resolution option, you are limited to a set of built-in graphics characters. These characters, like the letters of the alphabet, will be crisp and clear, but you cannot create your own special characters.

A great deal can be accomplished, however, by combining the different symbols of a built-in set. Cubes, stairs, and many other pictures can be created. It's like having perhaps 128 different shapes of paper. You can put them together in thousands of ways, but you can't customize them individually by cutting them with scissors.

High resolution, on the other hand, permits you to control the individual pixels (dots) anywhere on the TV screen. This means that you can create detailed figures of your own design – perhaps the Greek alphabet or an image of a starship – and that curved lines will look more like true curves. High resolution generally adds to the price of a computer, but does provide more flexible graphics, more visual possibilities. Alternatively, it is usually possible to add an optional high resolution capability to computers which do not offer it as a standard feature.

Typing Programs From COMPUTE!

I have seen several programs in **COMPUTE!** that have a "^" symbol in them and there is no such character on my keyboard. Did I miss it in the instruction manual (I triple-checked)? This upside-down V has stumped me. Also, what is the "–" symbol for? I know that an underline means to type the shifted version of whatever character is underlined, but what do you do when "–" stands alone?

Jim Lockridge

The "^" symbol indicates an "up-arrow" symbol (↑) on Commodore computers and represents "to the power of" something. Whenever you see it, type the key with the arrow pointing upwards with respect to the keyboard. Hold down the SHIFT key and hit the SPACE BAR when you see an underlined blank. See the "COMPUTE!'s Listing Conventions" page in each issue.

Atari 400: Can You Add Memory?

I am getting a computer in a few months, but I have a problem. Can the Atari 400 be expanded to hold more memory than it comes with? I constantly see ads for RAM expansion boards, but the ads for the 400 computer itself say "16K RAM (non-expandable)." Well, which is it?

Scott Bonder

Officially, the Atari 400 cannot be internally expanded. However, you can replace the 16K memory board inside your computer with a 32 or 48K board. Atari does not manufacture them, but several third party vendors do. There is even a 64K RAM board sold. Be aware, however, that opening your Atari 400 to replace the board might void your warranty.

How Can Chips Address 128K?

I have two questions, one of which has been bothering me for some time. How can an 8-bit chip such as the new MOS 6509 and 6510 address more than 64K? Commodore's new P128 computer will have 128K RAM. How is this possible? Although the Commodore 64 has 64K RAM, you have stated that only 38K (52K for M.L.) is available for programming. Does this mean that software that had been embedded in ROM in previous CBM machines must now be soft-loaded upon power-up, or is it contained in a plug-in ROM cartridge?

My second question is if programs written for the Commodore 64 will run on the P128, and vice versa. I am particularly interested in the compatibility of the various plug-in cartridges that will become available.

Ron Dagostino

The 6510 chip will not address more than 64K of memory. The 6509 can, however, through a technique known as "bank switching," whereby large blocks of memory (banks) *may* have the same addresses. *The microprocessor must then select which to use from among the banks. Details on exactly how the 6509 makes this selection are not yet available.*

In any case, the 64K limit on memory addressing is not because the 6500 family of chips are eight-bit processors. Rather, it is a result of the chips having 16 address lines ($2^{16} = 65536$). For example, the 8086

LAST NIGHT WE EXCHANGED LETTERS WITH MOM, THEN HAD A PARTY FOR ELEVEN PEOPLE IN NINE DIFFERENT STATES AND ONLY HAD TO WASH ONE GLASS...

That's CompuServe, The Personal Communications Network For Every Computer Owner

And it doesn't matter what kind of computer you own. You'll use CompuServe's Electronic Mail system (we call it Email™) to compose, edit and send letters to friends or business associates. The system delivers any number of messages to other users anywhere in North America.

CompuServe's multi-channel CB simulator brings distant friends together and gets new friendships started. You can even use a scrambler if you have a secret you don't want to share. Special interest groups meet regularly to trade information on hardware, software and hobbies from photography to cooking and you can sell, swap and post personal notices on the bulletin board.

There's all this and much more on the CompuServe Information Service. All you need is a computer, a modem, and CompuServe. CompuServe connects with almost any type or brand of personal computer or terminal and many communicating word processors. To receive an illustrated guide to CompuServe and learn how you can subscribe, contact or call:

CompuServe

Information Service Division, P.O. Box 20212
5000 Arlington Centre Blvd., Columbus, OH 43220

800-848-8990
In Ohio call 614-457-8650

An H&R Block Company

Inside Apple

Apple Computer Inc., 20525 Mariani Avenue, Cupertino, California 95014 Vol. 1 No. 2

For the authorized Apple dealer nearest you, call 800-538-9696 (800-662-9238 in California.)

Fruitful Connections.

There are more people in more places making more accessories and peripherals for Apples than for any other personal computer in the world.

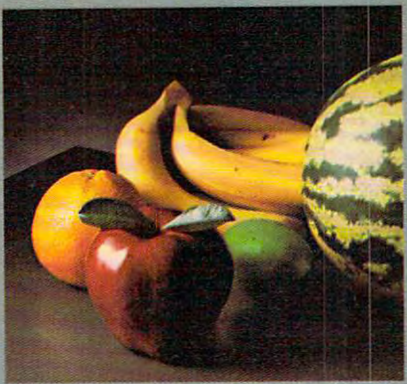
Thanks to those people — in hundreds of independent companies — you can make the humblest 1978 Apple II turn tricks that are still on IBM's Wish List for 1984.

But now we're coming out with our very own line of peripherals and accessories for Apple® Personal Computers.

For two very good reasons.

First, compatibility. We've created a totally kluge-free family of products designed to take full advantage of all the advantages built into every Apple.

Second, service and support.



Now the same kindly dealer who keeps your Apple PC in the pink can do the same competent job for your Apple hard-disk and your Apple daisywheel printer.

So if you're looking to expand the capabilities of your Apple II or III, remember:

Now you can add Apples to Apples.

A joy to behold.

The new Apple Joystick II is the ultimate hand control device for the Apple II.

Why is it such a joy to use?

With two firing buttons, it's the first ambidextrous joystick — just as comfortable for lefties as righties.

Of course, it gives you 360° cursor control (not just 8-way like some game-oriented devices) and full X/Y coordinate control.

And the Joystick II contains high-quality components and switches tested to over 1,000,000 life cycles.

Which makes it a thing of beauty. And a joystick forever.

Gutenberg would be proud.

Old Faithful Silentype® has now been joined by New Faithfuls, the Apple Dot Matrix Printer and the Apple Letter Quality Printer.

So now, whatever your budget and your needs, you can hook your Apple to a printer that's specifically designed to take advantage of all the features built into your Apple. With no compromises.

The 7x9 Apple Dot Matrix Printer is redefining "correspondence quality" with exceptional legibility. With 144x160 dots per square inch, it can also create high resolution graphics.

The Apple Letter Quality Printer, which gets the words out about 33% faster than other daisywheel printers in its price range, also offers graphics capabilities. See your authorized Apple dealer for more information and demonstrations. Because, unfortunately, all the news fit to print simply doesn't fit.

Up the creek without a paddle?

Or lost in space? Or down in the dungeons?

Whatever your games, you'll be happy to know that someone has finally come out with game paddles built to hold up under blistering fire. Without giving you blisters.

Apple Hand Controller II game paddles were designed with one recent discovery in mind:

People playing games get excited and can squeeze very, very hard.

So we made the cases extra rugged. We used switches tested to 3,000,000 life cycles. We shaped them for holding hands and placed the firing button on the right rear side for maximum comfort.

So you'll never miss a shot.

A storehouse of knowledge.

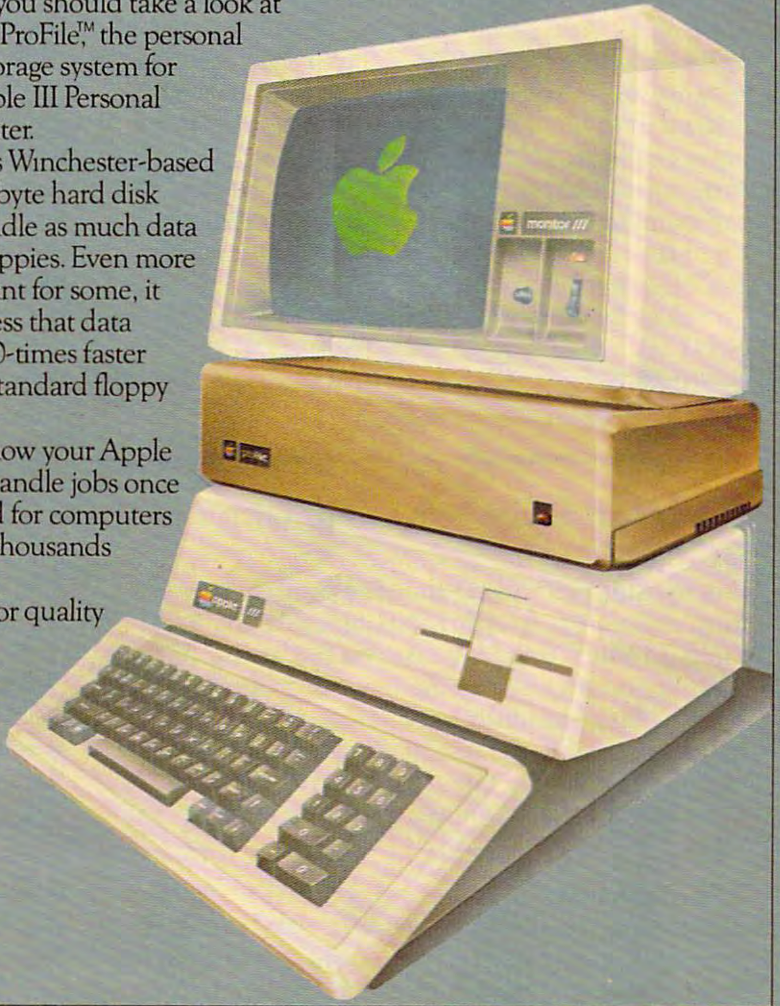
If you work with so much data or so many programs that you find yourself shuffling diskettes constantly, you should take a look at Apple's ProFile,™ the personal mass storage system for the Apple III Personal Computer.

This Winchester-based 5-megabyte hard disk can handle as much data as 35 floppies. Even more important for some, it can access that data about 10-times faster than a standard floppy drive.

So now your Apple III can handle jobs once reserved for computers costing thousands more.

As for quality and reliability, you need only store one word of wisdom:

Apple.



Launching pad for numeric data.

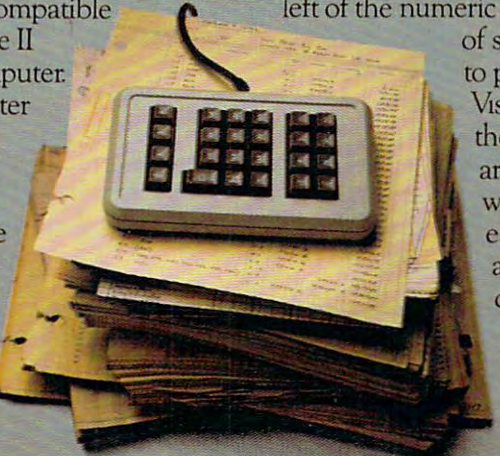
Good tidings for crunchers of numerous numbers:

Apple now offers a numeric keypad that's electronically and aesthetically compatible with the Apple II Personal Computer. So you can enter numeric data faster than ever before.

The Apple Numeric Keypad II has a standard calculator-style layout. Appropriate, because unlike some other keypads, it can actually function as a calculator.

The four function keys to the left of the numeric pad should be of special interest to people who use VisiCalc.® Because they let you zip around your work sheet more easily than ever, adding and deleting entries.

With one hand tied behind your back.



VisiCalc is a registered trademark of VisiCorp, Inc.

microprocessor used in the IBM Personal Computer provides a 20-bit address which can directly address one megabyte of memory ($2^{20} = 1048576$).

The Commodore 64 has 64K of memory. BASIC ROM takes up part of this space, and quite a bit of RAM is used by the computer for pointers, screen memory, sprites, etc. This is why only 38K is normally available to the user. (See **COMPUTE!**, *October 1982, for a 64 memory map.) You can bank-switch the 64 by POKEing address 1. POKE 1,6, for example, makes BASIC go away and the RAM "behind" it is available. The bottom three bits of address 1 are memory control bits and direct the computer's attention to the available alternatives. Of course, what takes control of the machine when BASIC is gone is up to you.*

Look for an article on controlling the 64's "hidden" RAM in an upcoming **COMPUTE!**.

Commodore is not yet sure whether the P128 will be software compatible with the 64.

Can Atari Make Tapes For Other Computers?

It is my understanding that the programmable tone generators in Atari also generate the baud rate and tones used for cassette recording. If this is so, it should be possible to save programs in Kansas City Standard or other two tone formats, making possible the use of Atari editing features to prepare cassette tapes for other computers. How can this be actually done?

Steven S. Coles

While the Kansas City standard cassette interface is in use by many computers, it should be noted that it is far from a full standard. Most computers deviate from the original standard one way or another. One nearly universal deviation is to operate at 1200 baud rather than the original 300 baud spec.

The Kansas City standard uses a frequency of 1200 Hz to indicate a "0" bit and a frequency of 2400 Hz to indicate a "1" bit. The frequency change is performed when the waveform crosses the zero voltage level. The actual data transfer is usually done through as USART, but it is quite possible to simulate this in software. At this point we run into a brick wall: just about everybody saves the data on the cassette in their own way, so you will have to find out what method is used for the computer you wish to adapt to. If at this point you are still *interested in giving it a go, I suggest you go down to your local library and drag out the April 1977 issue of BYTE. On page 40 you will find an excellent article by Carl Helmers on how to do it.*

VIC Custom Characters

I recently added 8K to my VIC-20. This causes new locations in RAM to be assigned to the start of BASIC, the screen area, and color control area.

There is a technique for using custom characters on the 5K VIC-20 that involves moving down the end of the BASIC RAM working area from page 30 to page 28 and loading pages 29 and 30 with the custom characters. Every custom character program that I've seen uses this technique – POKE 56,28:CLR.

Both the screen RAM and custom character area must be located below page 30 in RAM to work. Unfortunately, with the 8K expansion (and without 3K) there is no space available below page 30. The obvious thing to do is to relocate the start of BASIC text from page 18 to page 30, since with 8K the top of RAM is page 64. This appears possible by changing locations 44, 46, 48, and 50 from 18 to 30. Unfortunately, it doesn't work. The RUN command results in a Syntax Error and GO TO yields Syntax Error in statement 0. (LISTS do work though.)

This wordy prelude leads to my question: Is it possible to have custom characters (and alternate screens) with the 8K expansion (and without the 3K)? If yes, how?

Dick Gough

Several readers have inquired about this. Unfortunately, there is no known solution at this time. Several programmers are currently working on this problem and we'll publish the answer as soon as it is solved.

A Time-saving Tip

I'd like to share a trick I learned from the programmers who did our business software.

Make the first line in your program a REMark statement containing your SAVE command. For example:

```
10 REM SAVE 6, "INVENTORY",D80
```

or

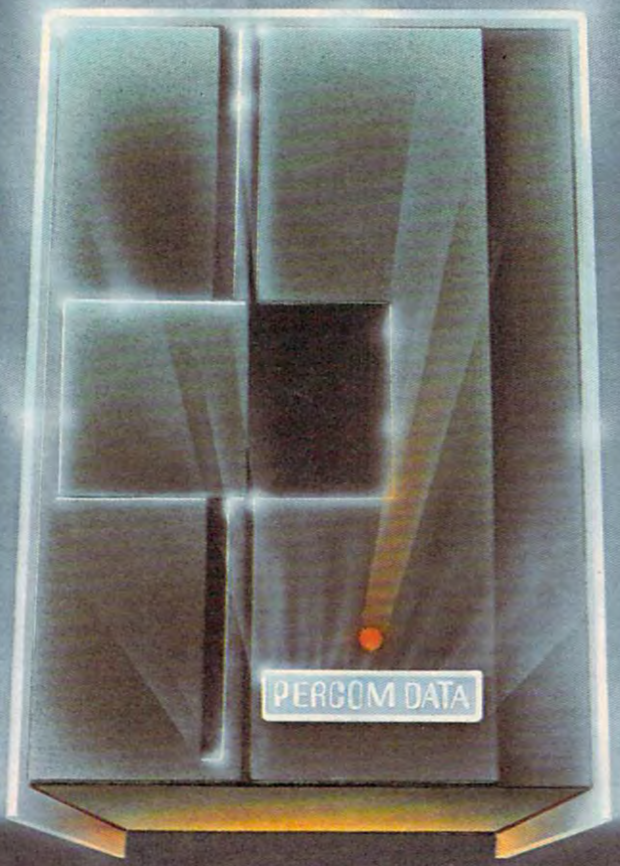
```
1 REM SAVE "INVENTORY",1.1
```

Then, whenever you make a change you don't have to remember where and how to save your program. Just LIST the first line, blank out everything before the SAVE command, and execute. It works equally well with tape or diskette.

Linda Johnson

Atari's Right Cartridge

Aside from "Monkey Wrench" [a programmer's aid package from Eastern House Software], does anyone know of any cartridge that goes into the right slot on the Atari 800? Does Atari have any



We've Got More Than A Fond Attachment For Your ATARI

We've Got A Disk Drive For $488.

Percom Data Corporation believes your Atari* home computer is more than just fun and games. We believe you should be able to get a single-density, floppy-disk-system for your Atari 400 or 800 at a price that will take you into the future without knocking you into the next galaxy.

Percom Data has been manufacturing disk-drive systems, and other accessories for personal computers since the mid-1970's and is the industry standard to follow when it comes to data separation and system compatibility.

The Percom Data AT-88 combines Percom Data quality and reliability at a price that is not a budget-buster.

The Percom Data AT-88 offers 88 Kbytes (formatted) in single-density, with plug-in ease of attachment to your Atari. The AT-88 has integral power supply, "no-patch" to Atari DOS and critical constant speed regulation.

Take advantage of this low introductory price of $488 by calling Percom Data now to get more information, or the name of an authorized dealer nearby. Call toll-free
1-800-527-1222

PERCOM DATA
CORPORATION

Expanding Your Peripheral Vision

DRIVES • NETWORKS • SOFTWARE

11220 Pagemill Road Dallas, Texas 75243 (214) 340-7081
1-800-527-1222

*Atari 400/800 is a trademark of ATARI, INC.

plans to ever utilize this slot, or was it an idea that didn't pan out?

Bill Lukeroth

The problem with the right slot is that the Atari 400 doesn't have one. Atari has evidently abandoned applications for the right slot since such an application could not be used on the 400. Atari is committed to supporting both machines equally. Also, because a full 16K can be put on the left cartridge, it alone is enough space for most applications.

Commodore Time

I have a VIC-20. On several occasions I have attempted to tally the amount of computer time utilized via the TI and TI$ functions.

Could you explain first the purpose of TI? Can it be modified manually – set to zero? What does a printout of 1429292 mean? What is its relationship to realtime and TI$?

Secondly, is TI$ supported to count realtime? If so, can you explain why 6 hrs. 31 mins. would show up on TI$ as 063639?

Lastly, is there any way to maintain a cumulative tally on the system of aggregate "on-time"?

D. L. Branam

You can tell Commodore computer's special time variable, TI$, what time it is (or set it to zero so it can keep track of how long the system has been on) by treating it like an ordinary string variable. For example:

TI$ = "033000"

would let the computer know that it was 3:30. This can be done either from within a program or by just typing it in from the keyboard in "direct mode." The string is arranged "HHMMSS" for the positions of hours, minutes, and seconds. It will take anything up to 240000 hours and must include all six numbers, even if a zero is in the first position as in the example above.

You can print out the time in a variety of ways. Here's one:

?LEFT$ (TI$,2) "/" MID$ (TI$,3,2) "/" RIGHT$ (TI$,2

or just:

? TI$

TI is the numeric clock variable. It is set to zero when power is first turned on or when you reset the clock: TI$ = "000000". The number in TI is counting time in 1/60ths of a second. To see TI:

10 ?"[HOME]" TI: GOTO 10

Six hours 31 minutes should be 063100 when TI$ is printed out.

Atari Memory Expansion Problems?

I own an Atari 800 with 16K memory and am careful about the quality of products I buy for my computer. There seems to be a rumor going around about the 32K RAM memory board made to fit the 800 model. I've heard that adding this board can cause errors to occur in the computer's performance. Please tell me if it's true and, if so, how or why it happens.

Allen Levy

We have heard of no problems related to the use of a 32K board on an Atari 800. These expansion boards are not manufactured by Atari, but rather by third party vendors. The requirements of an Atari board are fairly specific: they must not use too much power and they have to be fast enough (200 ns. or better). We haven't heard, though, of problems relating to expanding memory with these products.

INPUT That Puts Anything In

Here's an interesting Commodore input routine that I'd like to share with you. Ever notice that when you INPUT a string which has a comma or colon that the computer only takes in the part before the punctuation and then prints EXTRA IGNORED?

Here's how to get around it. Say you want to INPUT C$:

```
10 GOSUB1000:C$=B$
20 PRINTB$:END
1000 B$=""
1010 GETA$:IFA$=""THEN1010
1020 PRINTA$;
1030 IFA$=CHR$(13)THEN RETURN: REM 13 MEANS THE
     RETURN KEY WAS TYPED
1040 B$=B$+A$
1050 GOTO 1010
```

When you run this, you don't get the normal question mark. Now you can put in anything you want, but don't use A$ or B$ any place except in this subroutine. Whenever you want to put in a string, GOSUB to 1000 and, when you come back with RETURN, just let the string you're looking for (C$ in this example) be equal to B$.

George Trepal

This INPUT routine is great for people who will want to use a computer program, but don't know about avoiding commas, etc. Whatever they type, the program will take it in without stopping and going to an error message and then saying READY. It's also possible to use delete and insert to correct errors (but these "characters" will be included in the final string).

To prove it, run this and type TEST the first time. Then, when the program ends, type: ?LEN(B$) to see how long B$ is. You will get four as the answer. Now run it again and type TESX and then use the delete key to change the X back to a T. ?LEN(B$) will now give you a six because B$ still contains the X and a character for a delete. You don't notice these extra characters, though, because when B$ is PRINTed, it puts the X on the screen and then deletes X, replacing it with T. It's too quick to see. ©

HOW TO MAKE YOUR VIC-20 AWESOME

Right out of the box, your VIC-20® from Commodore is one great little computer. And it gets even better with DATA 20's easy to buy, easy to install, and easy to use enhancements. Here are four ways you can get new power and more sophisticated capabilities from your VIC...

VIDEO PAK gives you a computer that outperforms systems costing twice as much. Plug our cartridge into your expansion port, and your display instantly goes to the industry-standard 24 lines, with a choice of 40 or 80 upper and lower-case characters. At the same time, you up your memory from 5K to 20K or 70K, so you can handle more sophisticated functions—including most 8032 software. Our package also includes a terminal emulator and a screen print feature. VIDEO PAK is a must for word processing—and your key to increased performance on everything from games to spread sheets. Suggested retail is just $299.95 including 16K—or $399.95 for 64K.

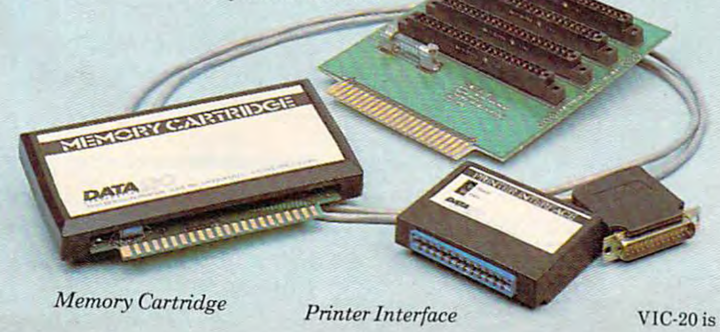
PRINTER INTERFACE is simple, yet sophisticated. Flexibility, continuous visual monitoring of the data transfer functions, and easy installation make this an exceptionally smart buy! A glance at the status lights, and you know if the printer is hooked up, if the data buffer is full, and if data is being transmitted. Easily configured DIP switches match your VIC-20 to most popular printers. The DATA 20 interface comes with cable and connector, needs no assembly, and virtually troubleshoots installation for you. Suggested retail: $69.95

EXPANSION CHASSIS lets you use 4 cartridges at once. Run a series of compatible memory, software, or game cartridges of any make. Just pop in any cartridge with the standard 22-pin edge connector. And don't worry about your VIC-20 power supply—our chassis protects it with a built-in 500ma fuse. Suggested retail: $64.95

MEMORY CARTRIDGE boosts your brainpower to 20K. Here's an ideal first add-on for your VIC-20. And when DATA 20 gives you more memory, you can forget about headaches. Our cartridge is housed in a rugged plastic case and features 200ns RAM's for reliability. Suggested retail: $99.95

Check out our AWESOME peripherals. Ask your computer dealer for a first-hand look at our extensive capabilities, high quality, and very reasonable prices. Or send $3.50 for a copy of our current catalog on VIC-20 compatibles.

DATA 20 CORPORATION
23011 Moulton Parkway, Suite B10, Laguna Hills, California 92653.



Expansion Chassis

Memory Cartridge

Printer Interface

DATA 20 CORPORATION

Price/Performance Peripherals

Questions Beginners Ask

Tom R. Halfhill, Features Editor

Are you thinking about buying a computer for the first time, but don't know anything about computers? Or maybe you just purchased a computer and are still baffled by what personal computing is all about. Starting this month, **COMPUTE!** *will tackle some questions which we are most frequently asked by beginners.*

Q: Which is the best computer to buy?

A: The best one for your needs.

Seriously, we're not trying to duck the question. People ask us this all the time, in letters, telephone calls, and at computer shows we attend. We get the feeling they are never really satisfied with our answers, since what they really want to hear is something like, "Buy the Atapple ZX-20, it's definitely the best one." Unfortunately, we cannot give such an answer. For one thing, since **COMPUTE!** covers many machines, the magazine must maintain objectivity. But more importantly, there is no one right answer. All the computers have their own strengths and weaknesses, and all computer buyers have – or should have – their own ideas of what they need in a computer. We think nearly anybody who buys one of the major brands with a clear idea of his or her needs will be satisified with the purchase.

The key is to identify your needs and desires. If game-playing will be a major use of your computer, then color graphics and sound will be important features. Someone primarily interested in word processing may well have no need for either feature.

If you've looked hard and long at the various computers in a certain price range and still can't decide between them, then perhaps the differences are too slight to matter anyway. Or maybe you should base your decision not on the hardware, but on the available software. If the computer will be used primarily for educational purposes, and you're attracted by a particular line of educational programs, you may lean toward the computer that those programs are designed to work on. The programs may not be compatible with or available for another machine.

If you still think we are sidestepping the whole question, then consider this: If one brand of computer were clearly superior, and if we at **COMPUTE!** were in a position to know about it, then it stands to reason that all of our editors would own that computer. But in fact, both at work and at home, we own and use many different computers. 'Nuff said?

Q: What are PEEK and POKE?

A: PEEK and POKE are words (instructions to the computer to do something for you) in a computer programming language known as BASIC (Beginner's All-purpose Symbolic Instruction Code). BASIC is the standard language on home/personal computers. PEEK and POKE allow you, as a programmer, to work directly with the computer's memory.

PEEK allows you to examine the contents of a single memory location (known as a "byte"). Each memory location in a computer has a numbered address, sort of like houses in a city. In turn, each memory location *stores* a number which usually has something to do with the operation of the computer or a computer program. If you type PRINT PEEK (8502), the computer will PRINT on the screen the number stored in that address. Therefore, PEEK is often used in programs to determine if a certain number is stored at a particular location, usually as a prelude to changing the number to achieve some desired result.

POKE is the word that allows you to make those changes, to change numbers stored in locations in Random Access Memory (RAM) – that part of a computer's "user memory" which can be changed by the programmer. For example, if you type POKE 82,0, the number 0 will be stored at memory location 82. POKEs can often change some facet of the computer's behavior. Since each model's memory is arranged differently, PEEKs and POKEs will not achieve the same results on different computers (in the above example, POKE 82,0 will make the left screen margin zero on an Atari).

Q: What is a CONTROL key?

A: A CONTROL key (often abbreviated CTRL) is

OUR GAMES ARE WORLDS APART FROM THE ORDINARY.

Games for the Apple and Atari computers.

Gamesters everywhere are discovering the many worlds of Brøderbund's Apple and Atari programs, and they're not going back to anything less. Because Brøderbund's entertainment software offers pleasures and perils found nowhere else in the galaxy.

Only Brøderbund can take you on a desperate rescue mission aboard CHOPLIFTER, then give you a fast lesson in train robbing with TRACK ATTACK. Next, try battling the metamorphic maze and slithery monsters found only in SERPENTINE and LABYRINTH. Or enjoy the lively pinball action of DAVID'S MIDNIGHT MAGIC. Trouble shooters will find plenty of targets with STAR BLAZER, while DUELING DIGITS will put your mind and reflexes to their ultimate test.

So next time you're looking for new worlds to conquer, remember Brøderbund. You'll find the experience most extraordinary.

Call or write for our free catalog.

Brøderbund Software

Brøderbund products are available at your retailer or by writing to:
Brøderbund Software, Inc., 1938 Fourth Street, San Rafael, CA 94901, Tel: (415) 456-6424

a special key found on many computer keyboards. In effect, it works something like a SHIFT key. Just as a SHIFT key adds a function to a regular key – i.e., changes a lowercase letter to uppercase, or changes the "4" key to a dollar sign – the CONTROL key also is used in combination with another key to select an additional function or symbol.

These functions and symbols vary among different models of computers. For example, holding down the CONTROL and "C" keys on an Apple II will usually stop (or "break") a BASIC program which is running. CONTROL-C on an Atari will print on the screen a small graphics character resembling the lower right corner of a box. On a Commodore VIC-20, CONTROL-C has no effect; another special key is used instead to print graphics symbols. The manual which comes with every computer explains the functions of its special keys. ©

Computers And Society

David D. Thornburg, Associate Editor

The Computer As A Tool For Discovery

The notion that the development of low-cost computers is "revolutionary" is not a new one, but the word revolutionary is used so much that one is likely to dismiss it as pure advertising hype along with words like "new" and "improved." And yet those of us who have been involved with this industry since its inception are aware that the development of the personal computer is not, by itself, revolutionary just because it may bring computer technology into people's homes.

"Revolutionary" is a special word – it implies that a technology or tool causes far-reaching changes in many aspects of our lives.

The development of the steam engine was revolutionary; the development of steam cleaning for carpets was not. The development of the telephone was revolutionary; the development of the answering machine was not. The development of the airplane was revolutionary; the development of in-flight entertainment was not.

Computer technology has had an impact that reaches far beyond the world of the computer itself. Computer users in industry and academia have known this for many years. Now that the power of the computer has reached the home, can we expect that people will start thinking about their world differently?

I think so.

The computer will help people to explore ideas that they wouldn't begin to explore if the computer hadn't given them the leverage to start thinking about them.

Beauty And Practicality

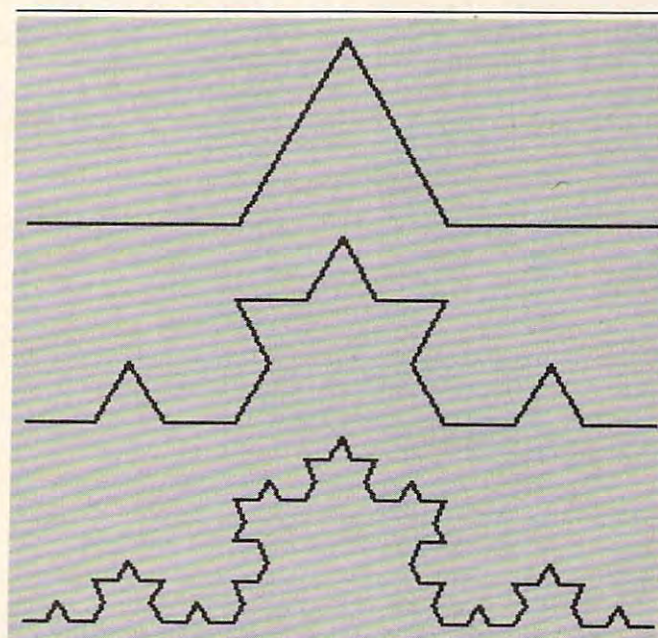
As an example of this, let's explore the development of a new field of mathematics called "fractal geometry." I have touched on this branch of mathematics in the "Friends of the Turtle" column a few times. I am intrigued by it because it deals with topics of considerable beauty and practical interest. Its seeds were planted a hundred years ago, but it was only after the development of the computer that anyone was able to begin to advance this field beyond the crudest level.

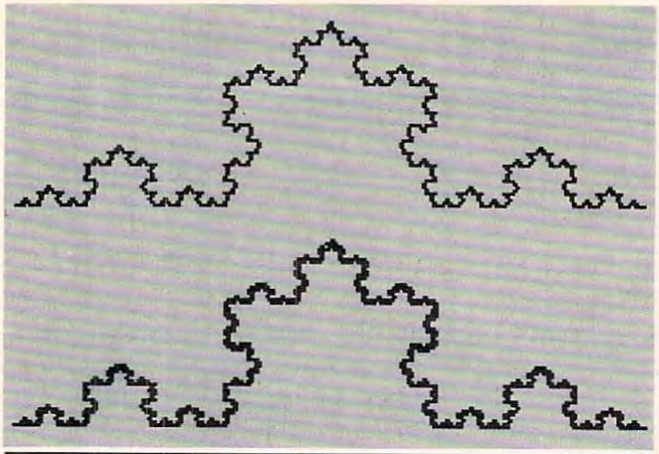
I realize the risk of illustrating a computer application based on mathemetics, since it tends to reinforce the erroneous concept that computers are primarily mathematicians' tools. The only reason for pursuing this example is because it is an interesting story in its own right.

In the late 1800s mathematicians were exploring some questions that went to the very foundations of geometry. One question of interest was if one could construct a curve that would fill a plane. At first thought, the idea of filling a two-dimensional surface with a curve made from a one-dimensional line is as absurd as asking for a roll of optically flat steel, or asking how many angels can dance on the head of a pin.

To the Italian mathematician Guiseppe Peano, this was a most intriguing question. In 1890 he published a proof that space-filling curves were, in fact, possible – that one could construct a curve that has the dimension of a surface. While this proof attracted the attention of several other mathematicians, the bulk of the academic community abhorred the thought of such "ill-behaved" curves.

In 1904 Helge von Koch continued the pursuit of strange types of functions by publishing the discovery of the "snowflake" curve. This curve is created by preparing successive generations from a simple motif. The rule to be followed is that each new generation is made by replacing each straight line in the previous generation with a copy of the motif itself.





If this process is carried on to infinity, one gets a very strange curve indeed. First, the curve is everywhere bumpy – there are no smooth regions. Second, even though the curve has clearly defined boundaries, it has infinite length. Third, the curve has a "dimension" that is intermediate between that of a line and a surface. To mathematicians of the early twentieth century, this curve was monstrous. To the contemporary mathematician Benoit Mandelbrot, it represented the need for a new field of mathematics, to be called fractal geometry.

The history and development of this field is beautifully illustrated in Mandelbrot's new book, *The Fractal Geometry of Nature* (W. H. Freeman, San Francisco). Through the pages of this richly illustrated volume, the reader is treated to a new way of thinking about geometry and nature.

For example, if you want to model a coastline, you are far better off to use a fractal curve than a smooth approximation, simply because coastlines are not smooth. Coastal lengths depend on the ability of the measuring stick to follow the nooks and crannies along the way. A coarse measuring stick gives a result corresponding to an early generation of a fractal curve. As the length of the measuring stick gets smaller, the total measured length of a coastline grows ever larger. This is also true for fractal curves.

Where does the computer fit in all of this? The notion of defining a curve in terms of itself may challenge the imagination, but it has a simple implementation in computer programming called recursion. Furthermore, the speed and accuracy with which computer-driven plotters can graph the various stages of curves free the mathematician to study their properties without being bogged down in drafting.

Computer graphics plays another pivotal role in the practical application of fractal geometry as well, since it is the tool that allows the creation of the simulated landscapes seen in movies such as *Star Trek II*. This practical application of a branch of mathematics would not have been possible were it not for the computer.

Those of you who read "Friends of the Turtle" know that fractal curves can be created on home computer systems using turtle graphics. Their expression in languages such as Logo is quite simple, and Mandelbrot's book provides hundreds of challenges for the interested programmer.

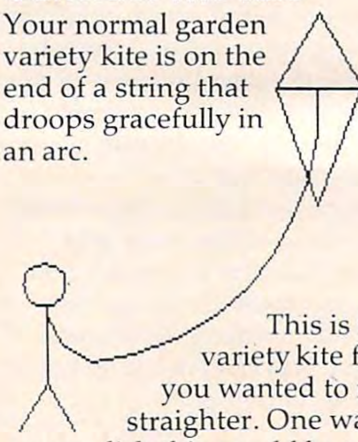
It is important to keep the role of the computer in perspective. The reason that these curves were not explored in depth in the early 1900s is that there was no appropriate tool to aid in their exploration. Now that the computer has made the study of fractals accessible to millions of people, one can expect the field to advance rapidly.

I Call It Kring

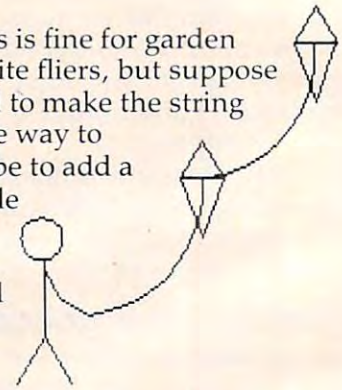
I saw a T-shirt that carried the message: "Recursion is a way of expressing the infinite in finite guise." My friend Sam Savage (the computer scientist/mathematician that invented the jigsaw puzzle called "Shmuzzles") likes to play with the infinite recursively. While I have used Logo to tinker with the latest of his ideas, you may wish to implement them mechanically.

Consider The Kite

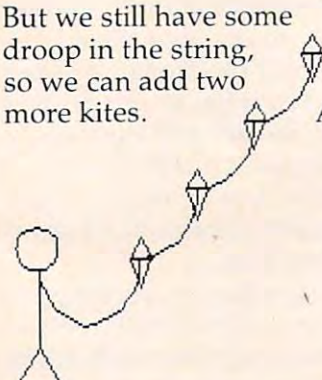
Your normal garden variety kite is on the end of a string that droops gracefully in an arc.



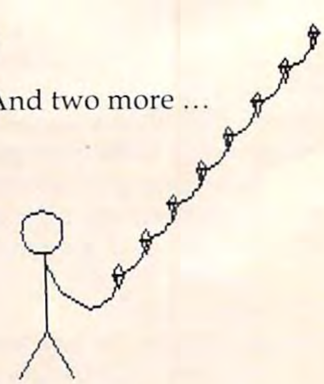
This is fine for garden variety kite fliers, but suppose you wanted to make the string straighter. One way to accomplish this would be to add a second kite in the middle of the string. Because each kite would carry less weight, they would each be smaller.



But we still have some droop in the string, so we can add two more kites.



And two more ...



If you're ready to learn intermediate BASIC programming

STEP BY STEP TWO IS READY FOR YOU!

In Each Parentheses, Choose One:

If you're one of the (happy, smart, ecstatic) thousands who learned (quickly, clearly, non-boringly) to use their APPLE computers with our beginners' BASIC tutorial, *The New Step By Step*, then we (know, believe, swear) that *Step By Step Two* is ready to (ease, breeze, squeeze) you into the (advanced, intermediate, grown-up) world of PEEK and POKE, hexidecimal numbers, concatenations, and (much, much, much) more.

On the other (hand, foot) if you didn't (grow with, thrill to, involve the whole family in) the PDI (learning, hands-on, fun) experience, here's what you missed:

- Teaching techniques that teach
- Involvement you enjoy
- Sounds that spur you on
- Graphics that simplify the complex
- Animation that makes this tutorial a stimulating experience.
- A friendly voice that guides you to course completion

The *Step By Step Two* program works this way:

- the computer program sets up screen displays or sample programs for you.
- the cassette voice tells you what's happening.
- you (deal with, figure out, guess at) the answer.
- the computer (praises, pans) your work.
- you (peruse, plunge into, practice in) the Work Book.
- after each lesson, you're (quizzed, queried, questioned).
- you're then (prepared, practiced, primed) for the next lesson.
- the final exam reveals all (superstars, slackers).

There's lots to learn in *Step By Step Two*:

- PEEK & POKE
- Default values
- Memory map
- CALL program
- CHR$ and ASC functions
- control characters
- ASCII codes
- RAM vs ROM
- hexidecimal numbers
- machine monitor
- string logic
- string arrays
- high resolution graphics
- screen memory

But don't take our (word, words, wordiness) about how (good, great, grand) the *Step By Step* method is. Listen to our (critics, reviewers, friends):

"If you want to learn BASIC or would like a little guidance and encouragement added to what you already know, then the way to go is *Step By Step*." —Softalk

"The *Step By Step* approach is the next best thing to having an experienced programmer by your side... *Step By Step* is a superb example of a successful blend of various media. The teaching principles are sound, the execution is virtually flawless, and the whole thing works." —Popular Computing.

If you want to move ahead in BASIC programming, the next (simple, logical, shrewd) step is *Step By Step Two*.

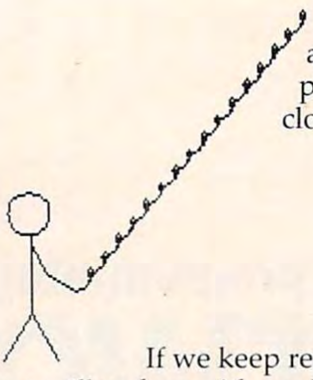
Step By Step Two is available at fine retail stores or direct from PDI for $89.95 plus $3.00 shipping and handling. (The package includes back-up discs.)

P.S. If you've yet to take your <u>first</u> step into BASIC, it's time to get *The New Step By Step*! Same great tutorial techniques, for $79.95 plus $3.00 shipping and handling.

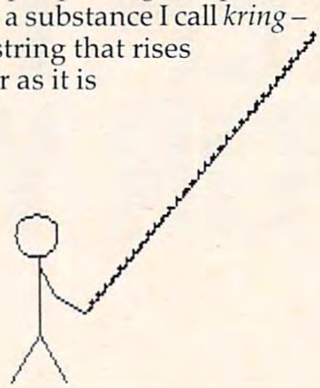
PDI
Program Design, Inc.
11 Idar Court
Greenwich, CT. 06830
203-661-8799

*Apple is a trademark of Apple Computer Corp.

And so on, reducing the area of each kite and placing them closer and closer together.



If we keep repeating this process, we will end up with a substance I call *kring* – a combination kite/string that rises straight up in the air as it is unfurled.

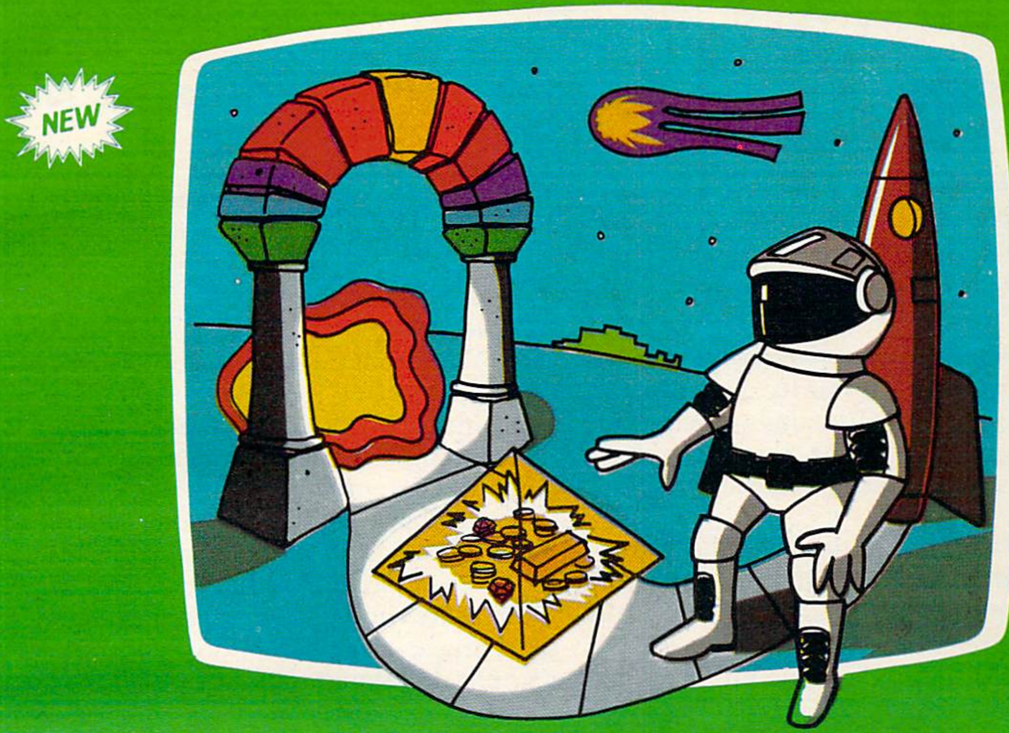


Now that's revolutionary!

©

JOURNEY TO THE PLANETS™

A space, adventure, and arcade action game for your ATARI® 400/800™ personal computer.



What is your favorite type of game; space, arcade, or adventure? "Journey To The Planets" presents an intriguing combination of all three as you find yourself on a strange planet in a strange universe. Luckily, the local gods are friendly and supply you with energy, a spaceship, and weaponry. In turn, you agree to search the universe for treasures for the gods. Board your ship, take off, accelerate through the upper atmosphere and out into space. Your flight should take you past many other inviting planets. With a slow approach and skilled maneuvering, you drop down through the planet's sky to a soft landing on its surface. Disembark and wander through several TV screens full of mystery and excitement. A different adventure awaits you on each planet.

Planetary adventures are designed to exercise your puzzle solving intellect, with arcade action thrown in to enhance the excitement. Although you are given as much time as necessary to solve each adventure, your score, which is based on many factors, favors those who are speedy.

After you capture all the treasures this universe has to offer, return to your adopted planet. Who knows, the gods may be so happy that you can convince them to send you back to your real home. You can't get there without their help!

Available from your local Atari retailer or send $29.95 in check or money order (California residents add 6½% sales tax) to JV Software Inc.

A 32K assembly language program written for your ATARI® 400/800™ computer.

Other products by JV Software include Action Quest and Ghost Encounters, both 16K real time adventure games. Available on cassette or diskette for $29.95.

JV SOFTWARE, INC

3090 MARK AVE. SANTA CLARA, CA 95051

Sound Synthesis

Tom R. Halfhill, Features Editor

Synthesized computer music is a recent development, but inventors have been working on "synthesizers" for decades. Today's home computers and microchips are now starting to open a new world of music and sound for everyone.

Hal Chamberlin, a leading authority on computerized music, remembers the days when adventuresome programmers used transistor radios and even line printers to squeeze music from their early computers.

"People used to tune a little AM radio to an open frequency and hold it next to their spacebars and listen to the sound of [program] loops," recalls Chamberlin, vice president of engineering for Micro Technology Unlimited in Raleigh, North Carolina.

The method worked because pulses flowing through the computer's logic circuits would emit radio frequencies which "leaked" from the computer into the radio's receiver. The programmers on these early IBMs – fiddling around when the boss wasn't looking – soon learned they could play different notes and tones by writing little machine language programs with carefully timed loops.

"They even used to make music by 'playing' the printer," says Chamberlin. "They found out they could control the little hammers in the printhead with a machine language program. So they wrote programs to fire the printhead hammers in a certain pattern to create rhythms.

"Of course," he adds, "it wasn't so great on the printheads."

Such experiments seem crude, even quaint, in this day of computerized music synthesis and home computers with built-in, multiple-voice sound synthesizers on a chip. But these early efforts illustrate that today's "modern" sound devices are really the result of years of research, inventing, and just plain fooling around.

In fact, people have been working on sound synthesizers since the 19th century. And although today's computerized synthesizers seem incredibly advanced in comparison, the leading experimenters in the field believe electronic music is only starting to make itself heard.

Telharmoniums, Theremins, And Rhythmicons

The first music "synthesizer" was built between 1896 and 1906 by American inventor Thaddeus Cahill. He called it a "Telharmonium." The Telharmonium is to modern synthesizers what ENIAC is to modern computers. The Telharmonium weighed more than 200 tons, and moving it to New York from Cahill's lab required several railroad flatcars.

Since the Telharmonium was a pre-electronic instrument, it functioned by means of electric drive motors, pulleys, belts, and gears. Yet it was similar in basic concept to today's synthesizers. It was *polyphonic* (as opposed to *monophonic*), meaning it could play more than one note at a time and thus create chords. It was equipped with a standard music keyboard, but the controls were so complicated that it took two people to play the thing.

The loudspeakers worked mechanically, and the machinery required to generate enough current to drive the speakers was so noisy that part of the Telharmonium had to be housed separately from the listening room. Unfortunately, after ten years of Cahill's work, the Telharmonium was a commercial failure.

For one thing, it was obsolete soon after it was finished. The diode tube was invented in 1904, followed by the triode tube in 1915, which made electronic amplifiers possible. It wasn't long before tube-powered electronic instruments began appearing.

The most successful of these was an instrument invented between 1920 and 1924 by Leon Theremin, originally called an "Etherophone" or "Thereminovox" but now known simply as a "Theremin." This odd instrument was played without being touched – the musician passed his or her hands through the air near two antennas which controlled the pitch and volume. To say the least, this made a Theremin very hard to play, since there were no pre-defined notes like the keys on a piano or the frets on a guitar. Still, Theremins became popular in the late 1920s.

Leon Theremin invented another electronic instrument in 1931 – the "Rhythmicon," the first electronic rhythm instrument. The Rhythmicon was quite sophisticated with features which have appeared on rhythm synthesizers only recently.

A SPEECH SYNTHESIZER ON A DISK!

That's why Apples and Ataris are saying: "Talk Is Cheap"

IT'S CALLED THE SOFTWARE AUTOMATIC MOUTH, S.A.M. FOR SHORT

It's a high quality speech synthesizer created entirely in software. You use it as a software utility, load it into RAM, and then use your machine as usual, except now you can make your programs talk. It generates the speech sounds on demand, so there is no limit to what it can say.

When you hear S.A.M., you'll probably agree that it sounds better than all the hardware speech synthesizers for Apple or Atari computers. And, it has a truly remarkable price.

YOU CONTROL INFLECTION, PITCH AND SPEED

With its user-variable inflection, S.A.M. can accent words on the right syllable and emphasize the important words in a sentence.

You can also make S.A.M.'s speech higher or lower, and faster or slower, over a wide range of settings.

USE EASY PHONETIC INPUT OR PLAIN ENGLISH TEXT

S.A.M. understands a simple phonetic spelling system, not a mysterious alpha-numeric code. S.A.M. helps you learn phonetic spelling by showing you your mistakes, and the owner's manual gets you started with an English-to-phonetics dictionary of 1500 words. So it's easy to make S.A.M. produce exactly the sounds you want.

But suppose you want to type ordinary English, or you want your machine to read a word processor file aloud. The S.A.M. disk comes with RECITER, an English text-to-speech conversion program that lets S.A.M. speak from plain English text.

ADD SPEECH TO YOUR PROGRAMS WITH EASE

In a BASIC program, you add speech with just a couple of commands. In a machine language program, it's just as easy. S.A.M. comes with four demonstration programs to show off its distinctive features and help to write your own talking programs. Write adventure games with talking characters, educational programs that explain aloud, or utilities with spoken prompts – put your imagination to work.

S.A.M.

S.A.M. programmed by Mark Barton

You can order S.A.M. directly from DON'T ASK. Add $2.00 for shipping and handling to your check or money order (or order C.O.D.)

S.A.M. for the Apple II/II+
Includes an 8-bit digital-to-analog converter and audio amplifier on a board. Only **$124.95**
Requires 48K, disk. (S.A.M. takes up to 9K; RECITER 6K.) You will also need a speaker.

S.A.M. for the Atari 400/800
S.A.M. talks through your television speaker. No additional hardware is required. Only **$59.95**
Requires 32K, disk. (S.A.M. takes up 9K; RECITER 6K.) Note: to produce the highest quality speech, S.A.M. automatically blanks the screen during vocal output; the display is preserved. S.A.M. can talk with the screen on, but the speech quality is reduced.

DON'T ASK INC COMPUTER SOFTWARE
2265 Westwood Boulevard, Suite B-150
Los Angeles, California 90064
Telephone: (213) 397-8811

Hear S.A.M. at your favorite computer store today! Dealer inquiries welcome.

THE 40/80 VIDE

O CARTRIDGES

Quantum Data, Inc. (QDI) produces two (2) 40/80 Video cartridges, the 40/80 Video Combo Cartridge with 16K basic user memory, and the 40/80 Video Cartridge which does not contain memory.

The 40/80 Video Cartridge and Video Combo Cartridge is the means to upgrade the VIC-20 computer to a 40 x 24 or an 80 x 24 character display which provides a wealth of new uses for the VIC-20. With the appropriate software, you can now accomplish quality word processing and various business functions that previously were very difficult to achieve with only the VIC's standard 22 character video display.

- *Features a high quality 8 x 8 dot matrix*
- *Character-by-character reverse video attributes allowing adjacent characters to have different attributes.*
- *Black & White composite video (6545 controller does not support color). The Black & White composite video output has the same connector as the VIC video output, 5 pin DIN jack.*
- *Includes two character sets: The ANSI standard 7-bit character set and the Commodore character set. The ASCII character set features all of the standard lower-case and upper-case letters, symbols and numbers. The character set is contained in a standard 2716 EPROM. It is used most often to allow the 40/80 column cartridge to display all of the Block Graphics supported by the VIC-20.*
- *Cannot be used with abbreviated commands nor does it support multiple line wrap around.*
- *Operates in VIC-20 Block Graphic mode.*
- *Features lower-case dot descenders.*
- *All features are accessible through BASIC using POKE commands (screen memory resides at $B800 through $BFFF).*
- *Contains 2K of CMOS internal video RAM; no system RAM is used by the Video Cartridge.*
- *Plugs directly into the VIC-20 memory expansion port or a Mini-Mother or Maxi-Mother board.*
- *Is small in size: 6½ x 4½ inches encased.*

The 40 character mode may be easily viewed on most standard T.V. sets but a monitor is required for the 80 column mode to provide the necessary additional resolution.

Video Combo $319.95

Video Cartridge $219.95

QUANTUM DATA, INC.

Suite A, Box 285, 14252 Culver Dr. • Irvine, CA 92714 • 714/966-6553

Visa & MasterCard are accepted. Above prices retail in U.S. dollars. Shipping and handling not included.

The most popular electronic instrument of the past half-century was invented in 1935 by Laurens Hammond – the Hammond organ, still widely used.

But although these devices were electronic *instruments*, music historians trace the origin of electronic *music* back to Paris in 1947-48. Acoustical engineer Pierre Schaeffer and composer Pierre Henry began experimenting with new sounds by using electronic filtering, speed changes on tape recorders, and other manipulation tricks done in studios. Their technique became known as *musique concrète*, and was quickly picked up by tinkerers elsewhere. By 1952, the first concert of electronic music was sponsored by Columbia University at the New York Museum of Modern Art.

The problem with these techniques was that it took many hours of tedious tape splicing and other tricks to produce only a few brief minutes of sound. And musicians couldn't even hear the results until they were done. That's why there was a lot of interest during the late '50s and early '60s in instruments which could produce electronic music directly. Even the old Theremins from the '20s – updated with transistors – were resurrected.

Toward A New Form Of Music

Robert A. Moog – whose name is virtually synonymous with sound synthesis – was selling kits for transistorized Theremins in the early '60s when he was inspired to invent his own electronic instrument. The result was the Moog Synthesizer, first built in the summer of 1964.

Although recognized by electronic musicians as an important development, the Moog Synthesizer was practically unknown to the general public until a few years later, when it was featured on a record album entitled *Switched-On Bach*. The album was a collection of Bach compositions performed entirely on a Moog Synthesizer by musician Walter Carlos. Almost instantly, *Switched-On Bach* catapulted up the charts like a pop record, and became the biggest-selling classical record of all time. It was especially popular with teen-agers, who astounded their parents by playing electronic Bach along with their Beatles and Rolling Stones records.

However, a few classical music devotees, stunned by the album's popularity, dismissed the electronic interpretations as "artificial." Some critics, although they are decreasing in number, argue that music which is synthesized by purely electronic means is somehow artificial or unnatural when compared to conventional instruments.

Today, Moog counters these arguments with: "The fact is, you don't find musical instruments in nature. The only 'natural' musical instrument is a human voice. The fact that a synthesizer produces its music by electronic means doesn't mean it's 'artificial' in any sense. It's no more artificial than taking a bunch of wood and gluing it together into a box and stretching some strings over it to produce sounds."

Electronic musicians, of course, never had any doubt that their instruments deserved equal billing with violins and woodwinds. In fact, years ago they recognized synthesizers as a rare historical opportunity to open a new world in music. Although synthesizers are often used to mimic "conventional" instruments, the most exciting electronic music takes advantage of the synthesizer's power to create totally new sounds. This provides the possibility of entirely new forms of music.

For example, would rock 'n' roll have happened without electric guitars? Did the invention of a musical instrument with a totally fresh sound spur the rise of a new genre of music? For the members of a whole generation, rock has become the dominant musical style. Synthesizers are now used in virtually every form of music, but even Moog isn't sure if they will "liberate" themselves and spark a new form which could replace rock. "Musicians are moving in so many different directions these days that it's hard to say if a new musical form will emerge."

It may be too early yet for the birth of a dominant musical form based on synthesizers, since the instruments themselves are changing so rapidly. Not only are they advancing technologically almost day by day, but the rising use of microchips is just beginning to make them affordable for everyone. To return to the rock 'n' roll analogy, it would have been difficult for the teen-age groups of the '50s and '60s to arise if electric guitars had cost thousands of dollars. Or if radical new advances were constantly rendering three-year-old guitars obsolete.

Synthesizers, on the other hand, are still passing through important phases in their development. Moog foresees a trend away from analog sound synthesis to digital, or at least to digitally controlled analog instruments. "There's so much more you can do with digital sound synthesis, especially in small keyboard instruments like the little Casios or Yamahas you can buy very inexpensively."

About a year ago, Moog set up a new company – Big Briar, Inc. – and relocated to a small town in rural North Carolina to work on such developments. Among his frequently used tools, he says, is an Apple II microcomputer. Mindful of the baffling array of controls on modern synthesizers, he's experimenting with new types of control devices aimed at making synthesizers easier to play. But he warns that the complex instruments will never be a cinch.

"Musical instruments will never be easy to

Make your fun and games computer get serious
For only $79.95

VIC-20[1] The Wonder Computer of the 1980's. Less than $300. One heckuva lot of fun. But if you're using your VIC-20 strictly for recreation and fun, we at Micro-Systems think you're not getting your money's worth. Because we've designed the Micro-Systems IEEE-488 Cartridge which allows you to interface VIC and CBM 64 to all existing Commodore IEEE peripherals (8050, 4040, 2031, and printers). In addition, we've designed the RS-232 Interface board to allow the VIC-20 and CBM 64 to communicate with various serial devices such as printers, modems, and other systems— **NOT JUST A BUFFER DRIVER.** Also available is Micro-Systems Modem for the VIC-20 and CBM 64 which allows your computer to utilize auto-dial and auto-answer control over the communication process. If you need to access more than one cartridge in your VIC-20 expansion port, we have the solution. The V-Expander is now available with 3 or 6 additional expansion ports. The 6-slot V-Expander is switch selectable.

Available from Micro-Systems

CIE Cartridge [IEEE-488]	$ 99.95
VIE Cartridge [IEEE-488]	$ 79.95
C-232 RS-232 C64 Interface	$ 45.00
V-232 RS-232 VIC Interface	$ 45.00
V-Modem	$ 129.00
V-Expander [3-slot]	$ 49.95
V-Expander [6-slot]	$ 85.00
VIC-1210 3K Expansion	$ 35.00
V8K RAM Expansion	$ 49.95
V16K RAM Expansion	$ 95.00
V24K RAM Expansion	$ 149.00
VIC-20 Color Computer	$ 195.00
VIC-1541 Single Disk Drive	$ 475.00
VIC-1525 Graphic Printer	$ 345.00
VAC-Audio Cassette Interface	$ 29.95
VMC/CMC-VIC & C64 Monitor Cables	$ 19.95
Joy Stick [Arcade Quality]	$ 29.95
VTE/CTE-Terminal Communicator program for VIC and C64 — Cassette	$ 10.00
Diskette	$ 15.00
VT 40-VIC 40 Col Terminal Communicator Cartridge	$ 65.00
Super Expander	$ 65.00
Programmers Aid Cartridge	$ 55.00
VICMON Machine Language Monitor	$ 55.00
Intro to Basic Programming	$ 24.95

IMMEDIATE DELIVERY

Yes, Please send me:

QTY.	MODEL#	NAME	PRICE
TOTAL	(In Texas, add 5% sales tax)		

Your Name ______

Address ______

City ______ State ______ Zip ______

payment method: Check ☐ Card-Exp. Date ______

Master Card # ______

Visa # ______

American Express # ______

Signature ______

Micro-Systems • 2554 Southwell • Dallas, Texas 75229 • (214) 484-7836

[1]Registered Trademark of Commodore Int.

play," he says. "If it's too easy to play, most musicians would say it's not a musical instrument, because you usually can't do much with an instrument that's 'easy to play.' It has too many limitations."

Still, as synthesizers get easier to manage and less expensive, they become more accessible to the average musician – and thereby more widely heard and appreciated by music listeners. Pop music historians may recognize this trend as the same sort of breeding ground for rock created by 45 rpm records in the '50s.

SID: Synthesizer On A Chip

One important way in which synthesizers are becoming accessible to people is within home computers. Virtually every new model introduced in recent years has featured more sophisticated sound capabilities.

Unfortunately, up to now, the sound capabilities have attracted less attention than the often more glamorous feature: graphics.

"Well, in terms of the human senses, sound inherently takes a backseat to sight," notes Frank Covitz, a New Jersey research scientist whose sideline is computer music. "Sight is the more important sense, so computer graphics naturally gets more attention."

For instance, very little has been written on the Atari computer's sound capabilities, although the built-in four-voice sound chip has represented the state-of-the-art in home computers for the last couple of years. Almost all the attention has been focused on the Atari's graphics. Perhaps this will change now that a computer with even more advanced sound has appeared on the market – the Commodore 64 with its SID (Sound Interface Device) chip.

The new SID chip is generating lots of interest among computer music enthusiasts. It may well be a herald of the sound capabilities of tomorrow's home computers. "I think machines of that class [home computers such as the 64] in the future will be expected to have sound chips, just as they are expected to have the BASIC built into them now," says Chamberlin, the MTU engineer. "For one thing, the sound chips are relatively cheap in large quantities, so there's no real reason not to."

SID is a hybrid digital/analog device with programmable attack, decay, sustain, and release for each of its three voices, a master volume control, a choice of four waveforms, 16-bit frequency resolution over a nine-octave range, and programmable high-, low-, band-, and notch-pass filters.

"The SID chip is basically a synthesizer on a chip," says its designer, Bob Yannes. "I played with synthesizers for years, so I'm quite familiar with them. I tried to put it all on a chip with the SID chip."

Yannes designed SID while an engineer for MOS Technology, which is owned by Commodore. He recently left Commodore to form his own company, Peripheral Visions, Inc. Although he won't say for sure what new products his company will introduce, it seems likely that computerized sound devices will be among them. He says chips such as SID are the key.

"There's no reason we can't take music systems being sold now for $4000 and bring them out for consumers for around $400 or $500 – a ten to one cost reduction. I consider the [Commodore] 64 to be only the first step. In the future I'd like to see something totally digital. I think that's the way to go.... I pretty much got the features that I wanted out of the SID chip in the 64, but not the performance I wanted. But now that I've done it once, I think I have a better idea about how to go about it next time."

Yannes says he was given specifications by Commodore only to develop a "sound chip," and then he decided to make it as much like a synthesizer as possible. But he had to work within the limitations of marketing considerations. For example, although SID allows each voice's envelope to be individually programmed, all three voices share the same volume control.

"I had to put separate envelope controls for each oscillator [voice] into the SID chip in order to satisfy the video game/sound effects marketing demands. If I had my way, the three oscillators would work in unison to create one voice. Anyway, that's why there're separate envelope controls for each oscillator but only one peak amplitude [volume] control – it was designed to function as one voice. You could vary the attack of the different oscillators, for example, to get a brassy sort of sound that way."

But Yannes bestowed SID with yet another feature to compensate for this limitation – an input line. It's possible to feed an outside sound source into a computer equipped with SID, process it through the chip's filters and volume controls, and output the extra source as a "fourth voice" in accompaniment with SID's regular three voices. In the case of the Commodore 64, for instance, the outside source would be routed through the RF modulator to the TV speaker – or a stereo system.

What kind of outside sources can be fed into SID? "You name it," says Yannes. "Tape recorders, radios, electric guitars, even another SID chip."

Note that last item: another SID chip. "One thing I thought you might be able to do is chain a bunch of SID chips together to get even better sound, without having to use external hardware," explains Yannes. "I designed the SID chip as a

standard 6502 peripheral chip with all the proper bus signals. You could put some SID chips in a cartridge and plug it into the 64, or the VIC-20, or the Atari – any 6502, 6809, or even 68000 system, even the Radio Shack Color Computer. It only requires 32 address locations, and the chips are pretty cheap, so there's not much to stop you."

It's an exciting prospect, but Commodore controls the SID chip, not Yannes. And for now, Commodore needs virtually all the SID chips it can make to meet demand for the new 64, plus the upcoming Max Machine, P Series, B Series, and BX Series computers soon to hit the market.

Still, a few SID chips have reached private hands, and the results are fulfilling their creator's hopes.

The Synthesizers Of Tomorrow

Chamberlin, the MTU engineer, got four SID chips from his friend, Yannes. Chamberlin used them to make a prototype sound board for the MTU-130, a high-end personal computer for which he designed most of the circuitry. He then passed the board and SID chips along to another friend, Frank Covitz, the New Jersey research scientist. Covitz added four more SID chips to the board, for a total of 24 individually programmable voices. The board is plugged into an MTU-130 equipped with an organ keyboard which, in turn, is controlled by its own 6502 microprocessor.

The instrument made its first public appearance recently when Covitz's son, Philip, gave a performance at the Personal Computer in the Arts Festival in Philadelphia.

Ironically, Covitz says he didn't play his own invention at the festival because he's not a good enough musician. But he's working on software which not only will exploit the instrument's souped-up capabilities, but which also will make it playable by mediocre musicians. This is called *non-realtime* playing.

Musical instruments are usually played in what's known as *realtime:* the music is heard instantaneously as the musician plays the instrument. When an instrument is played in non-realtime, the keying of notes is a separate event from the playing of the music. Notes are entered (the computer instrument is programmed), and then played back (the program is run).

An example of this on home computers is the Atari *Music Composer* cartridge. Essentially, it does for music composition what word processing does for writing. Notes are entered on the computer keyboard and plotted on staffs drawn on the screen. The notes, measures, and phrases can be edited and arranged at will, then played back at the touch of a key. Similar composition programs are available for other personal computers.

"One of the things that computers can do is change music from a physical endeavor to a programming endeavor," says Chamberlin. "That's one of the reasons why I got into computer music – my total lack of dexterity. Even if you're a total butterfingers like me, you can experiment with computer music."

Covitz is striving to push the concept even further. He's added four special keys to his prototype board: Record, Play, Fast Forward, and Rewind. But don't mistake it for a conventional tape recorder – the keys are similar in function, but not in method.

When the Record button is pressed, the computer will "remember" whatever music is played. But no recording tape is involved. Instead, each keypress on the organ keyboard generates information coded in four bytes: which key was pressed, the velocity (how hard the note was played), and the exact moment the key was pressed, accurate to a split-second. Another four bytes of information are generated when the key is released, for a total of eight bytes per note. All this information is stored in memory so the music can be reconstructed later.

After a musical part is "recorded," the Play button can be used to play it back – in accompaniment with a matching musical part played by the musician on the organ keyboard. And this duet, in turn, can be "recorded" in memory by a second unit. Using just two of these "memory recorders," the process can be repeated again and again, layering sound upon sound.

While the same thing can be done with conventional tape recorders, the sound would deteriorate with each generation of re-recording. Tape hiss and other defects would soon overcome the music. But since Covitz's instrument "records" the sound digitally, there is no degradation whatsoever. Beyond that, the music can be "edited." If a note is missed, the musician can correct it by rewriting the correct values into memory.

"This is what I see as the ultimate system," says Covitz. "Right now, this software doesn't exist anywhere except in my mind. I'm in the process of working on this now, and it's all being done in machine language."

The brief history of home computing – and indeed, home computing itself – indicates that advanced technology eventually works its way down to the personal level. It's not hard to envision the day when plug-in organ keyboards and cartridges with add-on synthesizer chips will transform home computers into the kind of instruments Covitz is experimenting with now.

"Seeing what the SID chip can do, and do digitally, I expect you'll see an explosion of that sort of thing," says Covitz. "There has to be. It doesn't require very much hardware. There definitely will be an explosion in complexity." ©


PETSPEED
Now for COMMODORE 64!
FAST ENOUGH FOR THE HUMAN RACE
Our alien won't hang around for slow software. He wants crisp responses and really fast processing.
For the human race too, slow PET BASIC is not good enough. When we run a program, whatever it is, we want fast efficient action.
PETSPEED, the compiler recommended by Commodore, is now available for the 64 and CBM 2. It can make any BASIC program run many times faster. It even speeds up disk handling. We guarantee that PETSPEED is easier to use and generates faster code than any other BASIC compiler for Commodore Systems.
Using PETSPEED is simple. Just type in the name of the program, wait a few minutes and then watch your software run up to 40 times faster.
Petspeed is not simply a compiler, it contains a powerful OPTIMISER. While PETSPEED is compiling, it breaks your program down into tiny fragments and reassembles it removing the unnecessary and simplifying the complex. Dazzling graphics. Lightning sorts. With PETSPEED anything is possible.
Also available INTEGER BASIC COMPILER — 150 to 200 times the speed of Basic. Integer Basic is for those applications where the speed of machine code is required without the inconvenience of assembly level programming. Ideal for scientific and educational users. Compatible with Petspeed.
PETSPEED (Commodore 64) $150
PETSPEED (8000 or 4000 series) $300
INTEGER BASIC (8000 or 4000 series) $175
SPECIAL OFFER: Petspeed PLUS Integer Basic $435
SSE (415) 964-8201
SMALL SYSTEMS ENGINEERING
1056 Elwell Court • Palo Alto, CA 94303
www.commodore.ca

Part I:

Writing Transportable BASIC

Edward T. Ordman
Department of Mathematical Sciences
Memphis State University

If you think your programs might ever be used on another computer with a different dialect of BASIC – the suggestions in this two-part article can go a long way towards easing the transition. This month the author covers documentation, vocabulary, and readability. The article concludes next month with an overview of highly machine-sensitive issues such as input-output and graphics.

So you finally got your own computer. Unfortunately, it is not the same model you had at school. Or you've arrived at high school or college and the computer there is not the same one that your junior high school or high school had. What are you going to do with all the programs you have accumulated? My own school has just bought several of the new IBM Personal Computers – but most of the programs we have on hand were written for a mainframe or for our OSI microcomputers. Come to think of it, we are changing mainframes next semester, too!

Of course, all of these machines have a version of BASIC. (Some of them, in fact, have several versions of BASIC.) But, as is clear to anyone who has read a program written in Apple BASIC and wished he could run it on his Atari (or PET or TRS-80 or ...), all BASIC interpreters are not the same.

What is the solution? There is no ideal solution, for all cases. Some published programs are difficult to convert from one dialect to another. We can, however, in writing programs for ourselves, for friends, and perhaps even for publication, try to make our programs *transportable*. That is, we can write the programs so that they can be adapted to another machine with a minimum of difficulty.

Self-documenting

A program is easily transportable from one machine to another if it can be entered and run in the second machine with no substantial rewriting – certainly no changes in the underlying logic or algorithms – and a minimum of minor changes. The program should be self-explanatory so that it can be rewritten without knowledge of the original machine – a knowledge of the machine we are rewriting it for should be enough.

I have one fairly complex simulation program that was first written about 12 years ago for a PDP-8. It has since been rewritten, by me or by others, for S-100 bus machines in CBASIC, Apple, TRS-80, IBM Personal Computers and IBM 370's, Xerox Sigma 9, PDP-11, and enough other machines that I have lost count. I suspect that it would have been forgotten after the second or third transportation to a new machine, if it had not been written so that it was usually just a matter of typing it in again.

I should warn you at the outset that all this article considers is how to write the BASIC program. It does not address the problems of getting a program from one machine to another without having to key it in again. Increasingly, it is possible to connect the two computers over a phone line, directly or via one of the dial-up timesharing services, and move the program as a text file to avoid retyping. Nevertheless, the focus of this article is transportable *programming* techniques.

What can you do, when writing a program, to make it easily transportable? We will divide the strategy into five main parts: 1) minimal vocabulary; 2) in-program readability; 3) formal structuring; 4) careful attention to input-output; and 5) limited graphics.

Minimal Vocabulary

First, let's consider the question of vocabulary – what features of BASIC we should use. Apparently, whenever a company produces a new computer or a new version of BASIC, it feels compelled to add features not found in anyone else's BASIC. Often these features are convenient and may make programming for that machine easier. However, they make transporting a program much harder. If at all possible, such features should be avoided when writing with transportability in mind.

MEET THE NEW KID ON THE BLOCK.

Catch-up with the newest, wackiest video game you've ever played! Kid Grid! But don't get caught with your guard down because the mischievous bullies Squashface, Thuggy, Muggy and Moose are after you in hot pursuit.

It's the wildest, fastest chase you've ever been in. You've got to connect all the dots on your grid: keep on the look-out for the mysterious bouncing question mark and don't slow down at corners! Squashface, Thuggy, Muggy and Moose are always on your shirttails. Your secret weapon—the joystick button. Press it and the bullies get zapped! When they turn white and freeze, make your escape. If the bullies finally catch you the results are explosive!

With Kid Grid you get all the sights, sounds and colors of arcade games. The better you get the faster and wackier Kid Grid becomes. Soon you'll be racing around at hyper-space speeds!

Kid Grid by Tronix. A hi-resolution video game written in 100% machine code. Designed for the Atari 400 & 800 home computer. Available now at your dealer for $29.95 (suggested retail price).

So meet the new Kid on the block. He'll be your pal forever.

Dealer inquiries invited.

Tronix Publishing, Inc., 701 W. Manchester Blvd.
Inglewood, CA 90301 (213) 671-8440

TRONIX

If we must use special features, they should be isolated in a subroutine near the end of the program and clearly labelled. The main program should stick to features found in virtually all versions of BASIC. This does not mean that string handling must be restricted to the limitations of Radio Shack Level 1 BASIC, which is an extreme example; nor are there universal rules as to what constructions are allowed. Some textbooks define "minimal BASIC" or restrict themselves in a similar way.

Educational institutions often belong to groups (consortia) which promote standards for exchanging programs; CONDUIT is one such educational group with a nice pamphlet on standard BASIC. If you have worked with several versions of BASIC, sticking to common features is a good guide for what will be transportable between them. For informal use, however, or for the individual who has just worked on one machine, here are the standards I have found useful in working with perhaps a dozen different machines, large and small.

Variables And Commands

Figure 1 suggests some guidelines for variable names, numbers, line numbers, DIM statements. Clearly, the list could be made much longer. For instance, how big can a real number be and not overflow? How small can a positive number be and still be distinguished from zero? Most BASIC programs do not depend critically on these figures, which may differ dramatically from one system to another.

Figure 1: Variables and Numbers

Line numbers: 1 to 9999

Variable names: One letter or one letter and one digit. Strings, one letter and $. Examples:

A B2 C9 F$ Z$

Dimensions: Always declared if needed; execute the DIM statement once, before using the variables. If possible, stick to one subscript for strings, two for numbers. Do not use variable sizes or reuse letters. DIM C$(50),D(20,10) is good; DIM B(N), A(50),A$(20) is bad.

If your program does depend on them, you should probably make this explicit (and include a REMark giving the limits on your system). For instance, if your program has a variable X that gets closer and closer to zero as you go around a loop, and you exit the loop by testing IF X=0 THEN ... , the program may behave very differently or even fail on another computer. Changing this to

```
500 IF ABS(X)<1E-50 THEN ... : REM USE A SMALL
    NON-ZERO NUMBER
```

will make the program transportable: the person converting it can check to see if the new computer will accept 1E-50. If it will not, he can substitute an acceptable number, e.g., 1E-30.

Figure 2: The most common statements

DATA	LET
DIM	NEXT
END	ON...GOTO
FOR...TO...STEP	PRINT
GOSUB	READ
GOTO	RESTORE
IF...THEN...	RETURN
INPUT	

Figure 2 shows a limited list of BASIC commands – a very limited list. While almost every BASIC accepts more commands than these, they differ on which statements those are. For each command not on this list, there is some computer around that will not accept it. To make matters worse, computers differ substantially in how they interpret some of these commands. Some, for instance, do strange things on a STOP but allow END only as the last line of a program. The cure: place 9999 END as the last line of the program, and terminate anywhere else by GOTO 9999.

GOTO and GOSUB should be followed just by a line number. GOTO 500 is fine; avoid GOTO A even if your computer likes it. In the statement FOR X=A TO B STEP C, it is best to restrict A, B, and C to integers (or expressions evaluating to integers) and to avoid changing them inside the loop. NEXT must name just one variable for the corresponding FOR, e.g., NEXT X.

IF...THEN statements require special attention, since so many computers have so many different extensions. A few computers accept only statements such as IF Y>=Z THEN 830, prohibiting calculations, logical operations, and not allowing anything but a line number after THEN. I am not seriously suggesting that you keep things this simple: the extensions are extremely helpful. However, it *is* a good idea to keep things simple enough so that your statements can be translated into this form. This will be discussed further in the section on structure, next month.

Numeric And String Functions

Figure 3 shows the most commonly implemented numeric functions. Either most BASICs have these functions, or the programmer using the machine will be prepared to fake them somehow. Two deserve special mention: RND and TAB.

RND is implemented differently on almost every computer. Some use X=RND, some use

THE ULTIMATE IN FAST ACTION FOR YOUR VIC-20

Dive Into a Fearsome Fight!

Grab your gloves, survival pack and headgear! Your Sidewinder commandos are whooshing off into the most frenzied fight in the far side of the galaxy! You've got to out-maneuver deadly Battle pods, dodge destructive Stalker bombs and go head-to-head with alien Oblitojets... all at speeds you wouldn't imagine possible!

Suggested Retail Price $29.95

Blast An Insect Invasion!

Dive into the most awesome all-out battle ever! You'll be attacked from all directions by a barrage of deadly android wasps and alien creatures like you've never encountered before! Any contact means instant destruction! If it's not the fastest game you've ever played, you're from another planet.

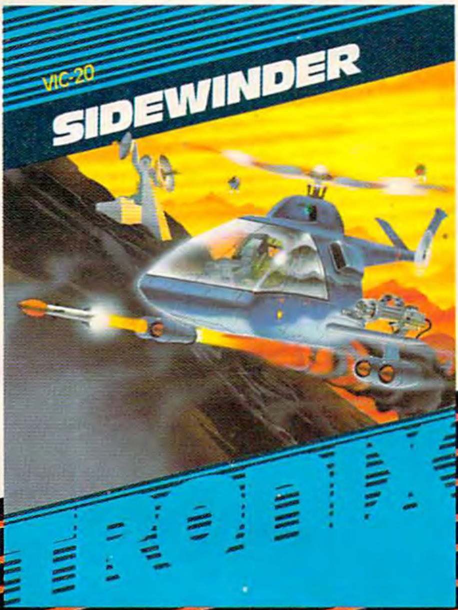
Suggested Retail Price $29.95

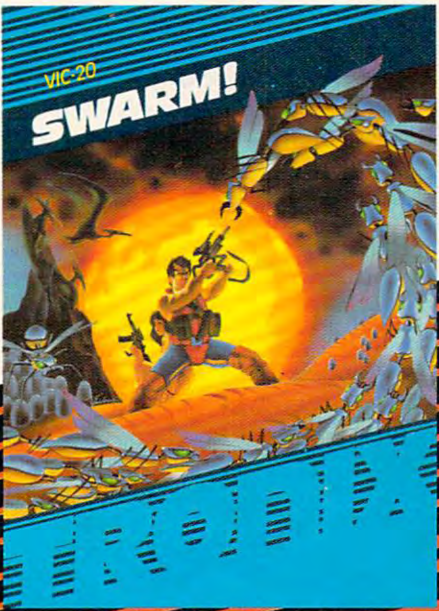
Attack Crazy Aliens!

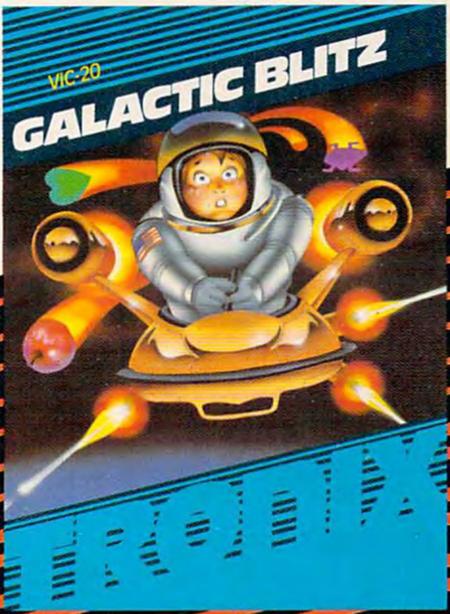
Fall back into the far fields of the galaxy! That bumbling formation coming at you is the crazy Galactic Blitz. These aliens have 15 different play patterns. And each time you take one out of the game they come back mad as ever! So go for the galactic score full speed ahead! If you're merely a spectator, find another sport.

Suggested Retail Price $24.95

Dealer inquiries invited. *Tronix Publishing, Inc., 701 W. Manchester Blvd., Inglewood, CA 90301. (213) 671-8440.*







TRONIX

Figure 3: The most common numeric functions

ABS	COS	INT
RND	SIN	TAB
ATN	EXP	LOG
SGN	SQR	TAN

X=RND(0), some use RND(1), some use RANDOMIZE to start (seed) the random number generator and some do not. You should assume that every line containing RND will have to be rewritten. You should make this as easy as possible, by minimizing the number of lines involved and making your intention clear. If you need a random number in 20 different places in your program, do not have RND appear in 20 places; place it in a subroutine. That is, incorporate in your program

```
9000 REM *** GET RANDOM NUMBER, CHANG
     E FOR OTHER COMPUTERS ***
9010 X=RND(1):REM  RANDOM,0<X<1,NEW S
     EQUENCE EACH RUN
9020 RETURN
```

and then place GOSUB 9000 wherever needed in your program. Here is a more typical use, near the start of a game program:

```
150 N=INT(100*RND)+1 :REM RANDOM INTEGER
    1 TO 100 ***************
```

Here the string of asterisks warns you, when transporting the program, that the line is likely to change. The remark tells what is wanted and will save a lot of time if the new computer achieves this by N=RND(100).

Turning briefly to TAB : there are computers that like TAB(N) (go to column N), those that like SPC(N) (print N spaces), those that like both, and those that like neither. Most people know how to juggle spacing on their own machine, so making your intention clear (by remarks or a sample printout) is probably more important than the exact way you write your PRINT statements. There will be more on this in the discussion of input-output, next month.

Figure 4: The most common string functions

ASC(X$)	LEFT$(A$,N)
CHR$(N)	RIGHT$(A$,N)
VAL(X$)	MID$(A$,I,J)
STR$(X)	

The functions given in Figure 4 are now remarkably widespread in *microcomputers*. It is probably safe to use all of them freely in that context. That is, if the person rewriting the program does not have LEFT$, he probably has a reasonbly direct substitute. You cannot count on the format produced by STR$ being the same from one machine to another – some pad with blanks on the left, some on the right, some not at all. Functions that match a substring are present on many machines, but absent on many others. Many systems will crash if you call LEFT$(A$,N) and A$ has less than N characters, so you should always test for this before you call LEFT$ even if your system does not insist on it.

Large computers differ substantially in how they handle strings, and are often *more* restrictive than small computers. ASC and CHR$ are frequently absent; many large computers do not even use the ASCII character set. Avoid extensive string manipulations, or at least place them in a subroutine, if your program may have to run on a large mainframe next year.

Readability

Next, if our program is to be readily transportable to another version of BASIC, it must be readable. First, can the reader understand our individual lines, and translate them for the new system? Second, can the reader understand our general strategy or procedure (our *algorithm*) well enough to debug the program if errors creep in, or if his BASIC interprets some command very differently than expected?

The most important consideration, for the second of these, is to make the program sufficiently modular and to provide appropriate REMarks for each module; this is addressed more in the discussion of structure, later. There are a number of "tricks of the trade" that make individual lines easier to read, however. Here are a few principles:

1. Leave plenty of space between line numbers. Even if you have only one command per line, some one-line commands on your system may become multiple commands on another. If you use several commands per line, the situation gets far worse. This is not to condemn all multiple-command-per-line statements, since they can add to the clarity of the program. Just remember that while your computer may allow:

```
500 INPUT "WHAT IS YOUR NAME?"; N$
```

someone else's may require

```
500 PRINT "WHAT IS YOUR NAME";
501 INPUT N$
```

This is an easy change if you left a line number available. It is quite possible for a complex one-line statement on one system to require six or eight lines on another.

2. Leave plenty of blanks in your commands, where appropriate. You may have no trouble understanding 250PRINTT5 or 300FORI5=PTOM

but a reader will find 250 PRINT T5 and 300 FOR I5=P TO M much easier to copy or edit. Many BASICs *do* insist on the spaces; the new IBM Personal Computer is one that does. Your computer may allow a larger program or run faster if you delete spaces and remarks, but you make the program much harder to transport when you delete them. It may be worth keeping two programs, a transportable copy and a condensed, quick-run copy.

3. Avoid unprintable characters. Where a few are necessary, find a way to make their presence visible. For instance, a disk read in Applesoft requires that you PRINT a CONTROL-D followed by a string. You can make this readable by

```
200 D$=CHR$(4) :REM CONTROL-D
...
540 PRINT D$;"OPEN FILENAME" :REM DOS
      COMMAND STARTS CTRL-D
```

It is a good idea to indicate what other CHR$ characters are when they are created, too – for instance when CHR$ is used to put a quote mark into a string, or manipulate carriage returns or line feeds.

4. Identify specific features you depend on. This happens most often in connection with PRINT and INPUT statements. Most of us can guess what someone else's PRINT statements are supposed to do, but the INPUTs are another matter.

Some systems input a sentence like "TODAY IT RAINS" by INPUT A$ and the response ?TODAY IT RAINS; others by INPUT A$ and response ?"TODAY IT RAINS"; others by INPUT LINE A$ or by LINPUT A$ or even by INPUT (FIELD 40) A$. You can make this clear to the reader – so that he can try to do the appropriate thing on his system – by remarks, but clear user instructions within the program are probably even better. For example,

```
110 PRINT "TYPE IN A SENTENCE SURROUNDED
      BY QUOTE MARKS"
120 INPUT A$ :REM SAMPLE "HELLO, JOE,
      WADDAYA KNOW."
```

5. Make cues to the user extremely clear. Remember that you won't be around to show people how to use it; in fact, *no* expert on the program will be around. Give sample answers whenever possible, and protect against invalid answers.

```
130 PRINT"DO YOU WANT TO PLAY AGAIN (
    Y/N)";
140 INPUT A$
150 IF A$="N" THEN 9999
160 IF A$<>"Y" THEN 130
```

Note that invalid answers will cause the question to be asked again.

Next month, examples of portable program structure, input-output, and graphics programming. ©

Mattel's New Home Computer

Tom R. Halfhill, Features Editor

Judging from the inquiries we've been receiving at **COMPUTE!**, people are having a tough time choosing between the current crop of low-end home computers: the Atari 400, Commodore VIC-20, Radio Shack Color Computer, Sinclair/Timex, and Texas Instruments TI-99/4A.

Well, it's about to get even tougher.

Mattel Electronics has announced a home computer aimed squarely at the low-end market. It is *not* to be confused with the long-delayed Intellivision keyboard attachment – which has been redesigned again, incidentally. Mattel says the new computer, dubbed the Aquarius, is due "very early in 1983."

The Aquarius will sell for under $200 retail. The price is expected to vary because the machine will be sold through mass consumer outlets. This means it will be available at a wide range of TV-electronics stores, audio/video shops, department stores, discount stores, and catalog showrooms. As we've seen with the other low-end home computers being marketed this way, prices are greatly discounted because competition is so fierce.

Expands To 52K RAM And CP/M

The Aquarius will come with 4K of Random Access Memory (RAM), expandable to 52K RAM in 4K and 16K steps with plug-in cartridges. Microsoft BASIC is built-in. There is one voice for sound effects or music, and an expansion option (described below) provides three voices. The maximum graphics resolution is 320 by 200 pixels (screen dots) in 16 colors.

The Aquarius can display 256 characters. This includes a 128-character ASCII set with upper- and lowercase, and 128 user-programmable characters, similar to the redefinable character sets on the Atari, Commodore 64, VIC-20, and TI-99/4A computers.

For the Central Processing Unit, the central "brain" of the computer, Mattel chose the Z-80A, an eight-bit microprocessor chip. The use of this chip allowed Mattel to give the Aquarius CP/M capability. CP/M (Control Program for Microcomputers) is an operating system primarily used for business applications. The Aquarius can run CP/M with the addition of a disk drive and disk controller card. It is very unusual that a low-end home computer would have CP/M capability, but a Mattel official explained, "Some home users will be professionals who will prefer to work at home." With CP/M, a huge library of existing business programs will work on the Aquarius.

The keyboard has 49 keys. It's more than the membrane keyboard found on the Atari 400, but is not quite a full-stroke typewriter keyboard like the VIC-20's. The keystroke travel is 1.5 to 2 millimeters, and the keys are made of a rubber-like material instead of hard plastic, similar to the keys on Sinclair's new ZX Spectrum (see **COMPUTE!**, August 1982). The keyboard accepts overlays for special applications. For example, an overlay for BASIC programming allows one-key entry of BASIC commands.

While all of these features are standard in the under-$200 Aquarius, Mattel says it will offer a complete system "in the $500 range" which will include a data recorder (tape drive), a printer, and the Aquarius Mini-Expander. The Mini-Expander is an attachment which adds three-voice sound, two game controllers on eight-foot cords, and two slots for plug-in cartridges. One slot is for memory expansion and the other accepts cartridge programs.

All of the software initially released for the Aquarius will be on cartridges. Mattel promises that eight to ten cartridges will be available when the Aquarius is introduced. This will include education, home management, personal improvement, and entertainment software.

One cartridge will be a low-cost Logo with turtle graphics, the acclaimed learning language for children. Mattel says its Logo is designed to work on a minimum system without extra accessories. Another cartridge will be *FileForm*, a word processor. Mattel says more cartridges will be released monthly after the computer is introduced, and that additional software may be available on cassette in the future. All the software has been developed at Mattel, although the company is now talking to outside developers.

Other expected add-ons include a disk drive and a 40-column thermal printer. No prices have been disclosed for these extras, but a Mattel official did say that the memory expansion boards would be "extremely price-competitive."

Apparently, the Aquarius will lack special function keys, sprite graphics, and full-screen editing. Reportedly, it will have 8K of ROM (Read Only Memory), a 40 by 24 screen display, dimensions of 13 by 6 by 2 inches, and weigh four pounds. It will come with simplified instruction cards so beginners can get the machine working without reading the manual. ©

Richard Mansfield, Senior Editor

Myths About Programming

Whenever someone says that they are "not the type" or they "weren't good in math" as a reason for not learning to program in BASIC, I like to compare learning BASIC to learning how to drive a car. A few people never learn to drive, but most of us do. What's more, it takes a few weeks at most to catch on to either driving or BASIC. You improve over the years, but the essentials can be grasped pretty quickly. And in the coming Age of Information, not knowing how to program could well be as inconvenient as not having a driver's license is today.

After all, there are only some 50 BASIC words to learn. Several of them are very rarely needed (you can look through hundreds of programs and never find WAIT, POS, or TAN). Also, many BASIC words mean exactly what they say: STOP stops a program, RUN runs, END ends.

If someone is still doubtful, the most convincing argument is a demonstration. You can say: "Let's try something complex. How about printing your name 1,000 times on the screen?" Then type: 10 ? "Alan"; : GOTO 10.

Could It Explode?

Another factor which causes hesitation about learning programming is a fear of the unknown. It's not hard to see where this nervousness comes from. There is a category in movies which changes each generation, but could be called the Frankenstein Slot. A scientist gets too big for his britches, tampers with unknown forces, and his creation runs amok while the villagers, in their ancient wisdom, ineffectually stone the laboratory.

During the fifties there were dozens of films in which atomic radiation filled this slot. It caused moths and ants to grow to enormous size, rampaging through cities. When people learned that radiation, dangerous as it can be, could not cause insect giantism, popular entertainment found a new monster. Dozens of movies in the sixties (some of them excellent, like *2001*, *The Forbin Project*, and more recently, *The Demon Seed*) portrayed the computer as Frankenstein.

When people buy their first personal computer, they are not generally worried that it might destroy their house, but they often worry about the computer getting out of their control and damaging itself. They sense, correctly, that a computer is a powerful machine.

You'll see this hesitancy when people look up, their first time in front of the keyboard, and ask, "What should I be careful of? Can I hurt it?" We get letters from beginners wanting to know if they should use POKE, the BASIC word that changes what's in the computer's memory. They have a perfectly understandable fear that, as one New Yorker recently wrote, "I might damage the BASIC ROM chips."

Your computer watches out for most kinds of errors. It simply won't allow you to POKE into BASIC ROM chips. If you try to send a POKE to an address that's in ROM (Read Only Memory), it will just bounce off. Nothing happens. These addresses can only be read, not written (POKEd) to.

We always used to say that nothing you could type into the computer could hurt it in any way. If it doesn't understand what you write, or can't carry out your instructions, it will stop and tell you where it stopped and, in general terms, why. The worst that could happen would be an "endless loop," and you would have to turn power off and back on to get control. None of this, however, would start the computer smoldering.

This advice, unfortunately, is not 100 percent correct. It was discovered that there was one POKE to PETs with Original or Upgrade BASIC versions which would make the video display about six times faster. It's POKE 59458,62. PET/CBMs with the most recent 4.0 BASIC chips, however, *can* be damaged by this POKE if left running without turning off the power. Since there are about 17 million combinations of POKEs you can make into your computer, the odds are clearly against your accidentally making this error. Nevertheless, it does make it impossible to tell someone that nothing you type into any computer could hurt it.

NOT EVERYONE CAN TEACH THEIR ATARI™ NEW TRICKS...

WE MAKE USING AND LEARNING ABOUT COMPUTERS FUN!

PROGRAMMING GUIDE FOR BEGINNERS OR EXPERTS — MASTER MEMORY MAP.™ A 32 page book with hundreds of hints on how to use your computer. Over 500 memory locations! $6.95.

LEARN SOUND AND GRAPHICS with our exciting lessons called **TRICKY TUTORIALS.™** Each comes with a tape or disk full of examples, and a 12 to 64 page manual written in an easy to understand manner. **#1 DISPLAY LISTS** — Put several graphics modes on your screen at once. **#2 SCROLLING** — Move text or graphics smoothly up, down, sideways, or diagonally. **#3 PAGE FLIPPING** — Change TV screens as quickly as flipping pages in a book. **#4 BASICS OF ANIMATION** — A beginner's lesson in animation using PLOT, PRINT, and a surprise game. **#5 PLAYER MISSILE GRAPHICS** — Learn the basics of writing your own arcade games. **#6 SOUND & MUSIC** — Simple methods to play complete songs, with graphics. Includes PLAYER PIANO free! **#7 DISK UTILITIES** — 7 programs to help you use your disk drive. 32K. **#8 CHARACTER GRAPHICS** — The best editor available with examples using special characters YOU CREATE and ANIMATE. **#9 GTIA, GRAPHICS 9 to 11** — New tricks you can do with these 16 color modes. **#10 SOUND EFFECTS** — Many examples, from rainfall to laser blasts, with ample explanation. **#11 MEMORY MAP TUTORIAL** — 30 colorful examples of tricks your computer can do.

TUTORIALS cost only $19.95 each, except 5 and 7 which cost $29.95 each. 16K Tape/24K Disk required.

SPECIAL: Tutorials 1 through 6 in a binder for $99.95. SAVE $30.00!

USER SUBMITTED PROGRAMS

We sell many fine programs written by dedicated computer owners, such as **INSTEDIT, SPACE GAMES, MINI WORDPROCESSOR, DATA-BASE DIALER, PROTO'S ADVENTURES,** and many more.

OUR GUARANTEE: Your money back if unsatisfied!

NO LOCAL DEALER? CALL FOR A FREE CATALOG, OR ORDER DIRECT (CHARGE OR COD): 800-692-9520 OR (408) 476-4901

SOQUEL, CA

The Math Myth

When autos were first becoming popular, there were doubtless many people who refused to try driving, saying, "That's one thing that I will never try to learn. I could never control our horse." A weakness in math is often given as the reason for not trying programming. In fact, the two activities are hardly related. Mathematical words are available in BASIC, but programming does not need to involve much math beyond simple arithmetic unless you choose to solve mathematical problems. You will need to search a long time to find any use of the word SIN in the dozens of BASIC programs published each month in **COMPUTE!**.

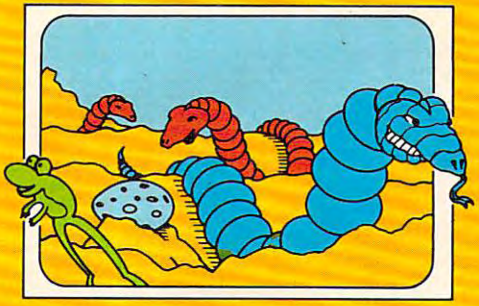
Personal computers are general-purpose tools. They *can* be used to solve complex equations, but to call programming "mathematical" would be too narrow it down to only one of its countless applications. And it would also mislead people into thinking that they need a special talent in math to become competent programmers.

Several years ago one of America's largest corporations undertook a study to find out what its best programmers had studied in college. To nearly everyone's surprise, English and music were the most common majors among the top computerists. Perhaps this is because these disciplines stress creativity and attention to detail. Perhaps they combine logical thinking with imagination. No one has yet given a satisfactory explanation.

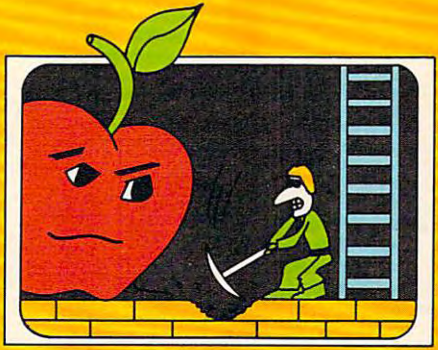
In any case, you don't need to become an ace programmer any more than you need to drive at the Indianapolis 500. To get where you want to go, ordinary driving or programming knowledge will suffice. There are very few people who can't learn the necessary skills.

If there is a topic that you would like to see discussed in this column, send a card or letter to: The Beginner's Page, **COMPUTE!** *Magazine, P.O. Box 5406, Greensboro, NC 27403.* ©

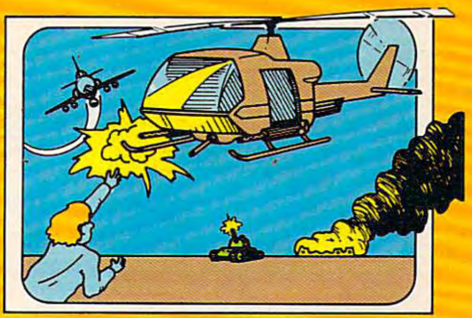
VIC-20®?



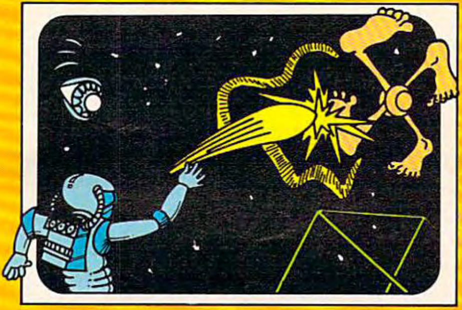
SERPENTINE



APPLE PANIC



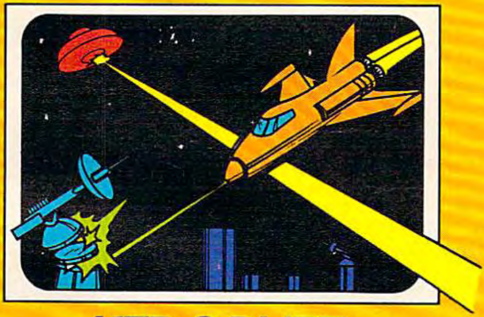
CHOPLIFTER



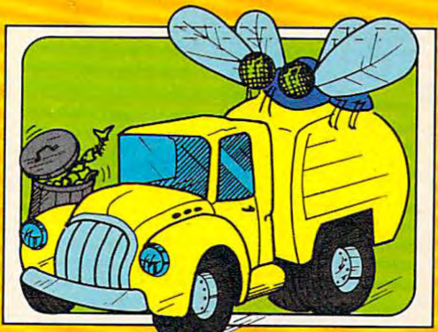
VIDEO MANIA

and these are just the games!

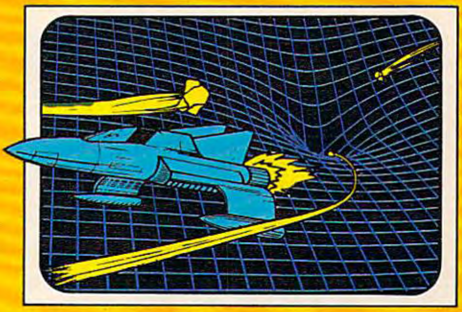
TERRAGUARD



ASTROBLITZ



TRASHMAN



BLACK HOLE

YOUR FULL LINE VIC-20® SOFTWARE COMPANY

CREATIVE SOFTWARE

201 San Antonio Circle • Mountain View, CA 94040

Atari's Sound System

John Scarborough, Novato, CA

If you're interested in the improvements to Atari sound and music possible via machine language – this will get you started. These simple demonstrations might convince you to abandon the SOUND command entirely.

Many programmers who decide to make the jump from BASIC programming to machine language programming find frustration in their attempt to print to the screen or produce sound. The problem is that there are now no print or sound statements. Machine language deals entirely with retrieving, manipulating, and storing data.

But even after the programmer discovers this it won't do him much good unless he knows where and what to store to produce results. Furthermore, this information is often not provided in the manuals that come with the computer. So how does he obtain this information? He must turn to other methods. Four common ones are:

1. The trial and error method (very inefficient, but sometimes necessary).
2. Advanced user's manuals.
3. Information obtained from a human source, such as from friends or teachers.
4. A magazine.

This article is in category four. After studying this article, you will have more control over the four voices provided by the Atari. The article will also briefly cover the built-in speaker.

Sound Commands

Look at the following sound command:

```
SOUND 0,121,X,X
```

This instructs the computer to store a value of 121, which will produce a middle C note, into Audio Frequency Control register 0. This register is located at memory location 53760 ($D200 hex). Thus, the following two commands will function identically:

```
SOUND 0,121,X,X
POKE 53760,121
```

The three remaining Audio Frequency Control registers are located at 53762 ($D202), 53764 ($D204), and 53766 ($D206). A POKE 53764,128 would store a value of 128 (a B note) into the Voice 2 Audio Frequency Control register. SOUND 2,128,X,X will do the same. (See Figure 1 for a clearer representation of the four Audio Frequency Control registers.)

Now you can store a given frequency (note) into any of the four Audio Frequency Control registers. But what about distortion and volume? Look at the following sound command:

```
SOUND 0,X,10,12
```

This tells the computer to produce a pure tone (10) and a volume level of 12. Upon execution, the computer will convert the number 10 to 160 (160 is the actual pure tone code. See Figure 2 to find the corresponding distortion codes for the eight additional distortion levels), add 12 to it, and then store the result into Audio Control register 0. This register is located at memory location 53761 ($D201 hex). Thus, the next two commands will perform the same task:

```
SOUND 0,X,10,12
POKE 53761,160+12
```

The three remaining Audio Control registers are located at 53763 ($D203), 53765 ($D205), and 53767 ($D207). A POKE 53767,160+7 would store a pure tone and volume level of 7 into Audio Control register 3. SOUND 3,X,160,7 will do the same. (See Figure 1 for a clearer representation of the four Audio Control registers.)

You should now know how to store any given note, tone and volume level into any of the four voices provided by the Atari (and without using sound statements). The following two BASIC programs function identically:

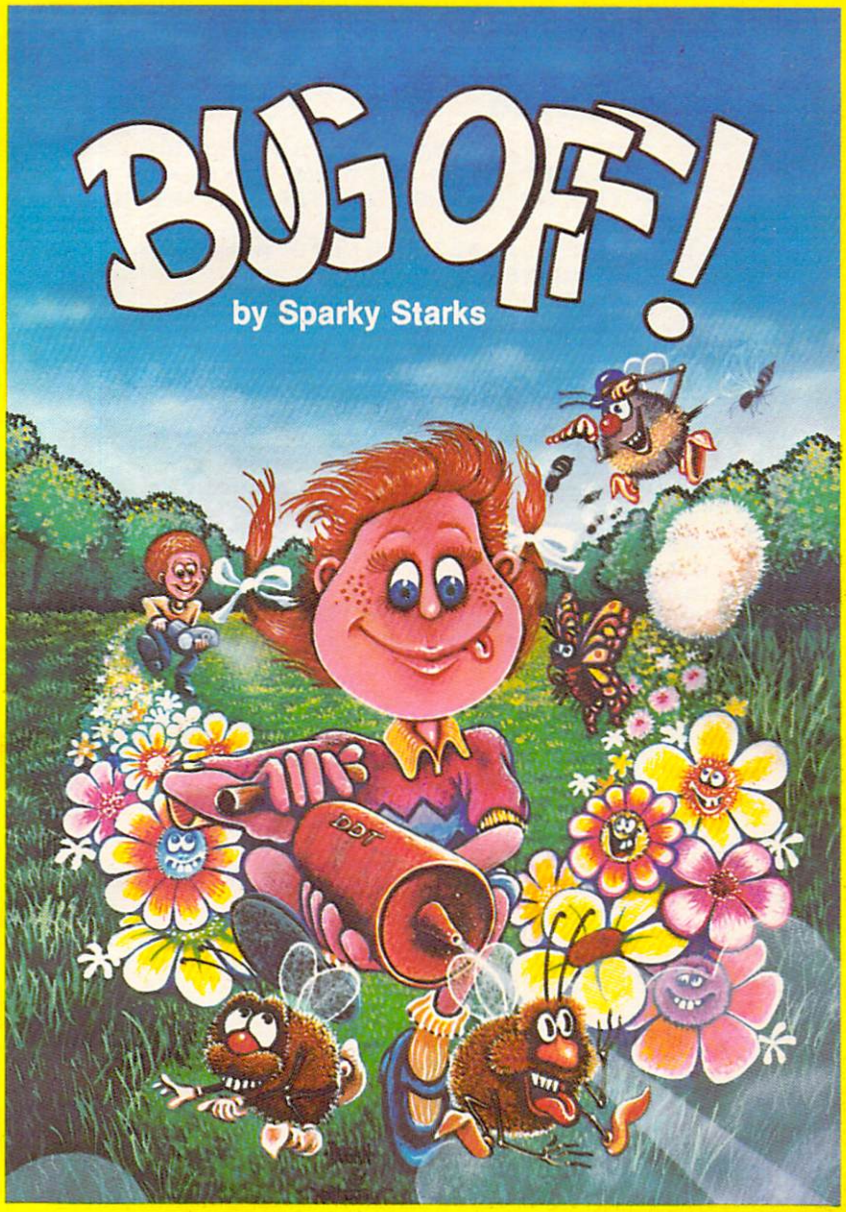
```
10 SOUND 0,121,10,12
20 GOTO 20
```

```
10 POKE 53760,121
20 POKE 53761,160+12
30 GOTO 30
```

The Built-in Speaker

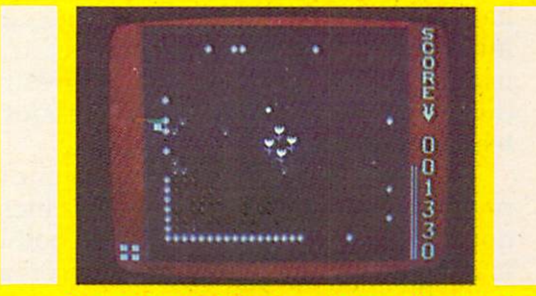
That is an overview of the sound that is channeled to the television speaker, but what about the built-

YOU'RE GOING TO HAVE FUN WITH YOUR ATARI!!



AND THE FUN GETS ROLLING WITH BUG OFF!

Yikes! The bugs are swarming here, there and everywhere and only a strong whiff of DDT can put 'em away. The object of the game is to control the seven different kinds of pests that are running helter-skelter over everything. The Army can airlift in more DDT to fill your bug sprayer . . . but will they make it in time?



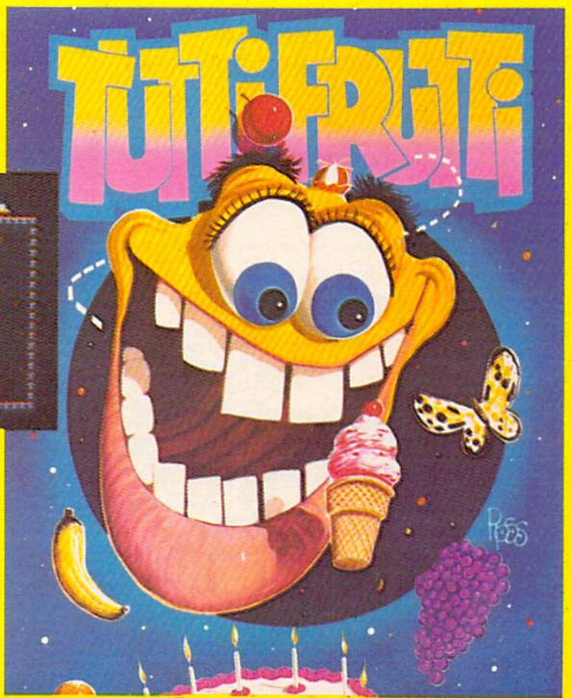
The action builds to a furious frenzy as an awesome assortment of insects attack anything and everything in sight. A definite case of "spray first and ask questions later." All this and hi-res graphics, too!

16K TAPE	050-0167	$29.95
32K DISK	052-0167	$29.95

TUTTI FRUTTI — by Alan Newman

"GRAPE FUN" FOR THE WHOLE FAMILY!

Somewhere between Never Never Land and next Thursday, there's a wild and wacky place where your joystick-controlled "Hungry" snacks on an orchard of goodies. But there are some nasty ol' bugs who'd just as soon snack on YOU, so look sharp and beware. **Fun**tastic action for all ages!



TUTTI FRUTTI . . . Wow! A crazy cast of characters in vivid hi-res color all performing in their natural habitat, otherwise known as TUTTI FRUTTI Land. Grape fun for ages 6 and up!

16K TAPE	050-0160	$24.95
32K DISK	052-0160	$24.95

Adventure INTERNATIONAL

To order, see your local dealer. If he does not have the program, then call 1-800-327-7172 (orders only please) or write for our free catalog.

Published by ADVENTURE INTERNATIONAL
a subsidiary of Scott Adams, Inc.
BOX 3435 • LONGWOOD, FL 32750 • (305) 862-6917

PRICES SUBJECT TO CHANGE

in speaker? The built-in speaker is controlled via location 53279 ($D01F).

Program 1 will make the built-in speaker randomly click. (Not a very spectacular sound effect, to be sure, but that's not to be expected from BASIC statements.) Program 2A and 2B (which function identically) will also make the built-in speaker randomly click, but they do not use BASIC statements to produce the sound and will therefore click the speaker much faster than will Program 1. compare the two for yourself. The comparison will give you some idea of the speed available to you from machine language programming.

If you compared Programs 1 and 2, you might have thought: "If machine language can do that much for that little built-in speaker...." Yes, by using machine language you can greatly increase the quality of the sound that comes out of your television speaker.

I leave you with a simple machine language program that will demonstrate this increase in quality. The program is written in both assembly language (3A) and BASIC (3B). This is a simple program; it would have been half as long without the delay routine. However, the sound would not be audible if there were no delay built into the machine language program!

Figure 1.

The Four Voices Provided By The Atari

VOICE 0	
Audio Frequency Control Register	**- 53760 ($D200)**
Audio Control Register	**- 53761 ($D201)**
VOICE 1	
Audio Frequency Control Register	**- 53762 ($D202)**
Audio Control Register	**- 53763 ($D203)**
VOICE 2	
Audio Frequency Control Register	**- 53764 ($D204)**
Audio Control Register	**- 53765 ($D205)**
VOICE 3	
Audio Frequency Control Register	**- 53766 ($D206)**
Audio Control Register	**- 53767 ($D207)**

Figure 2.

Examples Of The Eight Distortion Levels Using Random Voices

```
SOUND 0,X,0,V  = POKE 53761,0+V
SOUND 0,X,2,V  = POKE 53761,32+V
SOUND 3,X,4,V  = POKE 53767,64+V
SOUND 1,X,6,V  = POKE 53763,96+V
SOUND 0,X,8,V  = POKE 53761,128+V
SOUND 1,X,10,V = POKE 53763,160+V
SOUND 2,X,12,V = POKE 53765,192+V
SOUND 0,X,14,V = POKE 53761,224+V
```

Program 1.

```
5 REM -THIS PROGRAM UTILIZES THE BUIL
  T-IN SPEAKER USING BASIC STATEMENTS
10 A=INT(256*RND(1)):REM -LOAD A WITH
    A RANDOM NUMBER FROM 0 TO 255
20 POKE 53279,A:REM -STORE A AT 53279
30 GOTO 10:REM -START OVER
```

Program 2a.

```
5 ;THIS PROGRAM UTILIZES THE BUILT-IN
   SPEAKER FROM MACHINE LANGUAGE USIN
  G THE ASSEMBLER-EDITOR CARTRIDGE
10 *=$600
20 LOOP LDA $D20A ;LOAD A WITH A RAND
   OM NUMBER FROM 0 TO 255
30 STA 53279 ;STORE A AT 53279
40 JMP LOOP ;START OVER
```

Program 2b.

```
5 REM -THIS PROGRAM UTILIZES THE BUIL
  T-IN SPEAKER FROM MACHINE LANGUAGE
  USING THE BASIC CARTRIDGE
10 FOR LOOP=1536 TO 1544
20 READ DATA
30 POKE LOOP,DATA
40 NEXT LOOP
50 X=USR(1536)
60 DATA 173,10,210,141,31,208,76,0,6
```

Program 3a.

```
5 ;THIS PROGRAM UTILIZES THE TELEVISI
  ON SPEAKER FROM MACHINE LANGUAGE US
  ING THE ASSEMBLER-EDITOR CARTRIDGE
10 *=$600
20 FREQ=$660
30 ;STORE A PURE TONE (160) AND A VOL
   UME LEVEL OF 15 (160+15) INTO VOIC
   E 0
40 LDA #175
50 STA $D201
60 ;STORE CURRENT FREQUENCY INTO VOIC
   E 0
70 START LDX FREQ
80 STX $D200
90 INX ;INCREMENT FREQUENCY LEVEL
100 STX FREQ
110 ;DELAY PROGRAM EXECUTION
120 DELAY LDX #15
130 LOOP1 LDY #15
140 LOOP2 DEY
150   BNE LOOP2
160   DEX
170   BNE LOOP1
180   JMP START ;CONTINUE
```

Program 3b.

```
5 REM -THIS PROGRAM UTILIZES THE TELE
  VISION SPEAKER FROM MACHINE LANGUAG
  E USING THE BASIC CARTRIDGE
10 FOR LOOP=1536 TO 1563
20 READ DATA
30 POKE LOOP,DATA
40 NEXT LOOP
50 X=USR(1536)
60 DATA 169,175,141,1,210,174,96,6,14
   2,0,210,232,142,96,6,162,15,160,15
   ,136,208,253,202,208,248,76,5,6
```

ali baba
and the forty thieves

A fantasy role-playing adventure for one or more players

By Stuart Smith

ALI BABA AND THE FORTY THIEVES is written in high resolution color graphics and includes music and sound effects. Adventures can be saved to disk and resumed at a later time.

Available for Apple II and Apple II Plus 48K or Atari 800 32K.

On diskette only — $32.95

QS QUALITY SOFTWARE

6660 Reseda Blvd., Suite 105, Reseda, CA 91335 (213) 344-6599

VIC Sound Generator

Robert Lee, Vancouver, B.C.

Adding sounds to VIC can significantly slow down a BASIC program. The action stops and waits for the sound to finish. This could be especially annoying when you want a game to run as fast as possible. With this sound generator, you can add sounds in BASIC easily and without a speed penalty.

Among the novel features of the VIC-20 are its sound capabilities. These give it an advantage over the PET, bringing a new dimension to game programs. However, one of the problems I and undoubtedly other VIC owners have encountered is that, while manipulating the sound generators in a BASIC program, it is not possible to do anything else.

This is especially a problem in game programs written in BASIC and using extensive graphics. Either you have to write such programs without complex sound effects, or you have to settle for slow motion.

Faster Sound

Faced with this problem, I decided to write a machine language (ML) program for the VIC which adds speed to its sound generation capabilities. Most of the sound effects we use in game programs are sounds with increasing or decreasing tones. For example, a simple way to simulate the sound of a laser with the VIC is:

```
FOR K=250 TO 240 STEP -1 :POKE36876,K:NEXT
```

The ML program works along these lines, except that it is necessary to use only one POKE command. It generates sounds with increasing or decreasing frequency to make almost any kind of sound effect possible.

The program "VIC Sound" places a machine language program in the cassette buffer of the VIC. This means, of course, that you cannot transfer data using the cassette player while you are running the program. By changing the contents of memory locations 788-789 (decimal), the interrupt system of the computer is used to run the ML program.

As you know, the VIC has four "speakers" to make music and noise. The first and second speakers, activated by POKEing memory locations 36874 and 36875, are used for sounds with increasing tones. The third speaker (36876) is used for sounds with decreasing tones. The fourth speaker, activated by memory location 36877, is used mainly for explosions.

The ML program stores a starting number into the appropriate location and increases or decreases it for the period specified by the user. The interrupt of the computer will run through the program 60 times a second, which means that the starting number or tone will increase or decrease 60 times in one second.

Sound Duration

To make this a little clearer, let me explain that four memory locations have been assigned in the ML program to activate the four speakers, and four others to control the duration of the sounds.

Speaker	To activate	Duration
1st	846	858
2nd	847	888
3rd	848	918
4th	849	948

The number POKEd into locations 846-849 is the starting number which is stored in location 853 (dec); the initial value is 222, but this may be changed for the kind of sound you require. Locations 858, 888, 918, and 948 control the duration of the sounds. The program will generate the sounds for the number of jiffies (the 1/60th of a second interval used to measure time in Commodore machines) specified in these locations.

For a demonstration, RUN the program and then type SYS828; this will trap the interrupt. It will also set the volume control (location 36878) to maximum. Now POKE 846,222.

Location 858 contains 10 (dec), so the sound you heard was for ten jiffies. What the program has done is store 222 in location 36874 (first speaker), incremented it by one every 60th of a second until ten jiffies elapsed, then stored 0 into the memory location to switch off the speaker. To change the duration of the sound to, say, 20 jiffies, POKE 858,20. Now POKE 846,222.

The same method can be used for the other speakers. POKE 858,10. To change the starting number (i.e., to get a tone which starts higher or lower), simply POKE into memory location 853. For example, POKE 853,240. Now POKE 846,240.

Your VIC 20 never had it so good!

VIC 20 owners rejoice! HES presents a complete range of software from our exciting series of games to our professional group of utility and language programs.

Our new cartridge programs include: HES MON, an indispensible monitor for assembly language programmers; HES WRITER, a word processing program; Turtle Graphics, a fun and easy way to learn computer programming; and VIC FORTH, a powerful language that is many times faster than BASIC, yet easier to use than assembly language.

HES is committed to offering high-quality, well-documented computer programs on a continual basis. Look for our cartridge and cassette based software at your local dealer.

HES

Human Engineered Software
71 Park Lane
Brisbane, California 94005
Telephone 415-468-4110

VIC 20 is a registered TM of Commodore

Explosion Simulation

It is necessary to POKE the starting number into locations 846-849; any other number will give only silence. Try POKE 847,240 (second speaker); it gives a sound of increasing frequency like the first. Now POKE 853,222:POKE 848,222. You notice this gives a sound that decreases in frequency. POKE 849,222 will simulate an explosion. By manipulating the durations and starting number, you can get almost any kind of sound from the first three speakers and explosions from the fourth. However, when you are changing the duration of the sounds, make sure it is not too long; e.g., if you POKE 853,50:POKE 846,222 the program will store 222 in location 36874 and increment by one every jiffy for 50 jiffies. But in this case the contents of 36874 would increase to 255 and then cycle back to zero. You would hear a note for only 33 jiffies, since a number less than 128 in the sound generators of the VIC produces silence.

When using this program, you cannot generate sounds the normal way. To do so, you must first reset the interrupt vector by SYS996. This will also set the volume control to zero. To use the ML program, add the subroutine starting at line 8900 to your own BASIC program; and you can create sound effects using just one POKE, which would otherwise require a series of POKEs.

In a BASIC program with lines 8900-9240 added, you would first have a line like this in the main program to enter the ML into memory:

10 GOSUB 8900 : REM SOUND GENERATOR

```
1Ø PRINT"{CLEAR}"
2Ø PRINT"{Ø3 DOWN}{Ø8 RIGHT}{REV}VIC2Ø{OFF}"
3Ø PRINT"{Ø2 DOWN}{Ø6 RIGHT}VIC SOUND"
8ØØ GOSUB89ØØ
9ØØ END
89ØØ FORJ=828TO1Ø19:READF:POKEJ,F:NEXT
9ØØØ DATA169,15,141,14,144,12Ø,169,82
9Ø1Ø DATA141,2Ø,3,169,3,141,21,3
9Ø2Ø DATA88,96,1Ø,15,16,64,16Ø,Ø
9Ø3Ø DATA162,222,173,78,3,2Ø1,1Ø,176
9Ø4Ø DATA9,238,78,3,238,1Ø,144,76
9Ø5Ø DATA116,3,14Ø,1Ø,144,236,78,3
9Ø6Ø DATA2Ø8,6,14Ø,78,3,142,1Ø,144
9Ø7Ø DATA173,79,3,2Ø1,25,176,9,238
9Ø8Ø DATA79,3,238,11,144,76,146,3
9Ø9Ø DATA14Ø,11,144,236,79,3,2Ø8,6
91ØØ DATA14Ø,79,3,142,11,144,173,8Ø
911Ø DATA3,2Ø1,16,176,9,238,8Ø,3
912Ø DATA2Ø6,12,144,76,176,3,14Ø,12
913Ø DATA144,236,8Ø,3,2Ø8,6,14Ø,8Ø
914Ø DATA3,142,12,144,173,81,3,2Ø1
915Ø DATA64,176,28,238,81,3,173,81
916Ø DATA3,2Ø1,22,2Ø8,7,169,176,141
917Ø DATA13,144,24Ø,25,2Ø1,43,2Ø8,21
918Ø DATA169,16Ø,141,13,144,24Ø,14,14Ø
919Ø DATA13,144,236,81,3,2Ø8,6,14Ø
92ØØ DATA81,3,142,13,144,76,191,234
921Ø DATA169,Ø,141,14,144,12Ø,169,191
922Ø DATA141,2Ø,3,169,234,141,21,3
923Ø DATA88,96,Ø,Ø,Ø,Ø,Ø,Ø
924Ø RETURN
```

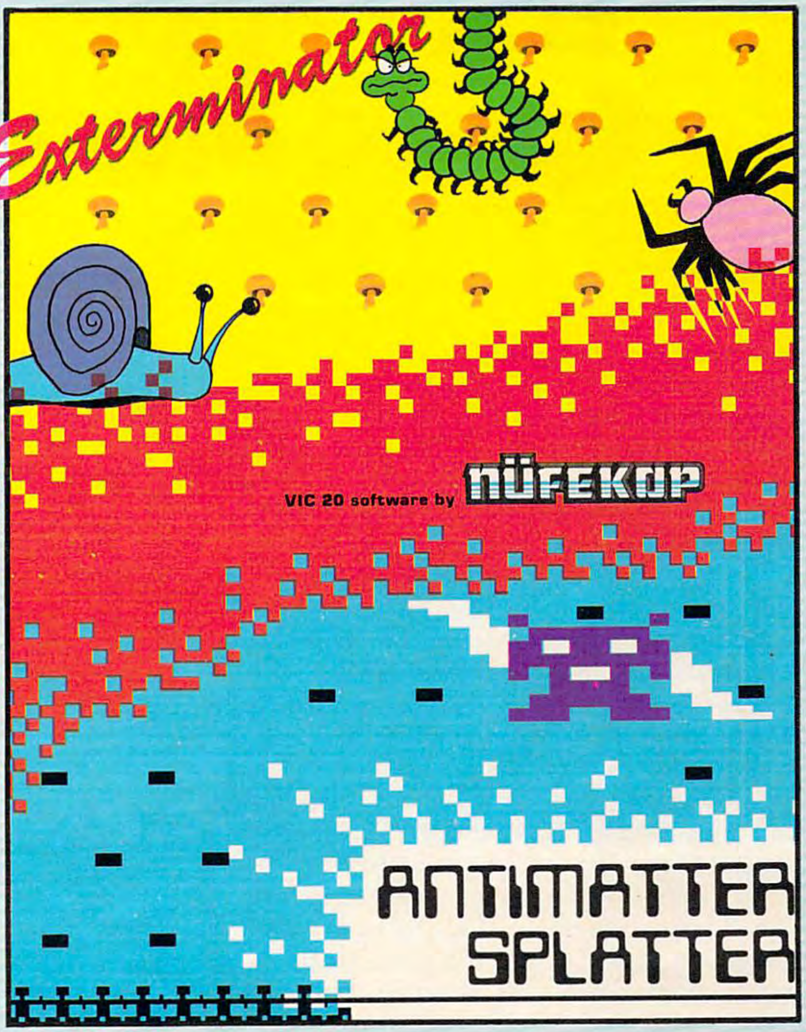
©

Exterminator **by Ken Grant**
First the bad news...this game is literally full of bugs. The good news? We guarantee hours of exciting entertainment trying to remove them. Some bugs you are likely to come up against are spiders, snails, fleas and centipedes in this rapidfire, 100% machine language, exceptional quality game. *Exterminator* runs in standard 5K VIC. **$24.95**

Antimatter Splatter! A more dastardly alien could scarcely be found than one who would wipe out an entire civilization by dropping antimatter anti-canisters, right? If your opinion of this alien troublemaker is the same as ours, probably your first thought was, get some matter! We say calm down! All is not lost. A mobile rapid splatter cannon capable of both breaking through his standard alien moving force fields and laying waste to the ever-increasing number of anti-canisters is even now hovering above us. If only our cannoneer hadn't called in sick...say, what are you doing today? *Anti-Matter Splatter* is 100% machine language and runs in standard 5K VIC. **$24.95**

3-D Man The exact maze from probably the most popular arcade game ever with perspective altered from overhead to eye level. The dots...the monsters...the power dots...the side exits. New on-screen radar. This game is *amazing. 3-D Man* requires at least 3K memory expander but will run with any memory add-ons (8K, 16K, 24K, etc.) that we have come across. **$19.95**

Defender on Tri As pilot of the experimental Defender-style ship "Skyes Limited," you are the only hope for an advance party of scientists trapped in ancient alien sphere which suddenly (heat from collision course with sun presumably—G.E.) came to life. Four screens worth of unique defenses, on-off shields, fuel deposits, alien treasures, running timer, energy, score and very nice graphics display make this one that does not quickly wax old. *Defender on TRI* requires at least 3K memory expander, but will run with any memory add-on (8K, 16K, 24K, etc.) we have come across. **$19.95**



★ ★ SPECIAL ★ ★

Another NUFEKOP Big Deal to close this year out **RIGHT!** We at NUFEKOP want to celebrate by giving you a gift. Our catalog...FREE. No, seriously, if you order both *Antimatter Splatter* and *Exterminator*, or three of the lesser-priced games, you may pick **ONE FREE GAME** from any of the above (except *Exterminator*.) **Happy New Year!!**

Alien Panic Standard 5K VIC 20/combination stick & keyboard. This arcade-type game pits you against time and an alien on a six level construction sight with ladders and pitfalls, but *not to worry!* You have a shovel. **$12.95**

Krazy Kong The crazy gorilla has taken three fair maidens up to the top of the giant stairway. And you (the valiant hero) will attempt to rescue them at the risk of your own life. Your timing must be totally accurate as you jump the barrels that Kong is rolling down at you. **$12.95**

Rescue From Nufon *Adventure* $12.95
Collide *Crunch* $12.95
Vikman *Classic* $12.95
Search *Challenging* $12.95

SPECIAL OFFER! With any order of *Exterminator* and *Anti-Matter Splatter* we'll send you a *FREE* 17" X 22" poster like the one shown above. Also available separately for $3.50.

nüFEKOP

P.O. Box 156, Shady Cove, Oregon 97539-0156
C.O.D. Orders...call (503) 878-2113
Mastercard and Visa cards accepted
Ask for our new FREE catalog!

Games will be on tape unless you request disk.

Easy Apple Disk Space Messages

Beirne L. Konarski, Kent, OH

As diskettes fill up, it is useful to know how much room is left on them. Two methods exist for finding this number. One is to get a calculator or pencil and paper and add the figures. This seems pointless, since the computer is supposed to do those tasks. The alternative is to run the FID program from the system master. This is a nuisance, though, because you often change diskettes.

Since many people incorporate a CATALOG command into their greeting program, this would be the most useful place for a sector-counting subroutine. The *DOS Manual* contains two sections which help to solve this problem. The first is the RWTS (Read or Write a Track and Sector) machine language program (p. 94). The second is the listing of sector allocation (p. 129). The RWTS subroutine can be used to read the sectors containing the catalog and tally the amount of sectors used.

The diskette directory is located in track $11. It contains 15 sectors of catalog information, each holding the names and sizes of seven programs. The program reads one sector at a time beginning with sector $F and places it into the memory range beginning at $2000. The sector is then checked before the next one is read from the disk.

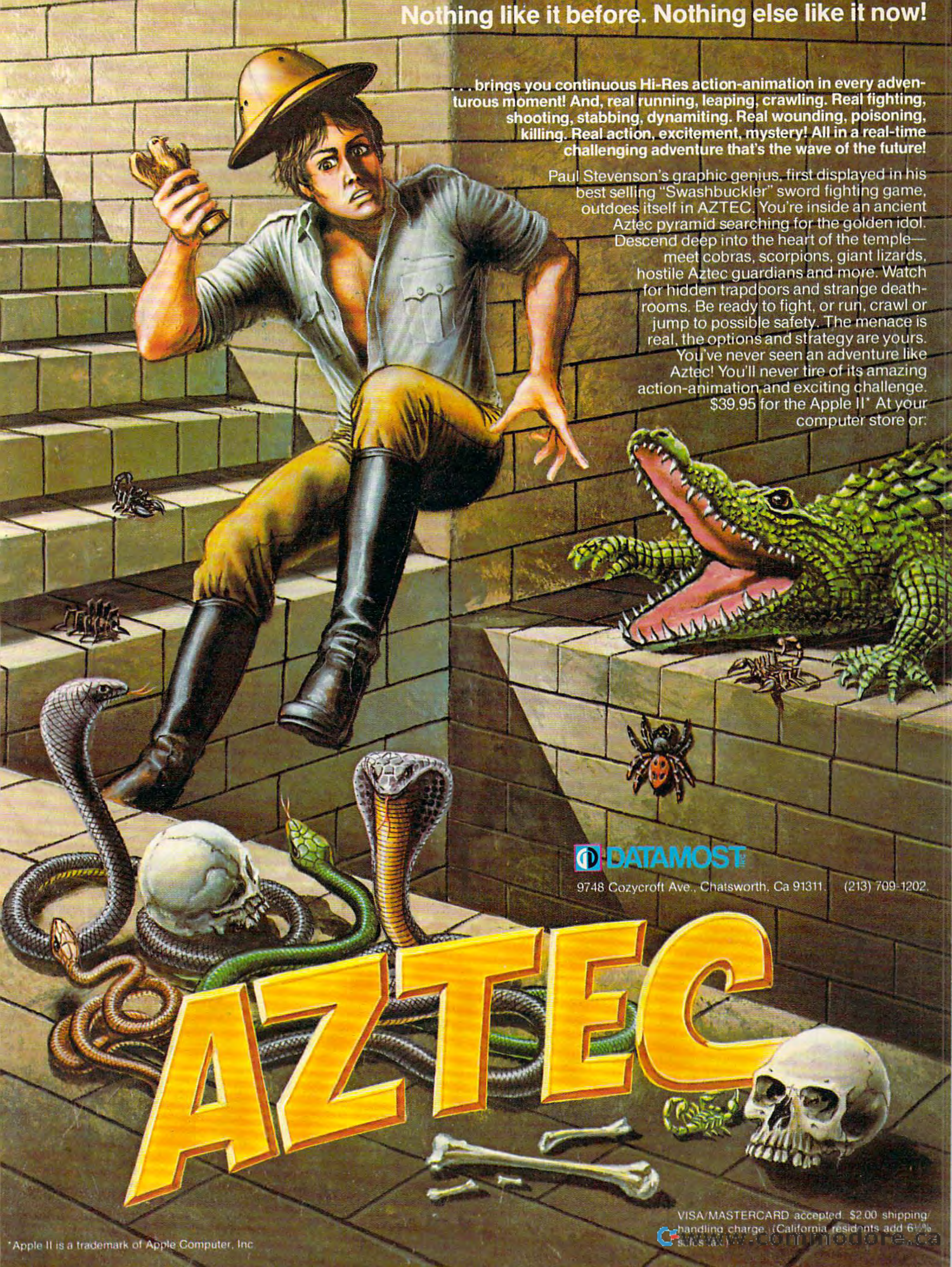
The seven program titles are checked for two things. The first is to see if the program is current. When a program is deleted, its catalog entry is not erased, but instead the first byte of the listing becomes $FF. If the program is current, then the bit containing the length is added to the running total. The Least Significant Byte of the total is stored in location $6074, or 24692. If there is a carry, it is placed in $6075. When all seven listings have been checked, the next sector is loaded, and the process continues until all fifteen sectors are read. The computer then returns to BASIC.

The rest of the BASIC program PEEKs the two locations containing the sum of the sectors used. The Most Significant Byte is multiplied by 256 and added to the LSB. This total is displayed and then subtracted from 496 to give the figure for the space remaining.

The program reads the DATA lines and enters the numbers beginning at $6000. It displays the CATALOG, then gives the results of the count. It can be substituted for your HELLO program, with your own personal touches like name and date placed before or after line 20, or after line 110.

```
10 D$ =  CHR$ (4)
20  PRINT D$;"CATALOG"
30  FOR K = 24576 TO 24694
40  READ A
50  POKE K,A
60  NEXT
70  CALL 24576
80 X =  PEEK (24692):Y =  PEEK (24693)
90 Z = X + 256 * Y
100  PRINT : PRINT "SECTORS USED: ";Z
110  PRINT "SECTORS REMAINING: ";496 - Z
120  END
130  DATA   169, 96, 160, 76, 32, 217, 3, 17
     3, 11, 32
140  DATA   201, 255, 240, 3, 32, 99, 96, 23
     8, 118, 96
150  DATA   173, 118, 96, 201, 7, 208, 28, 1
     69, 0, 141
160  DATA   118, 96, 169, 44, 141, 104, 96,
     169, 11, 141
170  DATA   8, 96, 173, 2, 32, 201, 0, 240,
     25, 141
180  DATA   81, 96, 76, 0, 96, 173, 104, 96,
      105, 35
190  DATA   141, 104, 96, 173, 8, 96, 105, 3
     5, 141, 8
200  DATA   96, 76, 7, 96, 96, 0, 1, 96, 1,
     0
210  DATA   17, 15, 94, 96, 0, 32, 0, 0, 1,
     0
220  DATA   0, 96, 1, 0, 0, 1, 239, 216, 0,
     24
230  DATA   173, 116, 96, 109, 44, 32, 141,
     116, 96, 144
240  DATA    3, 238, 117, 96, 96, 0, 0, 0, 0
```

©


Nothing like it before. Nothing else like it now!
. . . brings you continuous Hi-Res action-animation in every adventurous moment! And, real running, leaping, crawling. Real fighting, shooting, stabbing, dynamiting. Real wounding, poisoning, killing. Real action, excitement, mystery! All in a real-time challenging adventure that's the wave of the future!
Paul Stevenson's graphic genius, first displayed in his best selling "Swashbuckler" sword fighting game, outdoes itself in AZTEC. You're inside an ancient Aztec pyramid searching for the golden idol. Descend deep into the heart of the temple— meet cobras, scorpions, giant lizards, hostile Aztec guardians and more. Watch for hidden trapdoors and strange deathrooms. Be ready to fight, or run, crawl or jump to possible safety. The menace is real, the options and strategy are yours. You've never seen an adventure like Aztec! You'll never tire of its amazing action-animation and exciting challenge. $39.95 for the Apple II* At your computer store or:
DATAMOST INC.
9748 Cozycroft Ave., Chatsworth, Ca 91311. (213) 709-1202.
AZTEC
VISA/MASTERCARD accepted. $2.00 shipping/handling charge. (California residents add 6½%
*Apple II is a trademark of Apple Computer, Inc.
www.commodore.ca

Juggler

Doug Ferguson, Elida, OH

For Atari and VIC (with or without expansion), this game will challenge anyone's dexterity. If you can score 50 with two balls or 500 with three, there is an extra surprise.

"Juggler" is a fast-action game I wrote in September 1981 when the idea of programming BASIC was very new to me. It has undergone countless revisions since then, but the main loop (lines 860-970) remains what it was the first night I struggled.

The purpose of Juggler is to use the joystick to control the arms of a cartoon juggler in his attempt to keep two or three balls in the air. The juggler's hands move in tandem and can be positioned to catch the three colored balls (inside, middle, and outside). Since this game is not for the timid, it requires a joystick because only game addicts know how to work them.

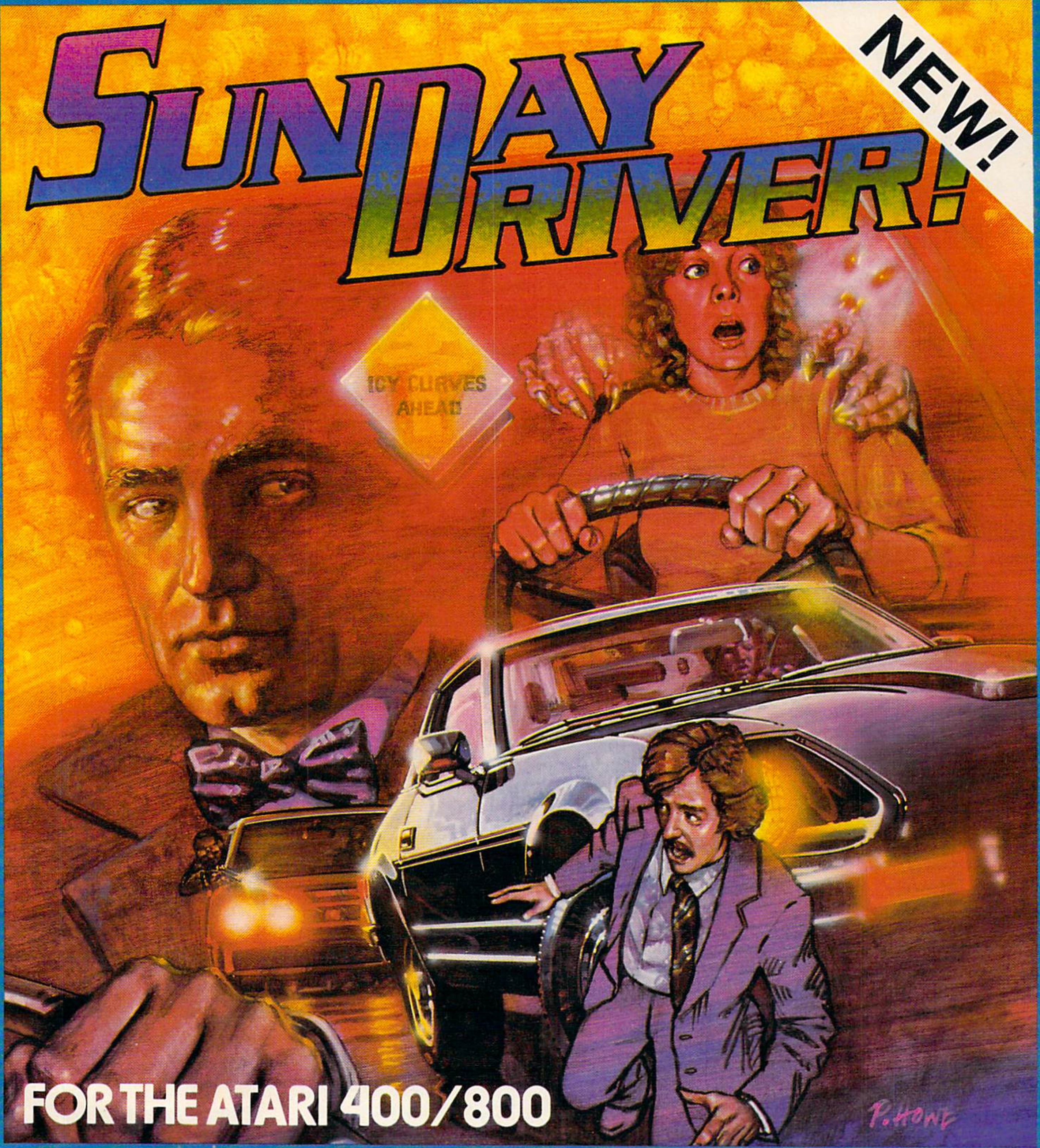
The balls are as random as I could make them and, contrary to the opinion of novice players, cannot land at the catching stage at the same time. The most important playing strategy is to make sure you hear the ball being caught before you move the hand toward another catch. The game can be slowed down merely by adding a few extra GOSUB 150 statements in the main loop, preferably at lines 860, 900, and 940.

The game will run on either the unexpanded or expanded VIC-20. I would rather not explain how it all works, mainly because it often follows inconsistent logic and layout. One technical footnote for those curious enough to investigate the program: instead of using zeros as values, I used a period (as in Y= .) because I read somewhere that this is faster and still zeros out the variable.

To start a game, push the joystick up or left for the three-ball or two-ball games, respectively. To repeat the same game, press the fire-button, or move the stick to change games. If you want to quit, hit Q.

Program 1: VIC-20 Version

```
100 POKE36879,75:X=RND(-TI):Y=2:GOT
    O670
110 POKEV,15:POKEV-2,N:FORT=1TO10:N
    EXT:POKEV-2,0
120 O=O+1+ABS(C)*9
130 PRINTTAB(7)"{WHT}SCORE "O"{HOME
    HOME}";:RETURN
140 DEFFNJ(X)=-((PEEK(37151)ANDX)=.
    ):RETURN
150 IFFNJ(16)ANDU=1THENGOSUB360
160 IFFNJ(16)ANDU=2THENGOSUB420
170 POKEG+2,127:Q=PEEK(G):POKEG+2,2
    55
180 IFQ=119ANDU=1THENGOSUB480
190 IFQ=119ANDU=.THENGOSUB420
200 RETURN
210 IFPEEK(H)<64THEN330
220 RETURN
230 IFPEEK(H+2)<64THEN340
240 RETURN
250 IFPEEK(H+4)<64THEN350
260 RETURN
270 IFPEEK(H+16)<64THEN330
280 RETURN
290 IFPEEK(H+14)<64THEN340
300 RETURN
310 IFPEEK(H+12)<64THEN350
320 RETURN
330 POKEJ%(K-A),32:D=K-1:GOSUB630:G
    OTO540
340 POKEU%(L-B),32:D=L:GOTO540
350 POKEG%(M-C),32:D=M+1:GOSUB640:G
    OTO540
360 POKEH+2,32:POKEH+3,32:POKEH-1,7
    4:POKEH,64:POKEH+1,75
370 POKEH+4,32:POKEH+5,32:POKEH+24,
    32:POKEH+23,77:POKEH+25,32
```



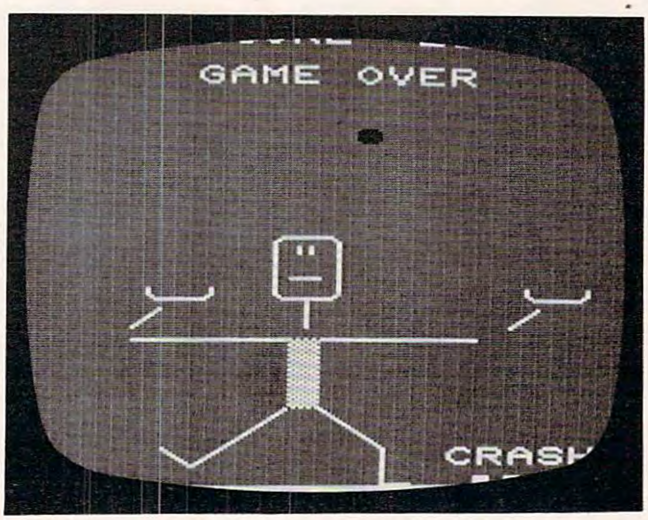
SUNDAY DRIVER gives you four scenarios to choose from. You must beat the clock as you drive along while avoiding pedestrians, other cars, and obstacles. In other versions it's winter and you're on ice-slicked roads. In game three it's nighttime (don't hit the ghosts). If this sounds too easy try the 007 option — it's you against **them** on twisty roads.

SEND CHECK OR MONEY ORDER TO:

ANALOG Software
P.O. Box 23 Worcester, MA 01603
MasterCard and VISA accepted (617)892-3488
16K cassette or disk $29.95.

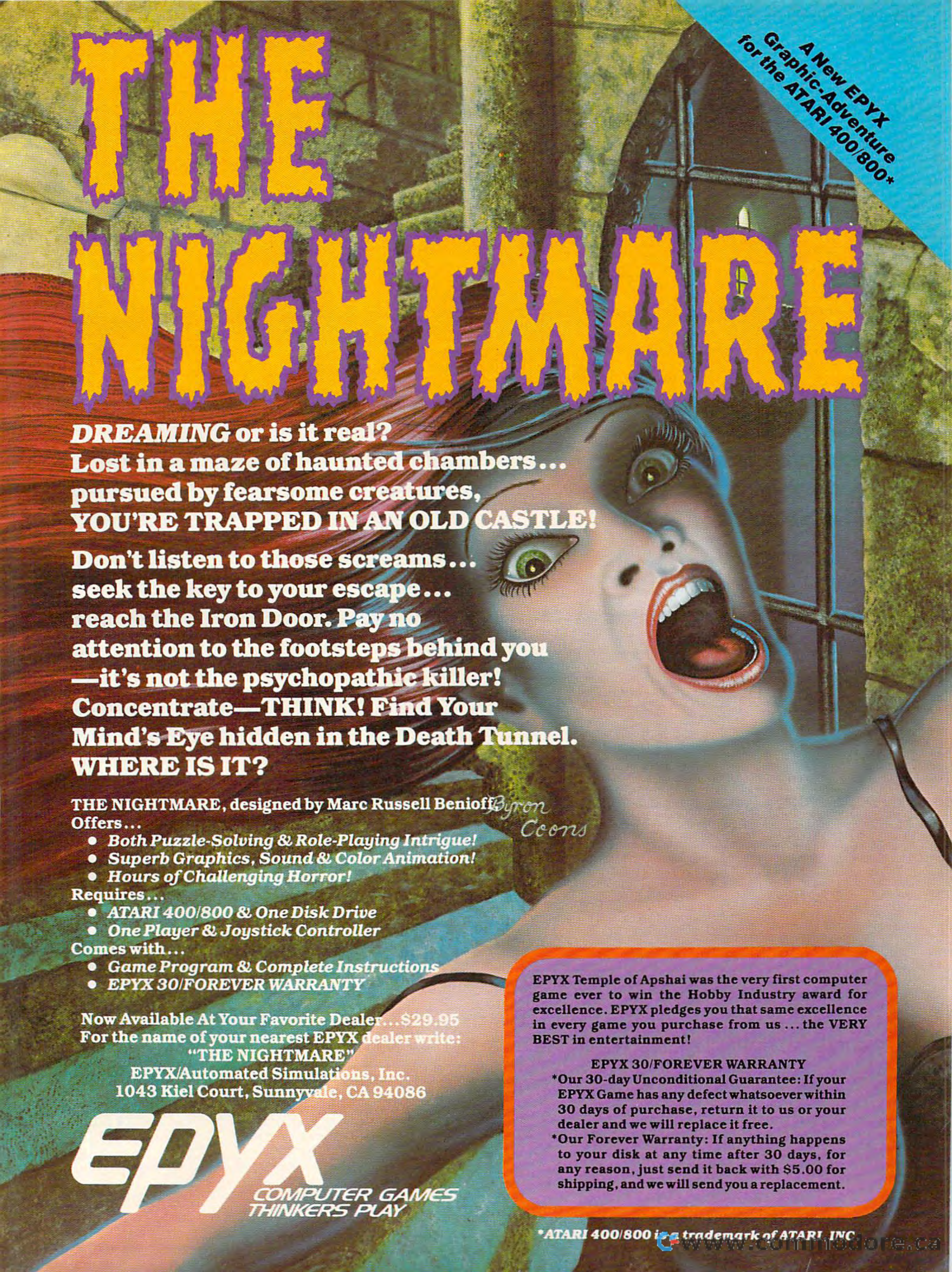
Dealer inquiries invited.

```
380 POKEH+14,32:POKEH+15,32:POKEH+1
    1,74:POKEH+12,64:POKEH+13,
    75
390 POKEH+16,32:POKEH+17,32:POKEH+3
    7,32:POKEH+35,77:POKEH+36,
    32
400 POKEH+187,77:POKEH+188,78:POKEH
    +209,32:POKEH+181,103:POKE
    H+203,122:POKEH+180,32:U=.
410 RETURN
420 POKEH+1,74:POKEH+2,64:POKEH+3,7
    5:POKEH,32:POKEH-1,32
430 POKEH+4,32:POKEH+5,32:POKEH+25,
    32:POKEH+23,32:POKEH+24,72
440 POKEH+13,74:POKEH+14,64:POKEH+1
    5,75:POKEH+12,32:POKEH+11,
    32
450 POKEH+16,32:POKEH+17,32:POKEH+3
    7,32:POKEH+35,32:POKEH+36,
    72
460 POKEH+181,103:POKEH+203,122:POK
    EH+187,101:POKEH+209,76:PO
    KEH+188,32:POKEH+180,32:U=
    1
470 RETURN
```



A fumbled ball crashes in "Juggler," VIC-20 version.

```
480 POKEH+3,74:POKEH+4,64:POKEH+5,7
    5:POKEH+2,32:POKEH+1,32
490 POKEH,32:POKEH-1,32:POKEH+25,78
    :POKEH+23,32:POKEH+24,32
500 POKEH+15,74:POKEH+16,64:POKEH+1
    7,75:POKEH+14,32:POKEH+13,
    32
510 POKEH+12,32:POKEH+11,32:POKEH+3
    7,78:POKEH+35,32:POKEH+36,
    32
520 POKEH+181,78:POKEH+180,77:POKEH
    +203,32:POKEH+187,101:POKE
    H+209,76:POKEH+188,32:U=2
530 RETURN
540 GOSUB650:PRINTTAB(D)"{21 DOWN}{
    WHT}CRASH{HOME}";:POKEV-1,
    N:POKE7954-SC,15:POKEH-36,
    34
550 FORT=15TOØSTEP-1:POKEV,T:POKEV+
    1,PEEK(V+1)AND248ORT
560 FORW=1TO100:NEXT:NEXT:POKEV-1,.
    :POKE36879,75
570 PRINTTAB(7)"{YEL}{Ø2 DOWN}GAME ~
    OVER{DOWN}":POKEH-14,64:GO
    TO59Ø
580 PRINT"      {BLK}↑{YEL} 3 BALLS
    DOWN}":PRINT"     {BLK}←
    YEL} 2 BALLS{DOWN}":PRINT"
        PRESS {REV}Q{OFF} TO Q
    UIT":GOSUB14Ø
590 GETA$:IFA$="Q"THENSYS65234
595 IFFNJ(32)ANDY<>2THENRESTORE:E=Ø
    :O=Ø:GOTO71Ø
600 IFFNJ(16)THENCLR:Y=1:GOTO7ØØ
610 IFFNJ(4)THENCLR:GOTO7ØØ
620 GOTO59Ø
630 FORT=38884TOT+6:POKET-CO,7:NEXT
    :FORT=389ØØTOT+6:POKET-CO,
    7:NEXT:RETURN
640 FORT=38884TOT+6:POKET-CO,6:NEXT
    :FORT=38899TOT+6:POKET-CO,
    6:NEXT:RETURN
650 FORT=8165+DTOT+2:POKET-SC,123:N
    EXT
660 RETURN
670 PRINT"{CLEAR}{DOWN}        {YEL}V
    IC JUGGLER{WHT}"
680 PRINT"{Ø3 DOWN}{Ø3 RIGHT}USE JO
    YSTICK ONLY{Ø4 DOWN}"
690 PRINT"{YEL}{Ø4 RIGHT}CHOOSE GAM
    E{DOWN}":GOTO58Ø
700 V=36878:H=7968:G=37152:DIMJ%(18
    ),U%(16),G%(13)
710 PRINT"{CLEAR}";:IFPEEK(36869)=1
    92THENSC=3584:CO=512
720 H=7968-SC:GOSUB14Ø
730 FORX=8Ø15TO8Ø25:POKEX-SC,64:NEX
    T:POKEX-6-SC,1Ø4
740 POKEH-37,93:POKEH-35,93:POKEH+7
    ,74:POKEH+9,75:POKEH+3Ø,93
    :POKEH+118,1Ø2
750 POKEH-15,93:POKEH-13,93:POKEH-5
    9,85:POKEH-57,73
760 POKEH-36,114:POKEH-14,82:POKEH+
    8,114:POKEH-58,64:POKEH+74
    ,1Ø2
770 POKEH+139,78:POKEH+141,77:POKEH
    +16Ø,78:POKEH+164,77:POKEH
    +96,1Ø2
780 GOSUB36Ø
790 FORK=2TO16:READJ%(K):J%(K)=J%(K
    )-SC:POKEJ%(K)+3Ø72Ø+SC-CO
```



A New EPYX Graphic-Adventure for the ATARI 400/800*
THE NIGHTMARE
DREAMING or is it real?
Lost in a maze of haunted chambers...
pursued by fearsome creatures,
YOU'RE TRAPPED IN AN OLD CASTLE!
Don't listen to those screams...
seek the key to your escape...
reach the Iron Door. Pay no
attention to the footsteps behind you
—it's not the psychopathic killer!
Concentrate—THINK! Find Your
Mind's Eye hidden in the Death Tunnel.
WHERE IS IT?
THE NIGHTMARE, designed by Marc Russell Benioff
Byron Coons
Offers...
• Both Puzzle-Solving & Role-Playing Intrigue!
• Superb Graphics, Sound & Color Animation!
• Hours of Challenging Horror!
Requires...
• ATARI 400/800 & One Disk Drive
• One Player & Joystick Controller
Comes with...
• Game Program & Complete Instructions
• EPYX 30/FOREVER WARRANTY
Now Available At Your Favorite Dealer...$29.95
For the name of your nearest EPYX dealer write:
"THE NIGHTMARE"
EPYX/Automated Simulations, Inc.
1043 Kiel Court, Sunnyvale, CA 94086
EPYX
COMPUTER GAMES
THINKERS PLAY
EPYX Temple of Apshai was the very first computer game ever to win the Hobby Industry award for excellence. EPYX pledges you that same excellence in every game you purchase from us ... the VERY BEST in entertainment!
EPYX 30/FOREVER WARRANTY
*Our 30-day Unconditional Guarantee: If your EPYX Game has any defect whatsoever within 30 days of purchase, return it to us or your dealer and we will replace it free.
*Our Forever Warranty: If anything happens to your disk at any time after 30 days, for any reason, just send it back with $5.00 for shipping, and we will send you a replacement.
*ATARI 400/800 is a trademark of ATARI, INC.

```
    ,7:NEXT
800 FORK=2TO14:READU%(K):U%(K)=U%(K
    )-SC:NEXT
810 FORK=2TO12:READG%(K):G%(K)=G%(K
    )-SC:POKEG%(K)+30720+SC-CO
    ,6:NEXT
820 K=INT(RND(1)*8)+2:A=1
830 L=INT(RND(1)*7)+2:B=1
840 IFY=0THENM=INT(RND(1)*6)+2:C=1
850 IFO>49+450*ABS(C)THENE=1:Y=.
860 GOSUB150
870 N=220:IFK=17THENGOSUB270:A=-1:G
    OSUB110:K=15-E
880 IFK=1THENGOSUB210:A=1:GOSUB110:
    K=3+E
890 POKEJ%(K-A),32:POKEJ%(K),81:K=K
    +A
900 GOSUB150
910 IFL=15THENGOSUB290:B=-1:GOSUB11
    0:L=13-E
920 IFL=1THENGOSUB230:B=1:GOSUB110:
    L=3+E
930 POKEU%(L-B),32:POKEU%(L),81:L=L
    +B
940 GOSUB150
950 IFM=13THENGOSUB310:C=-1:GOSUB11
    0:M=11-E
960 IFM=1THENGOSUB250:C=1:GOSUB110:
    M=3+E
970 POKEG%(M-C),32:POKEG%(M),81:M=M
    +C:IFE=.THEN850
980 POKEJ%(K-3*A),32:POKEJ%(K-2*A),
    81
990 POKEU%(L-3*B),32:POKEU%(L-2*B),
    81
1010 POKEG%(M-3*C),32:POKEG%(M-2*C),
    81:GOTO860
1020 DATA 7946,7902,7858,7815,7772,7
    730,7710,7712,7714,7738,77
    84,7829,7874,7918,7962
1030 DATA7948,7904,7860,7817,7774,77
    54,7756,7758,7782,7827,787
    2,7916,7960
1040 DATA7950,7906,7862,7819,7798,77
    78,7802,7825,7870,7914,7958
```

Program 2: Atari Version

```
100 REM JUGGLER - ATARI VERSION
110 GRAPHICS 18
120 GOSUB 660:GRAPHICS 18:POKE 756,CH
    SET/256:SETCOLOR 1,12,6:SETCOLOR
    3,4,8:SETCOLOR 2,0,10
130 SCR=PEEK(88)+256*PEEK(89)
140 POSITION 7,6:? #6;"  !"
150 POSITION 7,7:? #6;"<{H}{B}{D}>"
160 POSITION 7,8:? #6;"  {C}"
170 POSITION 7,9:? #6;" $ %"
180 BALLS=2:DIM BALL$(3):BALL$="G{E}
    {G}":POSITION 6,0:? #6;M$
190 GOSUB 320:POSITION 1,1:? #6;"ball
    s:";BALLS
200 POSITION 10,1:? #6;"{8 SPACES}":I
    F PRACTICE THEN POSITION 10,1:? #
    6;"PRACTICE"
210 K=PEEK(53279):IF K=7 THEN 210
220 IF PEEK(53279)=K THEN 220
230 IF K=5 THEN BALLS=5-BALLS
240 IF K=3 THEN PRACTICE=1-PRACTICE
```

Atari Notes

Use your joystick to move the Juggler's arms to any of three positions (far left, center, or far right) to catch and deflect the balls. You can select a game with either two balls or three balls by pressing SELECT when the game is RUN. If you press OPTION, you can play a "practice" game. In a practice game, you cannot make any points, but you can't lose, either. It's a good way to learn how to play without becoming frustrated.

A successful catch is greeted with a "bleep," but a miss gets you a raspberry. If you're not playing a practice game, it's all over when you miss a ball. The rest come tumbling down!

A note on strategy: let go of the joystick after each move. This will return the Juggler to the center position, where it is easiest to quickly react.

```
250 IF K<>6 THEN 190
260 POSITION 1,1:? #6;"{9 SPACES}"
270 GOSUB 1190
280 REM MAIN LOOP
290 GOSUB 320:GOSUB 320
300 GOSUB 440:GOSUB 320
310 GOTO 290
320 REM MOVE ARMS
330 IF PEEK(53279)=6 THEN RUN
340 POS=2-(1-PTRIG(0))+(1-PTRIG(1))
350 ON POS GOSUB 370,390,410
360 RETURN
370 POSITION 6,7:? #6;"<{H}{D}{B}>   "
    :POKE 77,0
380 POSITION 8,9:? #6;"$ +":RETURN
390 POSITION 6,7:? #6;" <{H}{B}{D}> "
400 POSITION 8,9:? #6;"$ %":RETURN
410 POSITION 6,7:? #6;"   <{B}{D}{H}>"
    :POKE 77,0
420 POSITION 8,9:? #6;"* %":RETURN
430 REM MOVE BALL
440 INDEX=(INDEX+1)*(INDEX<BALLS-1)
450 BPOS=BPOS(INDEX)
460 WHICH=WHICH(INDEX):BDIR=BDIR(INDE
    X):BIN=BIN(INDEX)
470 POKE BPOS,0
480 BPOS=BPOS+PB(WHICH,BIN)*BDIR
490 POKE BPOS,ASC(BALL$(INDEX+1))
500 BIN=BIN+BDIR:IF BIN=0 OR BIN>PB(W
    HICH,0) THEN 530
```

For the Atari 400/800 Home Computer

You are Sentinel I, the latest in highly maneuverable strike aircraft, and you have a mission, to protect the metropolis, but the alien attack will stop at nothing to destroy your very last lines of defense. Your senses are tuned for battle and the attack begins.

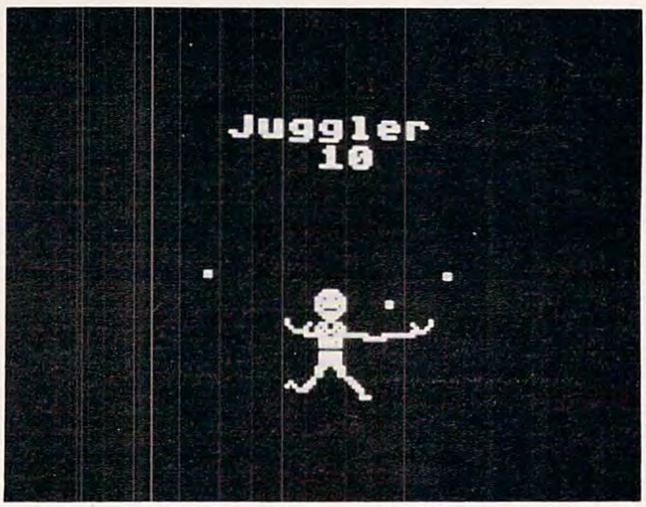
Aliens will block your path, destroy your ship, deplete your fuel and sacrifice their lives to stop your mission. You must destroy the aliens with your rapid fire lasers before they home in and destroy you. There is no escape — you must destroy them all for they will stop at nothing.

- 100% machine language
- 1 or 2 player option
- joy stick controls
- lateral scrolling screen
- superb graphics
- extensive color
- finest sound utilization
- available in 16K tape $29.95 U.S. funds
- 24K disc $34.95 U.S. funds
- call your local dealer for more information

INHOME SOFTWARE

Inhome Software Incorporated 2485 Dunwin Drive, Mississauga, Ontario L5L 1T1. (416) 828-0775.

```
510 BIN(INDEX)=BIN
520 BPOS(INDEX)=BPOS:RETURN
530 REM END OF TRAVEL...MISSED?
540 P=PEEK(BPOS+20):IF P<>28 AND P<>3
    0 THEN 580
550 FOR W=14 TO 0 STEP -2:SOUND 0,50,
    10,W:SOUND 1,60,10,W:NEXT W
560 IF  NOT PRACTICE THEN SCORE=SCORE
    +1+9*(BALLS=3):POSITION 10-LEN(ST
    R$(SCORE))/2,1:? #6;SCORE;" "
570 POKE BPOS,0:BDIR(INDEX)=-BDIR(IND
    EX):GOSUB 1260:RETURN
580 SOUND 0,100,12,8:FOR W=1 TO 50:NE
    XT W:SOUND 0,0,0,0:IF PRACTICE TH
    EN 570
590 FOR I=1 TO 10:FOR J=0 TO BALLS-1
600 POKE BPOS(J),ASC(BALL$(J+1))
610 BPOS(J)=BPOS(J)+20-BDIR(J):SOUND
    0,I*BALLS+J,12,10-I:NEXT J:NEXT I
620 POSITION 9,6:? #6;"[illegible]"
630 POSITION 5,0:? #6;"press START"
640 IF PEEK(53279)<>6 THEN 640
650 RUN
660 REM INITIALIZATION
```



Reaching with an outstretched arm in the Atari version of "Juggler."

```
670 GRAPHICS 2+16:DIM M$(10)
680 M$="JUGgLEr"
690 FOR I=1 TO LEN(M$):A=ASC(M$(I)):P
    OKE 712,INT(A/16)*16+14
700 FOR J=1 TO 11:COLOR 32:PLOT I+J-1
    ,J-1:COLOR A:PLOT I+J,J:SOUND 0,J
    +I*10,10,8:NEXT J:COLOR 32:PLOT I
    +J-1,J-1
710 FOR J=10 TO 6 STEP -1:COLOR 32:PL
    OT I+J/2+3.5,J+1:COLOR A:PLOT I+3
    +J/2,J:SOUND 0,J+I*10,10,8:NEXT J
720 NEXT I
730 FOR I=0 TO 240 STEP 10:POKE 712,0
740 A=PEEK(708):POKE 708,PEEK(709):PO
    KE 709,PEEK(710):POKE 710,PEEK(71
    1):POKE 711,A
750 POKE 711,PEEK(53770):POKE 53279,0
760 SOUND 0,I,10,4:SOUND 1,I+10,10,4:
    NEXT I
770 SOUND 0,0,0,0:SOUND 1,0,0,0
780 GRAPHICS 2+16:POSITION 7,6:? #6;M
    $
790 CHSET=(PEEK(106)-8)*256:FOR I=0 T
    O 7:POKE CHSET+I,0:NEXT I
800 RESTORE 840:IF PEEK(CHSET+8)=60 T
    HEN 1030
810 FOR I=128 TO 207:POKE CHSET+I,PEE
    K(57344+I):SOUND 0,I,10,8:POKE 53
    274,I:NEXT I
820 FOR I=1 TO 14:READ A,B:FOR J=0 TO
     7:POKE CHSET+A*8+J,PEEK(57344+B*
    8+J):SOUND 0,A,10,8:SOUND 1,B,10,
    8
830 POKE 53274,A:NEXT J:NEXT I:SOUND
    0,0,0,0:SOUND 1,0,0,0
840 DATA 34,34,33,97,44,108,51,115,26
    ,26,42,42,53,117,39,103,37,101,50
    ,114,48,48,35,99,52,116,41,105
850 READ A:IF A=-1 THEN SOUND 0,0,0,0
    :GOTO 1030
860 FOR J=0 TO 7:READ B:POKE CHSET+A*
    8+J,B:SOUND 0,B,10,8:POKE 53274,B
    :NEXT J
870 GOTO 850
880 DATA 1,60,126,219,255,189,195,126
    ,60
890 DATA 2,24,102,165,219,231,126,122
    ,126
900 DATA 3,0,126,126,126,126,231,195,
    195
910 DATA 4,3,6,12,24,16,112,0,0
920 DATA 5,192,96,48,24,8,14,0,0
930 DATA 6,60,90,219,255,231,231,126,
    60
940 DATA 7,0,0,0,24,24,0,0,0
950 DATA 8,0,0,0,0,63,224,0,0
960 DATA 9,0,0,0,0,252,7,0,0
970 DATA 10,1,67,166,28,0,0,0,0
980 DATA 11,128,194,101,56,0,0,0,0
990 DATA 28,128,130,68,60,7,0,0,0
1000 DATA 30,1,33,18,60,224,0,0,0
1010 DATA 127,16,24,28,30,30,28,24,16
1020 DATA -1
1030 REM MORE INITALIZATION
1040 REM Read in parabolas
1050 DIM PB(4,15):RESTORE 1060
1060 DATA 0,-19,-20,-21,-21,-1,19,19,
     20,20,0,-99
1070 DATA 0,-20,-21,-21,-21,-1,-1,19,
     20,21,21,0,-99
1080 DATA 0,-19,-20,-21,-21,-1,-1,19,
     19,20,21,0,-99
1090 DATA 0,-19,-20,-20,-21,-1,19,19,
     19,20,0,-99
1100 DATA 0,-20,-21,-21,19,19,20,0,-9
     9
1110 FOR I=0 TO 4
1120 FOR J=1 TO 15
1130 READ A:IF A<>-99 THEN PB(I,J)=A:
     NEXT J
1140 PB(I,0)=J-1
1150 NEXT I
1160 DIM BPOS(2),BPEEK(2),WHICH(2)
1170 DIM BDIR(2),BIN(2):RETURN
1180 REM INITIALIZE BALLS
1190 FOR I=0 TO 2:WHICH(I)=-1:NEXT I
1200 FOR I=0 TO BALLS-1
1210 BDIR(I)=-1+2*(RND(1)>0.5)
1220 INDEX=I:GOSUB 1260
1230 NEXT I
1240 RETURN
1250 REM INITIALIZE ONE BALL
1260 WHICH=INT(5*RND(0)):WHICH(INDEX)
     =-1
1270 BPOS(INDEX)=SCR+130+INT(3*RND(0)
     )-4*(BDIR(INDEX)=-1)
1280 FOR J=0 TO BALLS-1
1290 IF WHICH(J)=WHICH THEN J=BALLS:N
     EXT J:GOTO 1260
1300 NEXT J:WHICH(INDEX)=WHICH
1310 BIN(INDEX)=1+(PB(WHICH,0)-1)*(BD
     IR(INDEX)=-1)
1320 RETURN
```

©

...and so there were keys for the Atari 400.

In the beginning there was the membrane keyboard.

So it was to be done that Inhome Software would create a full-stroke keyboard for the Atari 400 Home Computer and it would be called the B Key 400, and would sell for $119.95 U.S. funds.

The new B Key 400 was made so easy to install that the owner could do it himself in a miraculous two minutes.

With the B Key 400 keyboard from Inhome Software, you will follow into the land of professional home computers that are powerful, easy to program and have a great capacity that can be made even greater with Inhome Software 48K and 32K memory boards. It was done and it was good.

INHOME SOFTWARE

Inhome Software Incorporated, 2485 Dunwin Drive, Mississauga, Ontario L5L 1T1. (416) 828-0775. Made in Canada.

KMMM Pascal for PET/CBM $85

A subset of standard Pascal with extensions.
- Machine language Pascal Source Editor with cursor oriented window mode
- Machine Language P-Code Compiler
- P-Code to machine language translator for optimized object code
- Run-time package
- Floating point capability
- User manual and sample programs

Requires 32K Please specify configuration.

EARL for PET (disk file based) $65

Editor, Assembler, Relocator, Linker

Generates relocatable object code using MOS Technology mnemonics. Disk file input (can edit files larger than memory). Links ,multiple object programs as one memory load. Listing output to screen or printer. Enhanced editor operates in both command mode and cursor oriented "window" mode.

RAM/ROM for PET/CBM

4K or 8K bytes of soft ROM with optional battery backup.

RAM/ROM is compatible with any large keyboard machine. Plugs into one of the ROM sockets above screen memory to give you switch selected write protectable RAM.

Use RAM/ROM as a software development tool to store data or machine code beyond the normal BASIC range. Use RAM/ROM TO LOAD A ROM image where you have possible conflicts with more than one ROM requiring the same socket. Possible applications include machine language sort (such as SUPERSORT), universal wedge, Extramon, etc.

RAM/ROM – 4K	$75
RAM/ROM – 8K	90
Battery Backup Option	20

SUBSORT by James Strasma $35

Subsort is an excellent general purpose machine language sort routine for PET/CBM computers. Sorts both one and two dimensioned arrays at lightning speed in either ascending or descending order. Other fields can be subsorted when a match is found, and fields need not be in any special order. Sort arrays may be specified by name, and fields are random length. Allows sorting by bit to provide 8 categories per byte. The routine works with all PET BASICs, adjusts to any memory size, and can co-exist with other programs in high memory.

SuperGraphics 2.0

NEW Version with TURTLE GRAPHICS

SuperGraphics, by John Fluharty, provides a 4k machine language extension which adds 35 full featured commands to Commodore BASIC to allow fast and easy plotting and manipulation of graphics on the PET/CBM video display, as well as SOUND Commands. Animations which previously were too slow or impossible without machine language subroutines now can be programmed directly in BASIC. Move blocks (or rocketships, etc.), or entire areas of the screen with a single, easy to use BASIC command. Scroll any portion of the screen up, down, left, or right. Turn on or off any of the 4000 (8000 on 8032) screen pixels with a single BASIC command. In high resolution mode, draw vertical, horizontal, and diagonal lines. Draw a box, fill a box, and move it around on the screen with easy to use BASIC commands. Plot curves using either rectangular or polar co-ordinates (great for Algebra, Geometry and Trig classes.)

The SOUND commands allow you to initiate a note or series of notes (or even several songs) from BASIC, and then play them in the background mode without interfering with your BASIC program. This allows your program to run at full speed with simultaneous graphics and music.

Seven new TURTLE commands open up a whole new dimension in graphics. Place the TURTLE anywhere on the screen, set his DIRECTION, turn him LEFT or RIGHT, move him FORWARD, raise or lower his plotting pen, even flip the pen over to erase. Turtle commands use angles measured in degrees, not radians, so even elementary school children can create fantastic graphic displays.

Specify machine model (and size), ROM type (BASIC 3 or 4)

SuperGraphics in ROM $45

Volume discounts available on ROM version for schools.

FLEX-FILE NEW VERSION II

for PET/CBM Computers

FLEX-FILE is a set of flexible, friendly programs to allow you to set up and maintain a data base. Includes versatile Report Writer and Mail Label routines, and documentation for programmers to use Data Base routines as part of other programs.

RANDOM ACCESS DATA BASE

Record size limit is 256 characters. The number of records per disk is limited only by record size and free space on the disk. File maintenance lets you step forward or backward through a file, add, delete, or change a record, go to a numbered record, or find a record by specified field (or partial field). Field lengths may vary to allow maximum information packing. Both subtotals and sorting may be nested up to 5 fields deep. Any field may be specified as a key. Sequential file input and output, as well as file output in WordPro and PaperMate format is supported. Record size, fields per record, and order of fields may be changed easily.

MAILING LABELS

Typical mail records may be packed 3000 per disk on 8050 (1400 on 4040). Labels may be printed any number wide, and may begin in any column position. There is no limit on the number or order of fields on a label, and complete record selection via type code or field condition is supported.

REPORT WRITER

Flexible printing format, including field placement, decimal justification and rounding. Define any column as a series of math or trig functions performed on other columns, and pass results such as running total from row to row. Totals, nested subtotals, and averages supported. Complete record selection, including field within range, pattern match, and logical functions can be specified.

FLEX-FILE II by Michael Riley $110

Please specify equipment configuration when ordering.

DISK I.C.U. $40

Intensive Care Unit by L.C. Cargile

COMPLETE DISK RECOVERY SYSTEM FOR CBM DRIVES
- edit disk blocks with ease
- duplicate disks, skipping over bad blocks
- complete diagnostic facilities
- un-scratch scratched files
- check and correct scrambled files
- recover improperly closed files
- extensive treatment of relative files
- optional output to IEEE488 printer
- comprehensive user manual (an excellent tutorial on disk operation and theory).

Furnished on copy-protected disk with manual.
Backup disk available, $10 additional.

PROGRAM YOUR OWN EPROMS $75

Branding Iron EPROM Programmer for PET/CBM software for all ROM versions. Includes all hardware and software to program or copy 2716 and 2532 EPROMs.

PORTMAKER DUAL RS232 SERIAL PORT $63

Two ports with full bipolar RS232 buffering. Baud rates from 300 to 4800. For PET/CBM, AIM, SYM.

CBM Software

TCL Pascal Version 1.6	135
Petspeed BASIC Compiler	225
Integer BASIC Compiler	110
CMAR Record Handler	110
UCSD Pascal (without board)	135
Wordcraft 80	300
BPI Accounting Modules	300
Professional Tax Prep Sys.	600
Intelligent Terminal Emulator	25
ASERT Data Base	375
Personal Tax Calc	55
Dow Jones Portfolio Mgmt.	110
Assembler Development	80
Legal Time Accounting	445

FORTH for PET

BY L. C. Cargile and Michael Riley $50

Features include:
- full FIG FORTH model.
- all FORTH 79 STANDARD extensions.
- structured 6502 Assembler with nested decision making macros.
- full screen editing (same as when programming in BASIC).
- auto repeat key.
- sample programs.
- standard size screens (16 lines by 64 characters).
- 150 screens per diskette on 4040, 480 screens on 8050.
- ability to read and write BASIC sequential files.
- introductory manual.
- reference manual.

Runs on any 16K or 32K PET/CBM (including 8032) with ROM 3 or 4, and CBM disk drive. Please specify configuration when ordering.

Metacompiler for FORTH $30

simple metacompiler for creating compacted object code which can be executed independently (without the FORTH system).

PaperMate 60 COMMAND WORD PROCESSOR

by Michael Riley

Paper-Mate is a full-featured word processor for CBM/PET by Michael Riley. Paper-Mate incorporates 60 commands to give you full screen editing with graphics for all 16K or 32K machines (including 8032), all printers, and disk or tape drives. Many additional features are available (including most capabilities of Professional Software's WordPro 3).

For writing text, Paper-Mate has a definable keyboard so you can use with either Business or Graphics machines. Shift lock on letters only, or use keyboard shift lock. All keys repeat.

Paper-Mate text editing includes floating cursor, scroll up or down, page forward or back, and repeating insert and delete keys. Text block handling includes transfer, delete, append, save, load, and insert.

All formatting commands are imbedded in text for complete control. Commands include margin control and release, column adjust, 9 tab settings, variable line spacing, justify text, center text, and auto print form letter (variable block). Files can be linked so that one command prints an entire manuscript. Auto page, page headers, page numbers, pause at end of page, and hyphenation pauses are included.

Unlike most word processors, CBM graphics as well as text can be used. Paper-Mate can send any ASCII code over any secondary address to any printer.

Paper-Mate functions with all CBM/PET machines with at least 16K, with any type of printer, and with either cassette or disk.

To order Paper-Mate, please specify machine and ROM type.

Paper-Mate (disk or tape) for PET, CBM, VIC, C64 **$40**

SM-KIT for PET/CBM $40

Enhanced ROM based utilities for BASIC 4. Includes both programming aids and disk handling commands.

BASIC INTERPRETER for CBM 8096 $200

A full interpreter implementation to automatically take advantage of the extra memory available with 8096.

PEDISK II Systems from cgrs Microtech available.
FILEX IBM 3741/2 Data Exchange Software available.

JINSAM Data Base Management System for CBM.
Comprehensive version available for most configurations.

COPY-WRITER Word Processor for PET/CBM. $159

Works like expensive word processors, plus has added features like double column printing and shorthand generator.

CASH MANAGEMENT SYSTEM $45

Easy to use disk system. Keeps track of cash disbursements, cash receipts, cash transfers, expenses for up to 50 categories.

252 Bethlehem Pike
Colmar, PA 18915 215-822-7727 **A B Computers**

WRITE FOR CATALOG

Add $1.25 per order for shipping. We pay balance of UPS surface charges on all prepaid orders. Prices listed are on cash discount basis. Regular prices slightly higher. Prices subject to change.

SOUND On The Sinclair/Timex

Arthur B. Hunkins
School of Music, UNCG, Greensboro, NC

Sound on the Sinclair/Timex? Beethoven symphonies, no; simple melodies, yes! All you need is one of the programs below, and a high-gain amplifier/speaker connected to the computer's mike output. (Radio Shack's battery-operated mini- or telephone-amplifier/speakers at $10-12 work fine.)

The software secret is a short Z-80 machine language (ML) routine buried in a REM statement at the beginning of a BASIC program. It is important that the routine be the FIRST program statement; if it isn't, all the USR and POKE addresses that follow must be changed. We'll make our opening REMark statement number 10, and leave it!

After typing 10 REM, press the following sequence of CHARACTERS (ignore commas, periods, and spaces): NEXT, A, /, I, =, :, COPY, INKEY$, PEEK, COPY, (, RETURN, INKEY$, <=, RETURN, (, RETURN, H, 4, LET, 9, 4, GOTO, TAN. Remember, these are *Characters – single keystrokes.* Spaces will appear in the display, but you don't type any in.

Some of the characters are FUNCTIONS; to register them, you first hit the function key. Others are KEYWORDS, a bit trickier. First hit THEN, which causes the K cursor to appear; then press the keyword; finally go back and delete THEN. Presto, a keyword in a REM statement! When you are finished entering characters, be sure to hit ENTER.

Now we'll add a second statement: 30 LET A=USR 16514. Attach the amplifier to the mike output, turn up the volume, and we're ready for a test. (You might want to save the program first – machine language crashes make you start over.) Be sure you are in FAST mode; sound doesn't work in SLOW. (In your own applications, you may switch back and forth from SLOW to FAST whenever you wish.)

RUN the short program. If all is well, you should hear a slightly low B above middle C for approximately one second. The screen display goes berserk during the note, showing horizontal black streaks similar to a LOAD. When the sound is finished, the previous display returns.

Note that the computer hum, which is quite audible except during the note itself, is caused by the screen display. When the screen is "off," including black during computer calculations, there is no hum. This program produces one note per subroutine call – a square wave, limited in range to pitches from B above middle C on up. (More sophisticated routines that extend the range into the bass register, and permit a variety of tone colors, are discussed later.)

Frequency And Duration

Let's construct a simple BASIC program to play our choice and length of note:

```
10 REM (as above)
12 LET B=255
14 LET D=5
17 POKE 16520,B
22 POKE 16518,D*1000/B
30 LET A=USR 16514
```

B is the frequency value, POKEd into location 16520; D is the duration value, which – converted to be constant for varying frequencies – is placed in 16518. B must be between 1 and 255, and D must be greater than 0. For higher frequencies, it is possible to ask for too long a duration; one may receive a report code of B in statement 22 here (an attempt to POKE a number larger than 255 into location 16518).

For longer tones, adjust tempo location 16516, which normally contains value 24. POKEing a smaller value speeds up the tempo, while a larger value makes everything last longer. Note that these three memory locations – for frequency, duration, and tempo – are the same for all programs in this article. So is the USR subroutine call address.

The single-byte frequency values for pitches from the B above middle C up through two octaves are given in the table. These are the values with zero as the high byte (ignore the zero and use only the low byte). For this particular program, *add three* to each value listed; thus a low B (the first pitch with high byte=0) should be 250 instead of 247. Experiment with different frequency and duration values.

Now let's explore a routine that permits a full range of pitches, down to two octaves below middle C or lower, if you wish. Here we require a two-byte frequency value. The machine language is more extensive. Again begin with 10 REM, adding the following sequence of character strokes: NEXT, A, /, I, =, upper left quarter square graphic, upper left quarter square graphic, COPY, VAL, PEEK, COPY, lower left quarter square graphic, left half square graphic, C, upper right quarter square graphic, (, RETURN, £, $, C, upper half square graphic, $, /, RUN, AT, <=, RETURN, lower left quarter square graphic, left half square graphic, C, upper right quarter square graphic, (, RETURN, £, $, C, upper half square graphic, $, /, RUN, H, 4, AND, 9, 4, STR$, TAN.

Whew! Again conclude by hitting ENTER, and double or triple-check the entry. Be sure to identify the correct graphics characters, preceding and following them by pressing the GRAPHICS key to obtain or cancel the G cursor. To get the graphics character rather than a reverse field letter, you must use the shift key.

This routine may be quick-checked also by adding the statement: 30 LET A=USR 16514. This time you should hear a slightly low B *below* middle C, for about two seconds. The complete BASIC program is listed below.

```
10 REM (as above)
12 LET B=1
13 LET C=255
14 LET D=10
15 LET X=B*256+C
17 POKE 16520,B
18 POKE 16521,C
22 POKE 16518,D*1000/X
23 IF PEEK 16518=0 THEN POKE 16518,1
30 LET A=USR 16514
```

Here B and C are the two-byte frequency values (high-low order). X sums the two values for purposes of calculating the duration to be POKEd into 16518. Statement 23 may seem problematic. It is needed because very low notes (high B and C values) may cause 16518 to contain zero, which will produce a *very long* tone in combination with a small D value. Statement 23 protects against this possibility. To obtain accurate rhythms on low short tones, speed up the tempo (POKE a value less than 24 into 16516) so that you can work with larger values of D. Again, try different frequencies and durations, using the frequency table at the end of the article. B is the high byte, C, the low. Use the values straight out of the table (do *not* add 3).

Varying Tone

Next we have a modification of the above ML routine that permits variation in tone color. It is more complex and difficult to use, but permits any width of pulse (rectangular) wave by changing a single variable. While requiring considerable additional overhead, only one more variable is specified (pulse width). Frequency values are the same as in the previous program. It will be easiest to enter this routine as a modification of the previous one.

Start, as usual, with 10 REM then: NEXT, A, /, I, =, upper left quarter square graphic, upper left quarter square graphic, COPY, PEEK, COPY, lower left quarter square graphic, left half square graphic, C, upper right quarter square graphic, (, RETURN, £, $, C, upper half square graphic, $, /, RUN, upper left quarter square graphic, upper left quarter square graphic, COPY, <=, RETURN, lower left quarter square graphic, left half square graphic, C, upper right quarter square graphic, (, RETURN, £, $, C, upper half square graphic, $, /, RUN, H, 4, OR, 9, 4, USR, TAN. Conclude with ENTER. Adding the statement 30 LET A=USR 16514 should result in the same tone as previously, since a square wave (50 percent pulse width) is specified.

As the BASIC overhead for this routine is fairly extensive, it may not run on the 1K ZX-81. All other programs should.

```
11 LET P=50
12 LET B=1
13 LET C=255
14 LET D=10
15 LET X=B*256+C
16 LET Y=P*.02*X
17 POKE 16520,INT (Y/256)
18 POKE 16521,Y-PEEK 16520*256
19 LET Y=(100-P)*.02*X
20 POKE 16538, INT (Y/256)
21 POKE 16539,Y-PEEK 16538*256
22 POKE 16518,D*1000/X
23 IF PEEK 16518=0 THEN POKE 16518,1
30 LET A=USR 16514
```

P is the new variable, representing pulse width expressed as a percentage (greater than zero and less than 100). The other statements are needed to calculate, from a single two-byte frequency number, the two sets of timing loop values for the top and bottom portions of the pulse wave. Suffice it to say that the two pairs of loop values go into locations 16520-16521 and 16538-16539, and that if the two pairs are identical, you get a square wave; otherwise, a variable-width pulse results. Incidentally, these variable pulse widths may be monitored on the screen, where the thicker streaks of white represent greater positive pulse widths at the same frequency. Note, too, that spacing of the streaks is proportional to frequency.

A Short Melody

Finally, let's return to our first sound routine – the one with one-byte frequency values, square waves, and high pitches only – and attempt a short melodic phrase. To do this, we define a series of frequency and duration variables in arrays, inserting them in order during a FOR/NEXT loop that calls the notes one at a time.

Observe that this program is not designed to perform entire compositions (though, given enough memory, it could). I will review two commercial programs in an upcoming issue of

COMPUTE! which allow you to code or perform, then play back, extended melodies.

```
10 REM (as above, first sound routine)
12 DIM B(6)
13 DIM D(6)
14 LET B(1)=157
15 LET D(1)=3
16 LET B(2)=186
17 LET D(2)=1
18 LET B(3)=235
19 LET D(3)=4
20 LET B(4)=186
21 LET D(4)=4
22 LET B(5)=157
23 LET D(5)=4
24 LET B(6)=117
25 LET D(6)=8
26 FOR I=1 TO 6
27 POKE 16520,B(I)
28 POKE 16518,D(I)*1000/B(I)
35 LET A=USR 16514
40 NEXT I
```

After SAVEing the program to prevent possible catastrophe, RUN it. Do you recognize the tune? If the tempo is too slow, you can always POKE the tempo location, 16516 – insert the statement, 11 POKE 16516,15 (or POKE any other number below 24). Experiment with different speeds between 1 and 255.

You may have noticed that there is no provision for rests. Rests are a bit awkward. Perhaps you might want to work out something inside the play loop that checked for a B (frequency) array variable of zero, and converted the D (duration) array variable into an index for a "do nothing" FOR/NEXT loop. Yes, it sounds complicated. Perhaps the following suggestion is some improvement.

Add two statements to the program above: 30 IF B(I)=255 THEN POKE 16528,255, and 32 IF B(I)<>255 THEN POKE 16528,254. Now, if you code a FREQUENCY value of 255, you'll get a rest of the specified duration rather than a pitch. Make sure to reserve the "pitch" of 255 for a rest. Or you may choose, and reserve, any other value greater than zero to 255 for this purpose. Unfortunately, zero won't work. After inserting this code, try substituting 255 for one of the B array values in the melody. You should get a note hole.

Comparable additions to the BASIC code for the other sound routines are also possible. I suggest reserving the value of 255 for the *lower* frequency byte. In the routine that deals with square waves throughout the frequency range, add the statements IF C(I)=255 THEN POKE 16540,255 and IF C(I)<>255 THEN POKE 16540,254 during the play loop. Note the change in memory location. The C array represents the lower frequency byte. For the routine with variable pulse width, use the same two statements, but POKE 16541 instead of 16540.

What follows is a list of delay loop (frequency) POKE values for equally tempered pitches through five octaves around middle C. The first pitch is two octaves below middle C, the last, three octaves above middle C. Be sure to *add three* to these values when using the one-byte, simple sound routine. This routine handles only pitches with a high byte of zero, which is omitted.

Pitch Values

PITCH	HIGH BYTE	LOW BYTE
C	7	105
C#/DF	6	250
D	6	145
D#/EF	6	35
E	5	218
F	5	130
F#/GF	5	50
G	4	228
G#/AF	4	155
A	4	90
A#/BF	4	23
B	3	225
c	3	167
c#/dF	3	114
d	3	65
d#/eF	3	17
e	2	230
f	2	188
f#/gF	2	149
g	2	110
g#/aF	2	75
a	2	42
a#/bF	2	11
b	1	238
c1 (middle c)	1	210
c#1/dF1	1	184
d1	1	159
d#1/eF1	1	136
e1	1	113
f1	1	92
f#1/gF1	1	72
g1	1	53
g#1/aF1	1	36
a1	1	19
a#1/bF1	1	3
b1	0	247
c2	0	232
c#2/dF2	0	219
d2	0	207
d#2/eF2	0	195
e2	0	183
f2	0	173
f#2/gF2	0	163
g2	0	154
g#2/aF2	0	145
a2	0	136
a#2/bF2	0	128
b2	0	121
c3	0	114
c#3/dF3	0	107
d3	0	101
d#3/eF3	0	95
e3	0	89
f3	0	84
f#3/gF3	0	79
g3	0	74
g#3/aF3	0	70
a3	0	66
a#3/bF3	0	62
b3	0	58
c4	0	55

©

THUNDERBIRD

Dave Sanders, Garland, UT

For TRS-80 Color Computer, Atari and Unexpanded VIC, this game should prove a challenge for all age levels. So far, none of the players who've tried it have been able to get past the second level. But if you do, the game will keep getting harder.

"Thunderbird" will demand your undivided attention and all of the memory the unexpanded VIC-20 has to offer. The object of Thunderbird is to score as high as possible. The high score will be kept from game to game. The scoring is as follows: 200 points for taking out a tree, 50 points for taking out a saucer, 75 points for deflecting off either wing of the Thunderbird, 25 points for deflecting off the main body of the Thunderbird, and 1000 points for breaking out the bottom of the playing field. When the satellite drops into a well, 125 points are subtracted from the score.

You score these points by keeping the satellite in the playing field. The satellite can break out the top and the bottom of the screen. When it breaks out the bottom, you score 1000 points, and a new and more difficult playing field is set up for you. If the satellite breaks out the top of the field, your game is half over. You can lose only two satellites out the top. You prevent the satellite from breaking out the top by deflecting it back into the field with the Thunderbird. The Thunderbird is moved across the top of the field with the cursor control keys.

The display on the right side of the screen tells you if you are playing the first or second satellite. When the satellite drops into a well, the Thunderbird lasers down from one to three multicolored saucers to further hinder the satellite from breaking out the bottom. You will notice that the Thunderbird deflects the satellite one way off its main body and a different way off its wings. You have to keep the Thunderbird moving across the screen in conjunction with the direction the satellite is moving, or you will not play for very long.

With a little practice, the first breakout is not too hard. The second breakout will not be out of reach either, but no one in our neighborhood has broken out the third time. Just in case you are a whiz though, the game will continue to get harder.

Crunching It Into The VIC

Because of the length of the program, I had to use a technique known as "crunching." You can pack more instructions – and power – into your BASIC programs by making each program as short as possible.

Crunching programs lets you squeeze the maximum possible number of instructions into your program. It also helps you reduce the size of programs which might not otherwise run in a given size.

A list of keyword abbreviations is given in the Appendix D in the *Personal Computing Guide* that you received with your VIC-20. This is helpful when you program because you can actually crowd more information on each line by using these abbreviations. In this program it is mandatory to use this technique on many of the lines when you type them in. The most frequently used abbreviation in this program is PO (P shifted-O) which is the BASIC abbreviation for the POKE command. However, if you LIST a program that has abbreviations, the VIC-20 will automatically print out the listing with the full-length keywords.

If any program line exceeds 88 characters (four lines on the screen) with the keywords unabbreviated, and you want to change it, you will have to re-enter that line with the abbreviations before saving the program.

SAVEing a program incorporates the keywords without inflating any lines because BASIC keywords are tokenized by the VIC-20. Usually, abbreviations are added after a program is written and do not have to be LISTed any more before SAVEing.

REM statements are helpful in reminding yourself – or showing other programmers – what a particular section of a program is doing. However, when the program is completed and ready to use, you probably will not need those REM statements any more; you can save quite a bit of space by removing them. If you plan to revise or study the program structure in the future, it is a good idea to keep a copy on file with the REM statements intact.

computer mail order west

HOME COMPUTERS

ATARI

400

- 16K........ $199
- 32K (Non Atari Ram) $274
- 48K (Non Atari Ram) $359

410 Recorder	$74.00
810 Disk Drive	$429.00
822 Printer	$269.00
825 Printer	$589.00
830 Modem	$159.00
820 Printer	$259.00
850 Interface	$169.00
CX40 Joysticks (pair)	$18.00
CX853 Atari 16K Ram	$77.95

800

48K........ $499

New low price effective January 1, 1983.

Microtek 16K Ram	$ 74.95
Axlon Ramdisk (128K)	$429.95
Intec 48K Board	$159.00
Intek 32K Board	$ 74.00
One Year Extended Warranty	$ 70.00
CX481 Entertainer Package	$ 69.00
CX482 Educator Package	$130.00
CX 483 Programmer Package	$54.00
CX 484 Communicator Package	$344.00

PERCOM

DISK DRIVES FOR ATARI COMPUTERS

S1 Single Drive	$549.00
A1 Add-On Drive	$339.00
S2 Dual Drive	$879.00
Single Side Dual Head	$679.00
Dual Drive Dual Head	$1046.00

μ-SCI

MICRO-SCI DISK DRIVES FOR FRANKLIN & APPLE

A2	$299.00
A40	$369.00
A70	$499.00
C2 Controller	$79.00
C47 Controller	$89.00

SOFTWARE FOR ATARI

ATARI

Pac Man	$33	Missile Command	$29
Centipede	$33	Star Raiders	$35
Caverns of Mars	$32	Galaxian	$33
Asteroids	$29	Defender	$33

ON-LINE

Jawbreaker	$27	Mission Asteroid	$22
Softporn	$27	Mouskattack	$31
Wizard & Princess	$29	Frogger	$31
The Next Step	$34	Cross Fire (ROM)	$36

SYNAPSE

File Manager 800	$89	Shamus	$26
Chicken	$26	Protector	$26
Dodge Racer	$26	Nautilus	$26
Synassembler	$30	Slime	$26
Page 6	$19	Disk Manager	$24

DATASOFT

Pacific Highway	$25	Graphic Generator	$13
Canyon Climber	$25	Micro Painter	$25
Tumble Bugs	$25	Text Wizard	$79
Shooting Arcade	$25	Spell Wizard	$64
Clowns & Balloons	$25	Bishop's Square	$25
Graphic Master	$30	Sands of Egypt	$25

EPYX

Crush, Crumble	$24	Morloc's Tower	$16
Undead Crypt	$24	Rescue at Rigel	$24
Curse of Ra	$16	Ricochet	$16
Datestones	$16	Star Warrior	$29
Invasion Orion	$19	Temple Apshai	$29
Arthur's Heir	$24	Upper Reaches	$16

APX

Text Formatter	$18.50	Holy Grail	$24
Family Budget	$18.50	Player Piano	$18.50
Eastern Front	$24	Keyboard Piano	$18.50
Family Cash	$18.50	Number Blast	$13
Jukebox	$13	Frogmaster	$18.50
Downhill	$18.50	747 Land Simul.	$18.50
Outlaw	$18.50	Word Processor	$40

CBS

K-razy Shoot Out	$32	K-razy Antics	$32
K-razy Kritters	$32	K-star Patrol	$32

STICK STAND $6.99

VISICORP

For Apple, IBM & Franklin

Visidex	$189.00
Visifile	$189.00
Visiplot	$159.00
Visiterm	$89.00
Visitrend/Plot	$229.00
VisiSchedule	$229.00
Desktop Plan	$189.00
VISICALC for Apple II plus, Atari, CBM & IBM	179.00

CONTINENTAL

The Home Acountant(Apple/Franklin)	$59.00
The Home Accountant (IBM)	$119.00
1st Class Mail (Apple/Franklin)	$59.00

SIRIUS

Free Fall	$24	Space Eggs	$24
Beer Run	$24	Sneakers	$24
Snake Byte	$24	Bandits	$28

BRODERBUND

Apple Panic	$23	Arcade Machine	$34
David's Magic	$27	Choplifter	$27
Star Blazer	$25	Serpitine	$27

INFOCOM

Deadline	$35	Zork I	$29
Star Cross	$29	Zork II or III	$29

MPC

Bubdisk (128K Ram)	$719.00

FLOPPY DISKS

MAXELL		VERBATUM	
MD I (Box of 10)	$32	5¼" SS DD	$26
MD II (Box of 10)	$44	5¼" DS DD	$36
MFD I (8")	$40	ELEPHANT	
MFD II (8" DD)	$50	5¼" SS DD	$19.99

MONITORS

AMDEK

300G	$169.00
Color I	$339.00
Color II	$699.00
Color III	$429.00

BMC

12" Green	$85.00
13" Color 1400	$279.00
13" Color 1401 (Mid Res)	$369.00

ZENITH

ZVM121	$99.00

SHARP

Sharp 13" Color TV	$275.00

PANASONIC

TR-120 MIP (High Res. Green)	$159.00
CT-160 Dual Mode Color	$299.00

PRINTERS

Smith Corona

TP 1	$599.00

C.ITOH (TEC)

Starwriter (F10-40CPS)	$1399.00
Printmaster (F10-55CPS)	$1749.00
Prowriter 80 Col (P)	$499.00
Prowriter 80 Col (S)	$629.00
Prowriter 2 (132 Col)	$799.00

OKIDATA

82A	$429.00
83A	$659.00
84P	$1079.00
84S	$1199.00

IDS

MicroPrism	$649.00
132 (fully configured)	$1599.00
80 (fully configured)	$1399.00

Call for other configurations.

DAISYWRITER

Letter Quality	$1049.00

DIABLO

620	$1179.00
630	$1849.00

MODEMS

HAYES

Smart	$239.00
Smart 1200 (1200 Baud)	$549.00
Chronograph	$199.00
Micromodem II (with Term)	$309.00
Microdem 100	$309.00

NOVATION

Cat	$144.00
D-Cat	$159.00
Auto Cat	$219.00
212 Auto Cat	$589.00
Apple Cat II	$279.00
212 Apple Cat II	$609.00

ANCHOR

Mark I (RS-232)	$79.00
Mark II (Atari)	$79.00
Mark III (TI-99)	$109.00
Mark IV (CBM/PET)	$125.00
Mark V (OSBORNE)	$95.00
Mark VI (IBM-PC)	$179.00
Mark VII (Auto Answer Call)	$119.00
TRS-80 Color Computer	$99.00
9 Volt Power Supply	$9.00

800-648-3311

west — west

IN NV. CALL (702) 588-5654

P.O.BOX 6689 STATELINE, NV. 89449

INTERNATIONAL ORDERS: All shipments outside continental United States must be pre-paid by certified check only! Include 3% (minimun $3.00) shipping and handling. **EDUCATIONAL DISCOUNTS:** Additional discounts are available from both Computer Mail Order locations to qualified Educational Institutions.

Following is a list of REM statements that would have been in my program if there had been room.

Program Line No.	Description
4-17	Routine to set up playing field.
25-26	Subroutine for printing score.
50-54	Routine for making game more difficult.
55-59	Routine for displaying instructions and a short game.
65-80	Routine for firing laser and starting satellite back at a random location.
85	Routine for the graphics when satellite takes out saucers.
90-91	Routine for sounds and points on breakout.
95-98	Routine for sounds and colors on losing satellite out the top of the field.
100-103	Routine for moving Thunderbird across screen.
104-118	This section moves satellite and has all the PEEKs for the other routines in the program.

One of the easiest ways to reduce the size of your program is to eliminate all the spaces. Although programmers often include spaces in sample programs to provide clarity, you actually do not need any spaces in your program and will save memory if you eliminate them.

Instead of PRINTing several cursor commands to position a character on the screen, it is often more economical to use the TAB and SPC instructions to position words or characters on the screen. Well, that's enough on "crunching." You can find these and many other useful instructions in the VIC-20 *Programmers Reference Guide* VM110 published by Commodore.

On line 68 a couple of saucers are lasered down by the Thunderbird. The screen code POKEd for the saucers is the same as all the other saucers, but they certainly look different. This effect is achieved by POKEing a 9 into the color code location for these saucers. POKEing a color location with a number above eight will switch that location into multicolor mode. You can get some very interesting shapes and colors by using multicolor.

In lines four through seven, the (Q) is the ball graphic, and the (W) is the circle.

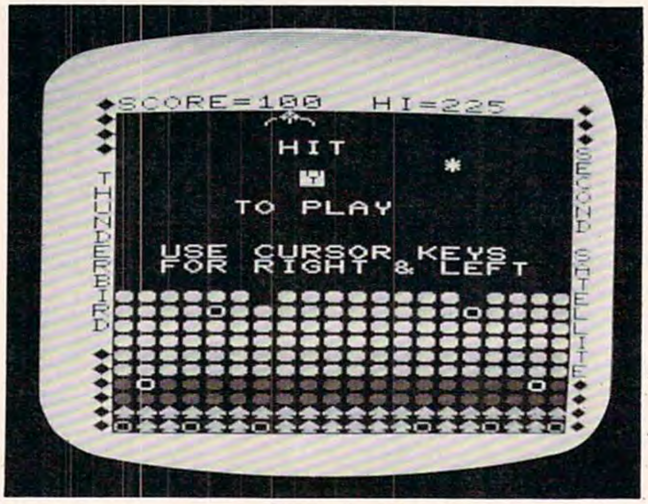
If you do not want to punch the program in, I will be happy to make a copy (VIC only) for you on tape. Send a cassette with a self-addressed, stamped (requires 40 cents postage) envelope, and a check for $3 to:

Dave Sanders
P.O. Box 533
Garland, UT 84312

Program 1: VIC-20 Version

```
2 VD=36874:F=125:OX=30742:OF=30720:P1=1:L=1:
   SC=Ø:HI=Ø:K=1:M=7703:RS=1:VA=VD+2:C=V
   A+3
4 PRINT"{CLEAR}":POKEC,1Ø5:FORR=1TO17:PRINT:
   NEXT:PRINT"{YEL}QQQQQQQQQQQQQQQQQQQQQ
   QQQQQQ{WHT}W{YEL}QQQQQQQQQQ{WHT}W{YEL
   YEL}QQQQQ"
5 PRINT"{WHT}{UP}QQQQQQQQQQQQQQQQQQQQQQQQQQQ
   QQQQQQQQQQQQQQQQQ"
6 PRINT"{CYN}{UP}QQQQQQQQQQQQQQQQQQQQQQQQQQQ
   QQQQQQQQQQQQQQQQQ"
7 PRINT"{PUR}{UP}QQ{WHT}W{PUR}QQQQQQQQQQQQQQ
   QQ{WHT}W{PUR}QQQQQQQQQQQQQQQQQQQQQQQQ
   "
8 FORR=8142TO8186:POKER,65:NEXT:J=87:G=81:FO
   RR=38423TO38442:POKER,7:NEXT:POKE3687
   8,15
9 FORR=38863TO38882:POKER,5:NEXT:FORR=38885T
   O38904:POKER,5:NEXT:POKE8165,J:POKE81
   68,J:POKE8171,J
10 POKE8178,J:POKE8181,J:POKE8184,J:FORR=7987
   TO8141STEP22:POKER+OF,1:NEXT:FORR=768
   ØTO8164STEP22
11 READA:POKER,A:NEXT:FORR=77Ø1TO8185STEP22:R
   EADA:POKER,A:NEXT:FORR=7681TO77ØØ:REA
   DA:POKER,A:NEXT
12 FORR=7966TO812ØSTEP22:POKER+OF,1:NEXT:REST
   ORE:IFP1=>2THEN5Ø
13 PRINTSPC(6)"{REV}{WHT}{22 UP}"SC:POKE7686,
   189:PRINTSPC(14)"{REV}{WHT}{UP}"HI:PO
   KE7694,189
14 X=1:Y=1:DX=1:DY=1:POKEM+1,85:POKEM+2,88:PO
   KEM+3,73:IFRS=1THENRS=RS+1:GOTO55
15 IFTT=5ØØTHENTT=1:X=12:L=1:SC=Ø:PRINT"{HOME
   HOME}{Ø7 RIGHT}{REV}      ":GOTO1Ø4
16 IFL<>2THEN1Ø4
17 POKE7767,147:POKE7789,133:POKE7811,131:POK
   E7833,143:POKE7855,142:POKE7877,132:G
   OTO1Ø4
25 PRINTSPC(6)"{UP}{REV}"SC:POKE7686,189:IFSC
   >HITHENHI=SC:PRINTSPC(14)"{REV}{UP}"H
   I:POKE7694,189
26 RETURN
5Ø FORR=7945TO7964:POKER,G:NEXT:POKE8Ø59,J:PO
   KE8Ø7Ø,J:FORR=7945TO7964:POKER+OF,7:N
   EXT
51 IFP1=>3THENPOKE8Ø12,J:POKE8Ø29,J
52 IFP1=>4THENPOKE81Ø5,J:POKE8112,J
53 IFP1=>5THENPOKE7951,J:POKE7958,J
54 GOTO13
55 POKE7754,8:POKE7755,9:POKE7756,2Ø:POKE7799
   ,153:POKE784Ø,2Ø:POKE7841,15:POKE7843
   ,16:POKE7844,12
56 POKE7845,1:POKE7846,25:POKEM+1,85:POKEM+2,
   88:POKEM+3,73:POKE7783,42:POKE79Ø3,21
   :POKE79Ø4,19:POKE79Ø5,5
57 POKE79Ø7,3:POKE79Ø8,21:POKE79Ø9,18:POKE791
   Ø,19:POKE7911,15:POKE7912,18:POKE7914
   ,11:POKE7915,5
58 POKE7916,25:POKE7917,19:POKE7925,6:POKE792
   6,15:POKE7927,18:POKE7929,18:POKE793Ø
   ,9:POKE7931,7
59 POKE7932,8:POKE7933,2Ø:POKE7935,38:POKE793
   7,12:POKE7938,5:POKE7939,6:POKE794Ø,2
   Ø
6Ø GETA$:IFA$="Y"THENSC=Ø:L=1:GOTO4
61 IFA$<>="Y"THENPOKEVA,Ø:TT=TT+1:IFTT=5ØØTHE
   N4
62 GOTO6Ø
65 SC=SC-F:G=M+2:IFHI=SC+FTHENHI=HI-F
66 POKEG+22,77:POKEC,1Ø:POKEG+OF+22,1:G=G+22
67 IFPEEK(G+22)=81ORPEEK(G+22)=65THENPOKEG,81
   :POKEG+OF,9:GOTO72
68 IFPEEK(G+22)=87THENPOKEG,81:POKEG-22,81:G=
   G-22:POKEG+OF,9:POKEG+OX,9:GOTO74
69 IFG>8185THENPOKEG,81:POKEG+OF,9:GOTO72
7Ø IFPEEK(G)=77THENPOKEG+22,78:POKEG+22+OF,1:
   G=G+22:GOTO67
71 GOTO66
72 IFPEEK(G-1)=32THENPOKEG-1,81:POKEG-1+OF,9
73 IFPEEK(G+1)=32THENPOKEG+1,81:POKEG+1+OF,9
74 POKEG-22,32:G=G-22:IFPEEK(G-22)=88THEN76
```

```
75 GOTO74
76 FORR=255TO128STEP-.9:POKEVA,R:NEXT:POKEVA,
   Ø
77 X=INT(RND(1)*18)+1:DY=1:Y=1:DX=1:IFX=>11TH
   ENDX=-DX
78 IFX=<12THENDX=+DX
79 IFDX=>5ØTHENDX=1
8Ø GOSUB25:FORR=1TO75Ø:POKEC,1Ø5:GOTO1Ø5
85 POKEBD,91:POKEBD,9Ø:DX=+DX:DY=-DY:POKEBD,9
   1:GOSUB25:POKEBD,32:GOTO1Ø5
9Ø FORR=1TO15:FORW=25ØTO24ØSTEP-1:POKEVA,W:NE
   XT:FORW=24ØTO25Ø:POKEVA,W:NEXT:POKEVA
   ,Ø:NEXT:P1=P1+1
```



Getting ready to play another game of the VIC-20 version of "Thunderbird."

```
91 FORR=1TO1ØØ:SC=SC+1Ø:POKEVA,245:GOSUB25:FO
   RW=1TO1Ø:NEXT:POKEVA,Ø:NEXT:GOTO4
95 IFL>1THENP1=1:FORR=ØTO255:POKEC,R:POKEVA,I
   NT(RND(Ø)*128+127):NEXT:POKEC,1Ø5:POK
   EVA,Ø:GOTO55
96 IFL<3THENL=L+K:POKE7767,147:POKE7789,133:P
   OKE7811,131:POKE7833,143:POKE7855,142
   :POKE7877,132:X=17
97 DX=1:Y=1:DY=1:POKEC,47:FORR=1TO28:READA:PO
   KEVA,A:POKEC,A:FORW=1TO5Ø:NEXTW,R
98 POKEVA,Ø:RESTORE:POKEC,1Ø5:DX=-DX:DY=+DY:G
   OTO1Ø4
1ØØ IFM<77Ø3THEN1Ø4
1Ø1 POKEM,85:POKEM+1,88:POKEM+2,73:POKEM+3,32:
    M=M-1:GOTO1Ø4
1Ø2 IFM>7718THEN1Ø4
1Ø3 POKEM+2,85:POKEM+3,88:POKEM+4,73:POKEM+1,3
    2:M=M+1
1Ø4 POKEBO,32:BO=77Ø3+X+22*Y:POKEBO,42
1Ø5 X=X+DX:IFX=ØORX=19THENDX=-DX:POKEVA,24Ø
1Ø6 Y=Y+DY:IFY=-1THEN95
1Ø7 IFY=22THEN9Ø
1Ø8 IFDX=ØTHENDX=1
1Ø9 POKEVA,Ø:POKEVD,Ø:BD=77Ø3+X+22*Y
11Ø IFPEEK(BD)=32THEN116
111 POKEBO,32:IFPEEK(BD)=JTHEN65
112 IFPEEK(BD)=81THENPOKEVA,238:POKEVD,238:SC=
    SC+5Ø:GOTO85
113 IFPEEK(BD)=65THENSC=SC+2ØØ:FORR=128TO255ST
    EP2:POKE36875,R:NEXT:POKE36875,Ø:GOTO
    85
114 IFPEEK(BD)=85ORPEEK(BD)=73THENPOKEVA,14Ø:S
    C=SC+75:GOSUB25:DX=+DX:DY=-DY:GOTO1Ø5

115 IFPEEK(BD)=88THENPOKEVA,212:SC=SC+25:GOSUB
    25:DX=Ø:DY=-DY:GOTO1Ø5
116 IFPEEK(197)=31THEN1ØØ
117 IFPEEK(197)=23THEN1Ø2
118 GOTO1Ø4
125 DATA 218,218,218,218,16Ø,148,136,149,142,1
    32,133,146,13Ø,137,146,132,16Ø,218,21
    8,218
126 DATA218,218,218,218,218,218,218,134,137,14
    6,147,148,16Ø,147,129,148,133,14Ø,14Ø
    ,137
127 DATA148,133,218,218,218,218,147,131,143,14
    6,133,16Ø,16Ø,16Ø,16Ø,16Ø,16Ø,136,137
    ,16Ø
128 DATA16Ø,16Ø,16Ø,16Ø,16Ø,16Ø
```

Program 2: Atari Version

```
100 REM THUNDERBIRD
110 REM Atari Version
120 GRAPHICS 0:BASE=(PEEK(106)-16)*25
    6:GOSUB 1560:REM remove old playe
    rs from screen
130 DIM A$(40),BALL$(4):POKE 82,0:BAL
    L$="*■{J}{■}":BALLS=4
140 CHSET=BASE:IF PEEK(CHSET+9)<>252
    THEN GOSUB 1200:REM If not initia
    lized
150 GRAPHICS 0:POKE 752,1:POKE 559,0:
    REM Turn off cursor, screen
160 DLIST=PEEK(560)+256*PEEK(561)+4:R
    EM location of display list
170 FOR I=3 TO 24:POKE DLIST+I,4:NEXT
     I:REM Change mode zero lines to
    IRG 4 (mulicolor character)
180 POKE DLIST-1,6+64:POKE DLIST+2,6:
    REM top two lines GRAPHICS 1
190 POKE 756,CHSET/256:REM turn on ch
    aracter set
200 SETCOLOR 0,0,12:SETCOLOR 1,3,6:RE
    M white and red
210 RESTORE 240:REM draw brick area
220 POSITION 0,0:? BALL$(1,BALLS):REM
     display # of balls (birds) left
230 REM Pattern of wall:
240 DATA 1,1,2,3,14,129,130,131
250 SCR=PEEK(88)+256*PEEK(89):REM loc
    ate screen memory
260 REM put bricks on screen
270 FOR I=SCR+520 TO SCR+800 STEP 40:
    READ A:FOR J=I TO I+39:POKE J,A:N
    EXT J:NEXT I
280 POSITION 5,0:? #6;"THUNDERBIRD"
290 P0=BASE+1024:PADR=P0+48:REM playe
    r zero.
300 POKE 704,28+176*(DIFF=1)+80*(DIFF
    =2):REM Gold, green, or violet
310 POKE 54279,BASE/256:REM single-li
    ne res.
320 POKE 53277,3:POKE 53256,3-2*(DIFF
    =1)-3*(DIFF=2):REM Start P/M DMA,
     select width according to diffic
    ulty
330 RESTORE 370
340 FOR I=0 TO 21:POKE PADR+I,0:NEXT
    I:REM clear out player
350 FOR I=0 TO 7*(3-DIFF) STEP 3-DIFF
    :READ A:FOR J=0 TO 3-DIFF:POKE PA
    DR+I+J,A:NEXT J:NEXT I
360 REM bird pattern
370 DATA 0,24,8,107,28,8,0,0
380 IF PEEK(547)<>6 THEN A=USR(1536):
    REM turn on VBLANK if necessary
390 POKE 559,62:GOSUB 750:REM turn on
     screen (single-line res. P/M), w
    ait for START
400 DY=1:DX=0.5:IF RND(1)>0.5 THEN DX
    =-0.5:REM Set up ball direction
410 BX=INT(40*RND(0)):BY=INT(7*RND(0)
    +3):REM select random starting po
```

AARDVARK

TRS-80 COLOR OSI VIC-64 VIC-20 SINCLAIR TIMEX

TUBE FRENZY
(by Dave Edson)
This is an almost indescribably fast action arcade game. It has fast action, an all new concept in play, simple rules, and 63 levels of difficulty. All machine code, requires Joysticks. Another great game by Dave Edson. TRS 80 COLOR ONLY. 16k and Joysticks required. $19.95.

QUEST – A NEW IDEA IN ADVENTURE GAMES! Different from all the others. Quest is played on a computer generated map of Alesia. Your job is to gather men and supplies by combat, bargaining, exploration of ruins and temples and outright banditry. When your force is strong enough, you attack the Citadel of Moorlock in a life or death battle to the finish. Playable in 2 to 5 hours, this one is different every time. 16k TRS-80, TRS-80 Color, and Sinclair. 13K VIC-20. $14.95 each.

CATERPILLAR
O.K., the Caterpillar does look a lot like a Centipede. We have spiders, falling fleas, monsters traipsing across the screen, poison mushrooms, and a lot of other familiar stuff. COLOR 80 requires 16k and Joysticks. This is Edson's best game to date. $19.95 for TRS 80 COLOR.

PROGRAMMERS!
SEE YOUR PROGRAM IN THIS SPACE!!
Aardvark traditionally pays the highest commissions in the industry and gives programs the widest possible coverage. Quality is the keyword. If your program is good and you want it presented by the best, send it to Aardvark.

CATCH'EM
(by Dave Edson)
One of our simplest, fastest, funnest, all machine code arcade games. Raindrops and an incredibe variety of other things come falling down on your head. Use the Joysticks to Catch'em. It's a BALL! – and a flying saucer! – and a Flying Y!– and so on. TRS 80 COLOR. $19.95.

ADVENTURES!!!
These Adventures are written in BASIC, are full featured, fast action, full plotted adventures that take 30-50 hours to play. (Adventures are interactive fantasies. It's like reading a book except that you are the main character as you give the computer commands like "Look in the Coffin" and "Light the torch.")

Adventures require 16k on TRS80, TRS80 color, and Sinclair. They require 8k on OSI and 13k on Vic-20. Derelict takes 12k on OSI. $14.95 each.

ESCAPE FROM MARS
(by Rodger Olsen)
This ADVENTURE takes place on the RED PLANET. You'll have to explore a Martian city and deal with possibly hostile aliens to survive this one. A good first adventure.

PYRAMID (by Rodger Olsen)
This is our most challenging ADVENTURE. It is a treasure hunt in a pyramid full of problems. Exciting and tough!

HAUNTED HOUSE (by Bob Anderson)
It's a real adventure—with ghosts and ghouls and goblins and treasures and problems — but it is for kids. Designed for the 8 to 12 year old population and those who haven't tried Adventure before and want to start out real easy.

DERELICT
(by Rodger Olsen & Bob Anderson)
New winner in the toughest adventure from Aardvark sweepstakes. This one takes place on an alien ship that has been deserted for a thousand years — and is still dangerous!

BASIC THAT ZOOOMMS!!
AT LAST AN AFFORDABLE COMPILER!
The compiler allows you to write your programs in easy BASIC and then automatically generates a machine code equivalent that runs 50 to 150 times faster.

It does have some limitations. It takes at least 8k of RAM to run the compiler and it does only support a subset of BASIC—about 20 commands including FOR, NEXT, END, GOSUB, GOTO, IF, THEN, RETURN, END, PRINT, STOP, USR (X), PEEK, POKE, *, /, +, -, >, <, =, VARIABLE NAMES A-Z, SUBSCRIPTED VARIABLES, and INTEGER NUMBERS FORM 0-64K.

TINY COMPILER is written in BASIC. It generates native, relocatable 6502 or 6809 code. It comes with a 20-page manual and can be modified or augmented by the user. $24.95 on tape or disk for OSI, TRS-80 Color, or VIC.

Please specify system on all orders

ALSO FROM AARDVARK – This is only a partial list of what we carry. We have a lot of other games (particularly for the TRS-80 Color and OSI), business programs, blank tapes and disks and hardware. Send $1.00 for our complete catalog.

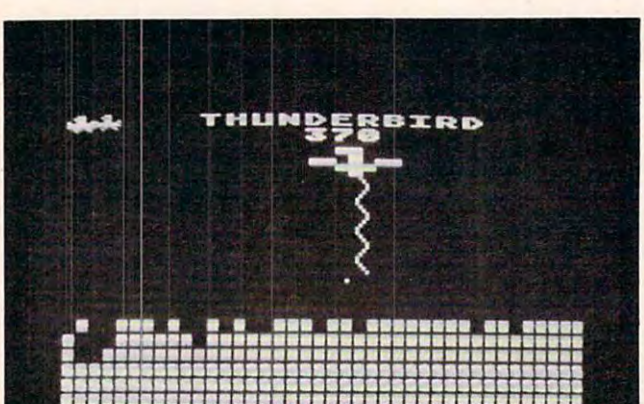
VISA

AARDVARK - 80
2352 S. Commerce, Walled Lake, MI 48088
(313) 669-3110

master charge

Phone Orders Accepted 8:00 a.m. to 4:00 p.m. EST. Mon.-Fri.

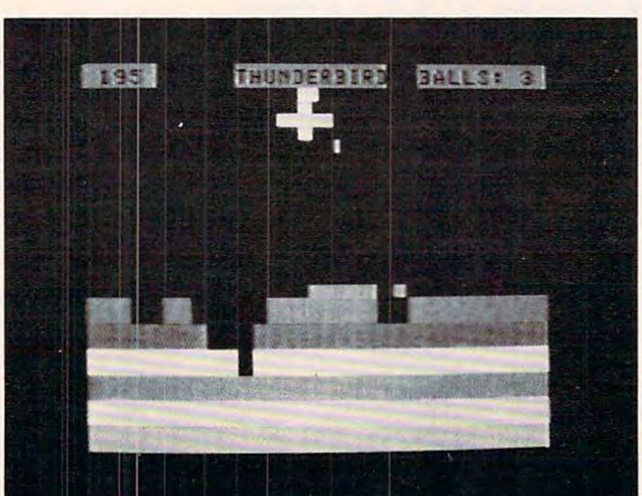
```
     sition
420 REM  Main Loop
430 IF STRIG(0)=0 THEN GOSUB 800:REM
    allow "thunder"
440 IF STICK(0)<>15 THEN POKE 77,0
450 TX=BX+DX:TY=BY+DY:REM update ball
460 IF TY<1 THEN GOSUB 600:GOTO 430:R
    EM check for miss
470 IF TY>20 THEN DY=-DY:GOSUB 920:GO
    TO 430:REM check for breakthrough
480 IF TX<0 OR TX>39 THEN DX=-DX:REM
    bounce off wall
490 TPOS=SCR+TX+40*TY:REM check for o
    bstacles
500 IF PEEK(TPOS)=0 THEN POKE TPOS,5:
    POKE SCR+BX+40*BY,0:BX=TX:BY=TY:G
    OTO 430
510 REM Rebound tiles (lasered down)
520 IF PEEK(TPOS)=4 THEN GOSUB 890:SC
    ORE=SCORE-50:DY=ABS(DY):GOTO 560
530 DY=-ABS(DY):IF RND(0)>0.5 THEN DX
    =-DX
540 FOR W=14 TO 0 STEP -2:SOUND 0,W*5
    ,10,W:NEXT W
550 SCORE=SCORE+(BY-11)*5:BLOCKS=BLOC
    KS+1:REM score according to row
560 POKE TPOS,0:POSITION 29-LEN(STR$(
    SCORE))/2,0:? " ";SCORE;" ";
570 IF BLOCKS=320 THEN 1000:REM BREAK
    -OUT!
580 IF SCORE<0 THEN 720
590 GOTO 500
600 REM Hit bird?
610 IF PEEK(53252) THEN DY=-DY:Z=1:GO
    TO 630
620 GOTO 660
630 FOR W=14 TO 0 STEP -2:SOUND 0,W+1
    0,10,W:NEXT W
640 POKE SCR+BX+40*BY,0:BX=BX+DX:BY=B
    Y+DY
650 POKE 53278,255:RETURN
660 REM Ball out of bounds (past bird
    )
670 POKE SCR+BX+40*BY,0
680 FOR W=100 TO 0 STEP -5:SOUND 0,W,
    12,8:NEXT W:FOR W=W=0 TO 100 STEP
     5:SOUND 0,W,12,8:NEXT W:SOUND 0,
    0,0,0
690 POKE 53278,255
700 BALLS=BALLS-1:POSITION BALLS,0:?
    " ";
```



Unleashing a lightning bolt in the Atari version of "Thunderbird."

```
710 IF BALLS>0 THEN 400
720 REM  GAME OVER
730 POSITION 5,0:? " GAmE OVeR "
740 GOSUB 750:RUN
750 IF PEEK(53279)=6 THEN POSITION 20
    ,0:? "{5 SPACES}":RETURN
760 IF PEEK(20)>20 THEN POSITION 20,0
    :? "PRESS"
770 IF PEEK(20)>40 THEN POSITION 20,0
    :? "start":POKE 20,0
780 GOTO 750
790 REM LASER DOWN
800 XPOS=(PEEK(1664)-48)/4+4:FLIP=0:R
    EM equate player pos. to screen p
    os.
810 FOR I=3 TO 12:WHERE=SCR+XPOS+40*I
820 P=PEEK(WHERE):POKE WHERE,6+FLIP:F
    LIP=1-FLIP:REM zig-zag line
830 SOUND 0,I*10,0,15-I:POKE 710,PEEK
    (53770):REM scintillate color
840 NEXT I
850 FOR I=3 TO 12:POKE SCR+XPOS+40*I,
    0:NEXT I:REM erase lightning
860 WHERE=SCR+12*40+XPOS:SOUND 0,0,0,
    0:POKE WHERE-1,4:POKE WHERE+1,4:P
    OKE WHERE,4:REM lay down tiles
870 SETCOLOR 2,9,4:RETURN
880 REM sound effect:
890 FOR W=0 TO 240 STEP 30:SOUND 0,W,
    12,15-W/17:SOUND 1,W+10,10,15-W/1
    7:NEXT W:SOUND 0,0,0,0:SOUND 1,0,
    0,0
900 RETURN
910 REM break-through
920 IF DONE THEN RETURN
930 FOR I=1 TO 100:POKE 53274,PEEK(53
    770):SOUND 0,I,0,15-I/10:NEXT I
940 SOUND 0,0,0,0:POSITION 4,0:? "bre
    akthrough":POSITION 22,0:? "1000
    point BONUS"
950 FOR I=1 TO 10:POSITION 22,0:? "10
    00":FOR W=1 TO 20:NEXT W:POSITION
     22,0:? "{4 SPACES}":FOR W=1 TO 2
    0:NEXT W:NEXT I
960 POSITION 4,0:? " THUNDERBIRD ":PO
    SITION 22,0:? "{17 SPACES}"
970 FOR I=1 TO 10:FOR J=0 TO 15 STEP
    5:SOUND 0,50+10-I,0,15-J:NEXT J:S
    CORE=SCORE+100
980 POSITION 29-LEN(STR$(SCORE))/2,0:
    ? " ";SCORE;" ";
990 NEXT I:DONE=1:RETURN
1000 REM  All bricks cleared
1010 GOSUB 1100:REM do "BLAST"
1020 FOR I=1 TO 50:FOR J=0 TO 3:POKE
     708+J,PEEK(53770):NEXT J:Z=Z*(Z<
     5)+1
1030 SOUND 0,I+Z,10,I/10:SOUND 1,I+Z+
     10,10,I/10:NEXT I
1040 SOUND 0,0,0,0:SOUND 1,0,0,0:GOSU
     B 1560
1050 GRAPHICS 18:POSITION 0,6:? #6;"
     {Q}{P}{L}{3 P} point bonus{R}"
1060 FOR W=1 TO 100:SOUND 0,PEEK(5377
     0),0,15-W/10:POKE 712,(3-FLIP*2)
     *16+FLIP*4+4:FLIP=1-FLIP:NEXT W
1070 SCORE=SCORE+10000:SOUND 0,0,0,0
1080 DIFF=DIFF+1:IF DIFF>2 THEN DIFF=
     2
1090 GOTO 150
1100 POKE 82,5:POSITION 5,10
1110 ? "!!!!  #{6 SPACES}■{4 SPACES}.
     ..  !!!!!"
1120 ? "!{3 SPACES}! #{5 SPACES}■ ■
     .{3 SPACES}.{3 SPACES}■"
```

```
1130 ? "!{3 SPACES}! #{4 SPACES}■
     {3 SPACES}■ .{7 SPACES}■"
1140 ? "!!!!  #{4 SPACES}■{3 SPACES}■
      ...{4 SPACES}■"
1150 ? "!{3 SPACES}! #{4 SPACES}■■■■■■■
     {5 SPACES}.{3 SPACES}■"
1160 ? "!{3 SPACES}! #{4 SPACES}■
     {3 SPACES}■ .{3 SPACES}.
     {3 SPACES}■"
1170 ? "!!!!  #### ■{3 SPACES}■  ...
     {4 SPACES}■"
1180 POKE 82,0:RETURN
1190 END
1200 REM  Initialization stuff
1210 POKE 88,0:POKE 89,BASE/256:? "
     {CLEAR}":GRAPHICS 2+16:REM CLEAR
     S OUT P/M AND CHARACTER MEMORY
1220 POSITION 5,0:? #6;"thunderbird":
     POSITION 6,4:? #6;"patience":POS
     ITION 5,8:? #6;"READING ML"
1230 RESTORE 1260
1240 FOR I=1536 TO 1611:READ A:SOUND
     0,A,10,8:POKE 712,A:POKE I,A:NEX
     T I
1250 A=USR(1536):GOTO 1400
1260 DATA 104,173,34,2,141,74
1270 DATA 6,173,35,2,141,75
1280 DATA 6,169,6,162,6,160
1290 DATA 23,32,92,228,96,24
1300 DATA 173,128,6,141,0,208
1310 DATA 173,124,2,208,6,206
1320 DATA 128,6,206,128,6,173
1330 DATA 125,2,208,6,238,128
1340 DATA 6,238,128,6,173,128
1350 DATA 6,201,1,176,5,169
1360 DATA 200,141,128,6,201,250
1370 DATA 144,5,169,32,141,128
1380 DATA 6,76,73,6
1390 REM
1400 POSITION 3,8:? #6;"LOADING  CHSE
     T"
1410 FOR I=128 TO 510:POKE CHSET+I,PE
     EK(57344+I):SOUND 0,I/2,12,8:POK
     E 712,I/2:NEXT I
1420 RESTORE 1460
1430 READ A:IF A=-1 THEN SOUND 0,0,0,
     0:SOUND 1,0,0,0:RETURN
1440 FOR J=0 TO 7:READ B:SOUND 0,B,10
     ,8:SOUND 1,B+10,10,8:POKE 712,B:
     POKE CHSET+A*8+J,B:NEXT J
1450 GOTO 1430
```



Making a "wing shot" in the TRS-80 Color Computer version of "Thunderbird."

```
1460 DATA 1,0,252,168,84,252,168,252,
     0
1470 DATA 2,0,168,168,252,252,168,168
     ,0
1480 DATA 3,0,216,120,184,228,180,212
     ,0
1490 DATA 4,0,0,0,219,150,0,0,0
1500 DATA 5,0,0,0,16,32,0,0,0
1510 DATA 6,192,192,48,48,12,12,3,3
1520 DATA 7,3,3,12,12,48,48,192,192
1530 DATA 10,24,40,24,153,126,255,20,
     34
1540 DATA 14,0,126,126,126,126,126,12
     6,0
1550 DATA -1
1560 REM KILL P/M GRAPHICS
1570 POKE 53277,0:FOR I=0 TO 3:POKE 5
     3261+I,0:NEXT I
1580 RETURN
```

Program 3: TRS-80 Color Computer Version

```
100 ' THUNDERBIRD
110 ' COLOR COMPUTER VERSION
120 CLS 0
125 BL$=CHR$(128)+CHR$(128)+CHR$(128)
    +CHR$(128)
130 PRINT @10,"THUNDERBIRD";
140 H1$=CHR$(128)+CHR$(157)+CHR$(154)
150 H2$=CHR$(128)+CHR$(149)+CHR$(158)
160 BD$=CHR$(156)+CHR$(157)+CHR$(158)
    +CHR$(156)
165 BALLS=4
170 ' DRAW WALL OF BRICKS
180 FOR I=2 TO 7
190 PRINT @(I+7)*32,"";
200 FOR J=1 TO 32:PRINTCHR$(143+I*16)
    ;:NEXT
210 NEXT
220 ' INITIALIZE OTHER VARIABLES
230 BY=RND(10)+6:BX=RND(32)-1
240 DY=1:DX=-1:IF RND(0)>.5 THEN DX=1
245 PRINT@23,"BALLS:";BALLS;
299 ' MAIN LOOP
300 X=INT(JOYSTK(0)/2)
310 IF X>28 THEN X=28
320 IF X=OLDX THEN 360
325 PRINT@OLDX+32,BL$;:PRINT@OLDX+64,
    BL$;
330 IF X<OLDX THEN PRINT@X+32,H1$; EL
    SE PRINT @X+32,H2$;
340 PRINT@X+64,BD$;
350 OLDX=X
360 IF PEEK(65280)=126 OR PEEK(65280)
    =254 THEN GOSUB 4000
499 ' MOVE BALL
500 TX=BX+DX:TY=BY+DY
515 IF TY=31 THEN DY=-DY:GOTO700
520 IF TY>5 THEN 600
530 IF TX<X*2 OR TX>X*2+LEN(BD$)*2 TH
    EN 1000
540 DY=-DY:IF RND(0)>.5 THEN DX=-DX
550 GOTO 700
600 IF TX<0 OR TX>63 THEN DX=-DX:GOTO
     700
610 P=POINT(TX,TY)
620 IF P=0 THEN RESET(BX,BY):SET(TX,T
    Y,0):BX=TX:BY=TY:GOTO300
625 IFP=1 THEN P=-5
630 SCORE=SCORE+P*5:PRINT@0,SCORE;
635 IF SCORE<0 THEN 1030
640 PRINT@INT(TX/2)+INT(TY/2)*32,CHR$
    (128);
```

```
641 IF TY=28 THEN GOSUB 2000
645 IF P=-5 THEN DY=ABS(DY):GOTO700
650 DY=-ABS(DY)
660 HIT=HIT+1:IF HIT=192 THEN 3000
700 SOUND 240,1
710 GOTO 300
1000 FORI=1TO10:SOUND1,1:SOUND255,1:N
     EXT
1010 RESET(BX,BY)
1020 BALLS=BALLS-1:IF BALLS>0 THEN 23
     0
1030 FORI=255TO1STEP-15:SOUNDI,1:NEXT
1040 CLS5
1050 PRINT@267,"GAME OVER";
1055 PRINT@0,SCORE;
1060 IF PEEK(65280)<>126 AND PEEK(652
     80)<>254 THEN 1060
1070 RUN
1999 ' BREAKTHROUGH
2000 IF FIRST=1 THEN RETURN
2010 FIRST=1
2015 PRINT@6,"1000 POINT BONUS";
2020 FOR J=1 TO 10 STEP 2
2025 T=1-T:IF T THEN PRINT@6,"1000";
     ELSE PRINT@6,BL$;
2030 FOR I=200+J TO 210+J
2040 SOUND I,1
2050 NEXT:NEXT
2060 PRINT@6,BL$;"THUNDERBIRD";BL$;
2070 SCORE=SCORE+1000
2080 PRINT@0,SCORE;:PRINT@23,"BALLS:"
     ;BALLS;
2090 RETURN
2999 ' ALL BRICKS HIT
3000 FOR I=1 TO50
3010 CLS RND(8)-1
3020 PRINT@263,"10,000 POINT BONUS!";
3030 NEXT:CLS 0
3040 SCORE=SCORE+10000
3050 DIF=DIF+1
3060 IF DIF=3 THEN DIF=2
3070 ON DIF GOTO 3100,3200
3100 H1$=""
3110 H2$=H1$
3120 BD$=CHR$(169)+CHR$(166)
3130 GOTO 180
3200 H1$="":H2$=""
3210 BD$=CHR$(243):GOTO180
4000 ' THUNDER DOWN
4010 FORI=3TO7
4020 PRINT@X+1+I*32,CHR$(233);
4050 NEXT
4060 SOUND245,10
4070 FORI=3TO7
4080 PRINT@X+1+I*32,CHR$(128);
4090 NEXT
4100 PRINT@X+I*32,CHR$(131);CHR$(131)
     ;CHR$(131);
4110 RETURN
```

Atari Notes:

Thunderbird

Charles Brannon, Editorial Assistant

Thunderbird for Atari requires a joystick and 16K of memory. You move the bird left and right with your joystick, using it to bounce a ball into a wall of bricks. As in Breakout, the object of the game is to clear out all the bricks, without letting the ball escape past you. A 1,000 point bonus is awarded when you break out the bottom of the wall (a "breakthrough"); and if you're really good, you get 10,000 points for clearing out all the bricks (no mean feat!).

Shades Of Zeus

But the Thunderbird is no mere Breakout paddle! It can unleash the most awesome power of nature – lightning – at the touch of a button (the fire button). Thunderbird will "beam down" several luminous "tiles" that serve to deflect the ball downwards when hit. You can lay down tiles like a cap over a hole the ball has created, to force it to widen the hole. Every time the ball hits a tile, it swoops downward, but 25 points are subtracted from your score. That should discourage overuse of this miraculous feature.

Vertical Blank And IRG 4

Here's a bit of information about the programming. The playing field is a mixed-mode display consisting of two rows of GRAPHICS 1 text, and 21 rows of a multicolored character mode, IRG 4. This lets us have multicolored bricks.

Player/missile graphics are used to represent the bird, which can be any of three sizes, depending on the skill level. The bird is moved left and right by a small machine language routine that is executed every 1/60 second during the TV's vertical blank (when the electron beam is traveling from the lower right-hand corner to the upper left-hand corner of the screen).

IRG mode 4, the multicolor mode, is quite interesting. A single character can be any of three colors. To design these colored characters, divide the character horizontally into four two-bit zones. Each two-bit block controls one pixel of color within the character (a multicolor character's resolution is 4x8). No color would be 00, color one is 01, two 10, and three 11 (simple two-bit binary). For example, one of the bricks consists of several colored bands:

```
1110
2220
3330
1110
2220
3330
1110
0000
```

The numbers correspond to a "COLOR" statement. One side and the bottom row are left blank, so the blocks won't touch. The pattern, when expanded into binary, would look like:

```
01010100
10101000
11111100
01010100
10101000
11111100
01010100
00000000
```

Such a "custom character" would look strange on a normal screen (although you would see some semblance of multicolors, due to artifacting). But when displayed on either an IRG 4 or IRG 5 mode screen, each character is like a tiny 4x8 block of GRAPHICS 7 pixels. Also, any character printed in inverse (with the Atari logo key) will look different. The COLOR 3 pixels in such a character will be displayed as COLOR 4 (normally available only in GRAPHICS 1 or 2).

To create an IRG 4 screen, you must replace the bytes for GRAPHICS 0 by modifying the display list. Luckily, the resolution of IRG 4 is identical to GRAPHICS 0, 40x24.

```
DL=PEEK(560)+256*PEEK(561)+4
POKE DL-1,4+64
FOR I=2 TO 24:POKE DL+I,4:NEXT I
```

See lines 160-180 of Thunderbird. You can also try out IRG 5, which displays these characters in double-height (40x12). ©

HOME ENERGY CALCULATOR

David Swaim, Atlanta

You (and your computer) could become pretty popular when word gets out that you can analyze the benefits of home improvements on fuel bills. This program is in Microsoft (Apple, PET, OSI, etc.) and Atari BASIC.

Lately there has been a great deal of interest in saving energy in the home. Nobody needs to be reminded that fuel costs are rising. We all want to reduce our energy bills. The way to do this is simple: reduce household energy consumption. There are a number of ways this can be done.

The cheapest way is to change habits. An example would be setting the thermostat back to a lower temperature and wearing heavier clothes. If you're not too keen on that, the next alternative is to improve the ability of the house to protect you from the elements. Insulation could be added to the walls, floors, attic, and heat ducts. Weather-stripping could be applied to windows and doors. Storm windows and doors could be added.

Improvements such as these reduce the amount of heat that the house will lose to the outside. But which of the above items would save us the most money? Which one would cost the least to implement? Or, better yet, which will give the greatest savings for the least amount of cost? It's this last question we really want to answer.

The best measure of the cost effectiveness of an energy saving improvement is the payback period. That is simply the amount of time (in years) it takes for the savings in energy costs to add up to the total cost of installing the improvement. Obviously, the item with the shortest payback period is the best candidate for implementation. To determine the payback period, we must know two things: how much it will cost to make the improvement, and how much it will save us on utility bills for a year (a heating season). Obtaining the improvement cost requires consulting a contractor or, if we plan to do it ourselves, a building supply store.

Predicting Effectiveness

Finding out how much the improvement will save us in heating costs over a season is not quite as easy to determine. One way would be to keep records of our heating bills for one season, make the improvement, and then keep records of our heating bills for the next heating season. There are two drawbacks to this method.

First, the severity of the weather will vary from one year to the next. If the first year is severe and the second is mild, our heating bills would be less even if we made no improvements. This problem can be corrected by adjusting the heating costs using weather data for the two years.

The second and biggest drawback to this method is that you can't find out if an improvement is cost effective until after you have installed it. If it turns out not to be cost effective, it is too late to decide not to implement it!

What we need is a way of *predicting* savings. If we know the weather and the heat loss characteristics of the house, we can estimate the heating cost. By calculating the heating costs based on heat loss characteristics of the house both before and after the improvements, we can obtain the estimated savings due to the improvements. This is what the program here does.

To gather the data needed by the program, you will need to make some measurements and observe insulation levels in your house. The first thing the program calculates is the heat loss of the house. Heat loss of a house depends on three things: the thermal resistance, known as the R-value, of the structure; the total area of the structure exposed to the elements; and the temperature difference between the inside and outside of the house. So we simply need the area, R-value, and the difference in temperature.

The only problem is that different parts of the house have different R-values. Windows will have a lower R-value than walls, for example. In general, you can divide the external area of the house into five categories: windows, doors, walls, ceiling, and floor. The program requests information on each of these five categories in turn.

For windows it requests height, width, number of windows (it calculates total window

Now the VIC 20 and 64 can communicate with PET peripherals

VIC and 64 users

Would you like to be able to access **any** of these peripherals from your computer?

- ⅓ megabyte disks (Commodore 4040 drive)
- 1 megabyte disks (Commodore 8050 drive)
- 10 megabyte disks (Commodore 9090 hard disk)
- Printers including a wide range of inexpensive IEEE and RS232 matrix and quality printers
- IEEE instruments such as volt meters, plotters etc.

Now you are no longer limited by the VIC or the 64's serial bus. Simply by attaching INTERPOD you can vastly increase the power of your VIC 20 and when used with the new 64, INTERPOD turns the computer into a really powerful system.

With INTERPOD the VIC and 64 become capable of running really professional quality software such as Word-processing, Accounting, Instrument control and many more.

INTERPOD will work with any software. No extra commands are required and INTERPOD does not affect your computer in any way.

Using INTERPOD is as easy as this:

Simply plug INTERPOD into the serial port of your computer, power-up and you are ready to communicate with any number of parallel and serial IEEE devices and any RS232 printer.

INTERPOD costs $180

INTERPOD

Oxford Computer Systems (Software) Ltd. Hensington Road, Woodstock, Oxford OX7 1JR, England Tel. (0993) 812700 Telex 83147 Ref OCSL

INTERNAL MEMO

I want a Word Processor that's simple to understand, fast, and easy to use, . .

. . it'll need up to 20,000 characters of text, a wide, clear screen, and be able to spell!

She needs 'Superscript'!

Superscript Features

- Provides full screen edit, delete, erase, insert, transfer and document merge facilities. Over 240 columns.
- Simple facilities to send mailshots to standard address lists.
- Powerful aids to produce tables with wide screen, horizontal, decimal and vertical tabs.
- Search with optional replace
- Comprehensive printer controls with control of margins, lines per page, underlining, bold print, super and sub scripts, variable line and character pitch.

Superspell Features

- Dictionary in excess of 30,000 words.
- User definable dictionary with facilities to add, delete, print or merge with standard dictionary.
- Spelling checker verifies largest *'Superscript'* document in less than two minutes.
- Displays totals of words, different words and unrecognized words.
- Editing of unrecognized words in document includes options to accept, ignore, change or add to user dictionary.

'Superscript' transforms your Commodore computer into a true Word Processor, enabling your secretary to turn out high quality letters, mailshots, quotations, etc., faster and easier than ever before.

But spell? — Adding *'Superspell'* gives you access to an extensive dictionary with automatic checking of any document entered, enabling recognition of spelling errors and mistypes.

If she wants a better machine, and you want the very best of Word Processing, then you both need a Commodore with *'Superscript'* and *'Superspell'* available from your local Commodore dealer.

Superscript* and *Superspell run on the 2001, 3016/32, 4016/32, 8032 and 8096 Commodore computers, 2040, 3040, 4040 and 8050 disk drives, all Commodore printers and a wide range of letter quality printers.

Precision Software

Superscript

A Professional's Word Processor

Precision Software Limited,
4, Park Terrace, Worcester Park,
Surrey KT4 7 JZ, England.
Telephone 01-330 7166

For further information concerning these outstanding software products contact your local SUPERSCRIPT distributor as shown.

Alabama
To Be Announced
Alaska
B. G. Systems
Anchorage
907 276 2986
Arizona
Gerald Hasty & Co.
Las Vegas
Nevada
702 737 5670
Arkansas
To Be Announced
California (North)
To Be Announced
California (South)
P. E. C.
Anaheim
714 778 3007
Colorado
To Be Announced
Connecticut
Multi Business Computer Systems
Portland
203 342 2747
Delaware
Professional Micro Services
Baltimore
Maryland
301 325 5725
Florida
To Be Announced
Georgia
M. A. G.
Athens
404 353 8090
Hawaii
To Be Announced
Idaho
To Be Announced
Illinois
Cambridge Business Systems
Chicago
312 525 3900
Indiana
Srepco
Dayton
Ohio
513 224 0871
Iowa
To Be Announced
Kansas
To Be Announced
Kentucky
Srepco
Dayton
Ohio
513 224 0871
Louisiana
To Be Announced
Maine
Best Business Equipment
Worcester
Massachusetts
617 755 1077
Maryland
Professional Micro Services
Baltimore
301 325 5725
Massachusetts
Best Business Equipment
Worcester
617 755 1077
Michigan
Newman Audio Video
Grand Rapids
616 243 3300
Minnesota
To Be Announced
Mississippi
To Be Announced
Missouri
To Be Announced
Montana
To Be Announced
Nebraska
To Be Announced
Nevada
Gerald Hasty & Co.
Las Vegas
702 737 5670
New Hampshire
Best Business Equipment
Worcester
Massachusetts
617 755 1077
New Jersey
Geneva Technology
Cranford
201 276 1144
New Mexico
To Be Announced
New York State (North & West)
Upstate Computer Shop
Whitesboro (Nr. Uttica)
315 768 8151
New York State (South)
Computer Emporium
Middletown
914 343 4880
New York State (Long Island)
Centerbrook Software
Livingston Manor
914 439 3591
New York City
Geneva Technology
Cranford
New Jersey
201 276 1144
North Carolina
To Be Announced
North Dakota
To Be Announced
Ohio
Srepco
Dayton
513 224 0871
Oklahoma
To Be Announced
Oregon
The Computer Place
Klamath Falls
503 882 9603
Pennsylvania (East)
Mainline Computer Center
Wayne
215 687 8500
Pennsylvania (West)
Srepco
Dayton
Ohio
513 224 0871
Rhode Island
Multi Business Computer Systems
Portland
Connecticut
203 342 2747
South Carolina
To Be Announced
South Dakota
To Be Announced
Tennessee
To Be Announced
Texas
To Be Announced
Utah
Gerald Hasty & Co.
Las Vegas
Nevada
702 737 5670
Vermont
Best Business Equipment
Worcester
Massachusetts
617 755 1077
Virginia
Professional Micro Services
Baltimore
Maryland
301 325 5725
Washington State
Computer Sales & Service
Moses Lake
509 765 9751
Washington D.C.
Professional Micro Services
Baltimore
Maryland
301 325.5725
West Virginia
Professional Micro Services
Baltimore
Maryland
301 325 5725
Wisconsin
Combridge Business Systems
Chicago
Illinois
312 525 3900
Wyoming
To Be Announced
Canada
Canadian Micro Distributors
Milton
Ontario
416 878 7277

If a SUPERSCRIPT distributor is not shown for your area, contact your nearest Commodore dealer for more information. If he does not stock SUPERSCRIPT or SUPERSPELL, write to Precision Software in England at the address shown opposite and we will send you the full distribution list.

Precision Software

area from these items), and type of frame and number of layers of glass. The number of types and/or sizes of windows is requested first. Most houses will have several sizes of windows, and there may be storm windows on some and not on others. The program allows for up to ten different types and/or sizes of windows. If you need more, change the dimension of S in statement 180.

Only one size and type of door is allowed. If you have sliding glass doors, you should consider them another type of window. You need to get the height, width, and number of doors. Remember: these are exterior doors only.

Information needed for the walls consists of type of construction and R-value of the insulation in the wall. If you enter a negative number for the R-value of the wall insulation, the program will give you a list of typical R-values for wall insulation. To get the area of the wall, the program asks for the ceiling height, total perimeter of the house, and the number of stories in the house. The program will calculate the gross wall area from this data and subtract the total window and door area to obtain the proper wall area.

One Hand Calculation

The only time you have to calculate area yourself is for ceiling and floor. For the ceiling, you will be asked for the number of inches of insulation in the attic and the type of insulating material. For the floor, the type of foundation is requested.

In addition to the heat losses mentioned so far, there are two others. The first of these is infiltration of outside air through cracks in windows and doors. The program asks if the windows and doors are weather-stripped. It uses this information and the total length of the cracks around windows and doors to calculate infiltration. The other heat loss is in the heat ducts from the furnace to the heat registers. The program asks if your heat ducts are insulated and where they are located. This concludes the input needed for calculating the total heat loss of the house. At this point the heat losses are displayed, and you are asked if you wish to make improvements to the house.

If the answer is "Y", you will be asked if you wish to improve each item. You can make improvements to one item or to any number of items. As you probably noticed, the first question you are asked is what the outside design temperature is. The outside design temperature for my area (Atlanta, Georgia) is 23 degrees. The outside design temperatures for other areas are tabulated in Table 1. For a more complete list, consult one of the references listed at the end of this article.

Actually, you do not need to put any specific temperature in here as long as it is less than 75 degrees, the inside design temperature used by the program. The program will still give you valid results for savings and payback. However, using the correct outside design temperature gives you the advantage of seeing what the furnace size would be for your house with and without the improvements. In fact, heating engineers use the same basic method as this program does to size furnaces for houses.

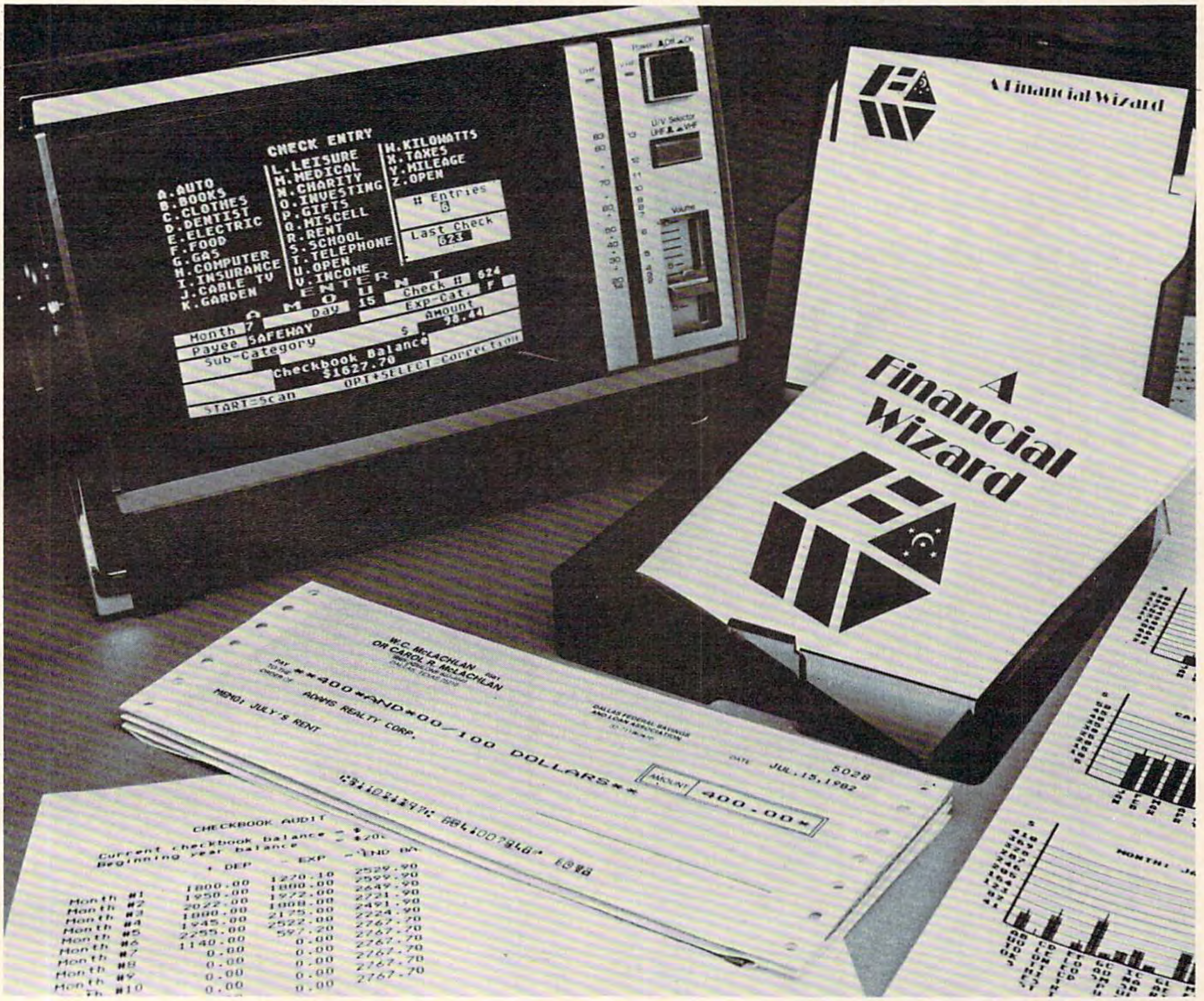
When the program finishes calculating the heat loss of the house after improvements, it is ready to do the cost analysis. First you are asked for the type of heating fuel you use: electricity,

Table 1: Winter Design Temperatures

CITY	TEMPERATURE
MONTGOMERY AL	26
JUNEAU AK	-4
PHOENIX AZ	34
LITTLE ROCK AR	23
SACRAMENTO CA	32
DENVER CO	3
HARTFORD CONN	5
DOVER DEL	15
TALLAHASSEE FL	29
ATLANTA GA	23
HONOLULU HI	62
BOISE ID	10
SPRINGFIELD IL	4
INDIANAPOLIS IN	4
DES MOINES IA	-3
TOPEKA KS	6
LEXINGTON KY	10
BATON ROUGE LA	30
AUGUSTA ME	-3
BALTIMORE MD	20
BOSTON MA	10
LANSING MI	6
ST. PAUL MN	-10
JACKSON MS	24
JEFFERSON CITY MO	6
HELENA MT	-13
LINCOLN NE	0
CARSON CITY NV	7
CONCORD NH	-7
TRENTON NJ	16
SANTA FE NM	11
ALBANY NY	5
RALEIGH NC	20
BISMARCK ND	-19
COLUMBUS OH	7
OKLAHOMA CITY OK	15
SALEM OR	25
HARRISBURG PA	13
PROVIDENCE RI	10
COLUMBIA SC	23
PIERRE SD	-9
NASHVILLE TN	16
AUSTIN TX	29
SALT LAKE CITY UT	9
BURLINGTON VT	-7
RICHMOND VA	18
OLYMPIA WA	25
CHARLESTON WV	14
MADISON WS	-5
CHEYENNE WY	-2

1983 - Take Control

A FINANCIAL WIZARD 1.5



THE HIGHLY ACCLAIMED PERSONAL FINANCE SYSTEM FOR ATARI*

- **Budget Forecast**–26 expense categories
- **Check Entry**–easy data entry–scan & modify 26 major & 36 sub-categories–information block
- **Check Search**–single or multiple parameters–(up to seven) to search entries
- **Tabulations**–detailed expense vs. budget comparisons by month, year-to-date, category
- **Bar Graphs**–screen displays in graph form expenses vs. budget–by month or category–printing with graphic capable printers
- **Check Reconciliation**–fast clearing of resident checks & deposits, complete summary report
- **Checkwriter**–prints your custom checks
- **Printouts**–most popular printers • **Audit Report**
- **Multi-Colored Graphics** • **Audio Enhancements**
- **7 Utility Programs** • **User-Friendly Operation**
- **Easy To Use Instruction Manual**
- **Handsome Tinted Plastic Storage Case**

The system is designed for Atari computers having a minimum of 24K and operating from a disk drive. The price is only $59.95 plus $3 for handling/postage.
If your dealer does not have **A Financial Wizard** . . .
Telephone orders are accepted on Mastercharge or Visa credit cards. Mail order must be accompanied by check or money-order or credit card #. Dealer Inquiries invited. (405)751-2783.

*trademark of Atari Inc.   OK residents ad 4% Tax.

A Financial Wizard Exclusively thru ON LINE COMPUTER CENTERS of OKC™ 10944 North May Oklahoma City, OK 73120 (405) 751-2781

☐ CHECK ENCLOSED ☐ VISA ☐ MASTERCHARGE

NAME (Print) ____________

Address ____________

City ____________ State ________ Zip ________

Card # ____________ Exp. ________ Sig. ________

fuel oil, or natural gas. Next you must input the cost per fuel unit of the heating fuel.

Note that this unit cost is in dollars, so if natural gas in your area is 35 cents per therm, you should input .35 dollars per therm.

Using this data and the heating degree days, the program calculates the total energy needed to heat the house for the entire heating season. The degree days and name of the city are on line 7010. You should change this line to reflect your own location. Some sample degree days for different cities are listed in Table 2, and a more complete list can be found in any of the references. The last thing you must input is the total cost of the improvements you made. From this data the program calculates the payback period in years.

I got pretty popular in my neighborhood when word got out that my home computer could calculate how cost effective it would be to add insulation. I have also learned a great deal about my own home from running this program. Much of what I concluded was what I expected, but some conclusions surprised me. The program can definitely help home owners in assessing home energy improvements; it can also enable a home owner to spot dishonest "energy-saving" schemes pretty quickly.

Table 2: Yearly Heating Degree Days

CITY	DEGREE DAYS
MONTGOMERY AL	2291
JUNEAU AK	9075
PHOENIX AZ	1765
LITTLE ROCK AR	3219
SACRAMENTO CA	2419
DENVER CO	5524
HARTFORD CONN	6235
WILMINGTON DEL	4930
TALLAHASSEE FL	1485
ATLANTA GA	2961
HONOLULU HI	0
BOISE ID	5809
SPRINGFIELD IL	5429
INDIANAPOLIS IN	5699
DES MOINES IA	6588
TOPEKA KS	5182
LEXINGTON KY	4683
BATON ROUGE LA	1560
PORTLAND ME	7511
BALTIMORE MD	4111
BOSTON MA	5634
LANSING MI	6909
MINNEAPOLIS MN	8382
JACKSON MS	2239
ST. LOUIS MO	4484
HELENA MT	8129
LINCOLN NE	5864
RENO NV	6332
CONCORD NH	7383
TRENTON NJ	4980
ALBUQUERQUE NM	4348
ALBANY NY	6201
RALEIGH NC	3393
BISMARCK ND	8851
COLUMBUS OH	5211
OKLAHOMA CITY OK	3725
SALEM OR	4754
HARRISBURG PA	5251
PROVIDENCE RI	5954
COLUMBIA SC	2484
RAPID CITY SD	7345
NASHVILLE TN	3578
AUSTIN TX	1711
SALT LAKE CITY UT	6052
BURLINGTON VT	8269
RICHMOND VA	3865
OLYMPIA WA	5236
CHARLESTON WV	4476
MADISON WS	7863
CHEYENNE WY	7381

References

1. *ASHRAE Handbook 1981 Fundamentals.* Atlanta, Georgia: American Society of Heating, Refrigerating and Air-conditioning Engineers, Incorporated, 1981.

2. *Other Homes and Garbage,* Jim Leckie, Gil Masters, Harry Whitehouse, and Lilly Young. San Francisco, California: Sierra Club Books, 1975.

3. *Refrigeration and Air-Conditioning,* Air-Conditioning and Refrigeration Institute. Englewood Cliffs, New Jersey: Prentice-Hall, 1979.

Program 1: Microsoft BASIC

```
100 PRINT"{CLEAR}{02 DOWN}  HOME EN
    ERGY PROGRAM
110 PRINT:PRINT
120 PRINT"  BY DAVID SWAIM
130 PRINT"      P. O. BOX 720126
140 PRINT"      ATLANTA, GEORGIA 303
    58
150 GOSUB 8000
160 REM COPYRIGHT 1981 DAVID C. SWA
    IM II
170 REM
180 DIM A(6),Q(6),R(6),RW(4,3),D(4)
    ,IW(2,3),S(10)
190 DIM RF(3),TC(3),N$(5),IC(5),DM(
    2,3,3),IN(2)
200 REM WINDOW R VALUES
210 DATA 1.01,2.22,1.815,3.155
220 DATA .909,1.667,1.437,2.137
230 DATA .909,2,1.724,2.564
240 REM DOOR R VALUES
250 DATA .41,.75,.95,1.1
260 REM FLOOR R VALUES AND TEMP COR
    R
270 DATA 3.2,0,3.2,30,1.23,0
280 REM CEILING INSULATION R PER IN
    CH
290 DATA 3.5,3,2.5,4.5,5.5
300 N$(1)="WINDOWS":N$(2)="DOORS":N
    $(3)="WALLS"
310 N$(4)="CEILING":N$(5)="FLOOR *"
320 REM DUCT MULTIPLIERS
330 DATA .2,.15,.1,.15,.1,.05,.1,.0
    5,.05
```

DISKEY

by Sparky Starks

Potential into practical reality. This is the core of DISKEY — a remarkable utility program that gives ATARI disk drive owners the flexibility to accomplish tasks that other utilities either ignored or only hinted at. With DISKEY, the user will be able to actually examine a disk and its directory, and repair some files that might otherwise have been lost. DISKEY also performs a multitude of other practical functions, including the following:

★ Automatically lists any unreadable or destroyed sectors
★ Sends contents of disk to printer selectively
★ Up to four separate drives may be addressed
★ Can be used to backup some of those "unbackupable" disks
★ Allows direct examination of any file
★ Over 50 separate key functions available

As an ATARI 400 or 800 owner, you have extraordinary power and versatility at your disposal. Tap into those resources effectively with DISKEY — new from Adventure International.

ATARI 400/800 32K DISK
052-0158 $49.95
(Note: Some features will require 40K)

SOME OF DISKEY KEY FUNCTIONS

A	Clear screen and filename
B	Byte compare, D1 to D2, OS to DS
C	Copy sectors, OD to DD, OS to DS
D	Toggle destination drive
E	Erase disk (format)
F	Select file sub-menu
L	Set automatic function lower limit (OS)
M	Modify Sector Map
N	New destination sector
O	Toggle originate drive
P	Print screen to printer
Q	Query (search for hex key, drive OD, sector OS to DS)
R	Read new OS, set DS to match
S	Search for ASCII key, drive OD, sector OS to DS
T	Tape to disk
U	Upper case conversion of printer lower case
V	Toggle write verify
W	Write memory buffer to sector DS, drive DD
X	Select EOR Sector Map screen print mask
Z	Zero memory buffer
+	Read upward, next sector on disk
–	Read downward
?	Directory information
!	Select directory sub-menu
cB	Byte compare, D1 to D2, whole disk
cC	Copy D1 to D2, whole disk
cD	Decimal to hex, ASCII conversion
cE	Erase disk (without new format)
cF	Modify sector forward sector chain reference
cH	Hex to decimal, ASCII conversion
cL	Locate bad sector on drive OD
cN	Modify sector file number reference
cO	Select one-drive functions sub-program
cP	Print current Disk Map
cR	RPM test drive OD
cS	Special file copy, no directory reference from source
cV	VTOC update and repair, drive OD
cY	Toggle Sure Response prompt enable
FA	File binary load address headers to printer
FD	Delete file
FF	Select filename for all file functions
FL	Lock file
FM	Show memory address load position in file
FQ	Relative Query
FR	Rename file
FS	Relative Search

BASIC ROUTINES FOR THE ATARI

by Jerry White

Finally, a self-help system that cuts through the stuffy technical jargon and allows the user to learn effective programming techniques fast! BASIC ROUTINES FOR THE ATARI has been written especially for ATARI 400 or 800 users who wish to write programs in BASIC. This package comes complete with an extensive manual and your choice of a tape or disk which contains all of the routines from the manual — which means you'll be able to actually see each of them in action on your ATARI. Some of the BASIC routines include joystick, sound, player missile strings and much more — and presented in a way that you'll quickly be able to learn and apply to your own programs.

If your programming ability lies somewhere between beginning and advanced, then look no further — BASIC ROUTINES FOR THE ATARI is the system for you.

BASIC ROUTINES book with 16K TAPE
051-0154 $24.95
BASIC ROUTINES book with 24K DISK
052-0154 $24.95

Published by ai Adventure International

© COPYRIGHT 1982

a subsidiary of Scott Adams, Inc.
BOX 3435 • LONGWOOD, FL 32750 • (305) 830-8194

To order, see your local dealer. If he does not have the program, then call 1 (800) 327-7172 (orders only please) or write for our free catalog.

```
340 DATA .2,.15,.1,.1,.1,.05,.05,.0
    5,.05
350 REM AIR CHANGES PER FOOT OF CRA
    CK
360 DATA 39,74,52,24,32,33
370 REM READ WINDOW R VALUES
380 FOR F=1 TO 3
390 FOR G=1 TO 4
400 READ RW(G,F)
410 NEXT G,F
420 REM READ DOOR R VALUES
430 FOR I=1 TO 4:READ D(I):NEXT I
440 REM READ FLOOR R VAL AND TEMP C
    ORR
450 FOR I=1 TO 3:READ RF(I),TC(I):N
    EXT I
460 REM READ INSULATION R PER INCH
470 FOR I=1 TO 5:READ IC(I):NEXT I
480 REM READ DUCT MULTIPLIERS
490 FOR KD=1 TO 2
500 FOR K=1 TO 3
510 FOR J=1 TO 3
520 READ DM(KD,J,K)
530 NEXT J,K,KD
540 REM READ AIR CHANGES FOR INFILT
    RATION
550 FOR I=1 TO 2
560 FOR J=1 TO 3
570 READ IW(I,J)
580 NEXT J,I
590 REM INSIDE DESIGN TEMPERATURE
600 IT=75:PK=1
605 GETA$:IFA$=""THEN605
610 PRINT"{CLEAR}{DOWN}WINTER OUTSI
    DE DESIGN TEMPERATURE";
620 INPUT OT
630 DT=IT-OT
640 GOSUB 1000:REM   WINDOWS
650 GOSUB 2000:REM   DOORS
660 GOSUB 3000:REM   WALLS
670 GOSUB 4000:REM   CEILING
680 GOSUB 5000:REM   FLOOR
690 GOSUB 5200:REM   DUCTS
700 GOSUB 6000:REM   REPORT RESULTS
710 Q1=TQ/DT
720 PRINT"{DOWN}DO YOU WISH TO MAKE
     IMPROVEMENTS?"
730 GET A$:IF A$="" THEN 730
740 PK=2:IF A$="N" THEN 999
750 INPUT"{CLEAR}{02 DOWN}DO YOU WI
    SH TO IMPROVE WINDOWS";A$
760 IF LEFT$(A$,1)="Y" THEN GOSUB 1
    000
770 INPUT"{CLEAR}{02 DOWN}DO YOU WI
    SH TO IMPROVE DOORS";A$
780 IF LEFT$(A$,1)="Y" THEN GOSUB 2
    000
790 INPUT"{CLEAR}{02 DOWN}DO YOU WI
    SH TO IMPROVE WALLS";A$
800 IF LEFT$(A$,1)="Y" THEN GOSUB 3
    000
810 INPUT"{CLEAR}{02 DOWN}DO YOU WI
    SH TO IMPROVE CEILING";A$
820 IF LEFT$(A$,1)="Y" THEN GOSUB 4000
830 INPUT"{CLEAR}{02 DOWN}DO YOU WI
    SH TO IMPROVE FLOOR";A$
840 IF LEFT$(A$,1)="Y" THEN GOSUB 5
    000
850 INPUT"{CLEAR}{02 DOWN}DO YOU WI
    SH TO IMPROVE DUCTS";A$
860 IF LEFT$(A$,1)="Y" THEN GOSUB 5
    200
870 GOSUB 6000:REM REPORT RESULTS
880 Q2=TQ/DT
890 PRINT:PRINT"HIT RETURN TO GET S
    AVINGS"
900 GET A$:IF A$="" THEN 900
910 GOSUB 7000:REM CALCULATE A YEAR
     OF SAVINGS
999 END
1000 REM WINDOW SUBROUTINE
1010 I=1:IF PK>1 THEN 1040
1020 PRINT"{CLEAR}{DOWN}HOW MANY DIF
     FERENT TYPES OF WINDOWS";
1030 INPUT NX
1040 IX=1:CW=0:A(I)=0:Q(I)=0
1050 PRINT"{DOWN} ARE WINDOWS WEATHE
     RSTRIPPED";
1060 INPUT WW$
1070 IF LEFT$(WW$,1)="Y" THEN IX=2
1080 FOR J=1 TO NX
1090 PRINT"SIZE";J:IF PK>1 THEN 1160
1100 PRINT"NUMBER OF WINDOWS";
1110 INPUT NW
1120 PRINT"SIZE OF WINDOWS (H,W) FT"
     ;
1130 INPUT H,W
1140 S(J)=H*W*NW
1150 CW=CW+(H+W)*NW
1160 A(I)=A(I)+S(J)
1170 PRINT"TYPE OF WINDOWS"
1180 PRINT"   1. SINGLE GLASS"
1190 PRINT"   2. SINGLE + STORM"
1200 PRINT"   3. DOUBLE PANE"
1210 PRINT"   4. TRIPLE (DOUBLE + ST
     ORM)"
1220 INPUT G
1230 PRINT"TYPE OF WINDOW FRAME"
1240 PRINT"   1. WOOD"
1250 PRINT"   2. METAL OR JALOUSE"
1260 PRINT"   3. FIXED"
1270 INPUT F
1280 RM=RW(G,F)
1290 Q(I)=Q(I)+S(J)*DT/RM
1300 R(I)=RM
1310 PRINT"{CLEAR}{DOWN}";
1320 NEXT J
1330 IN(I)=0.018*DT*IW(IX,F)*CW
1340 RETURN
2000 REM DOORS SUBROUTINE
2010 I=2:IF PK>1 THEN 2080
2020 PRINT"{CLEAR}{DOWN}NUMBER OF DO
     ORS";
2030 INPUT N
2040 PRINT"SIZE OF DOORS (H,W) FT";
2050 INPUT H,W
2060 A(I)=H*W*N
2070 CD=(H+W)*N
```

WORDCRAFT 20 ...A First! Word Processing for VIC 20®

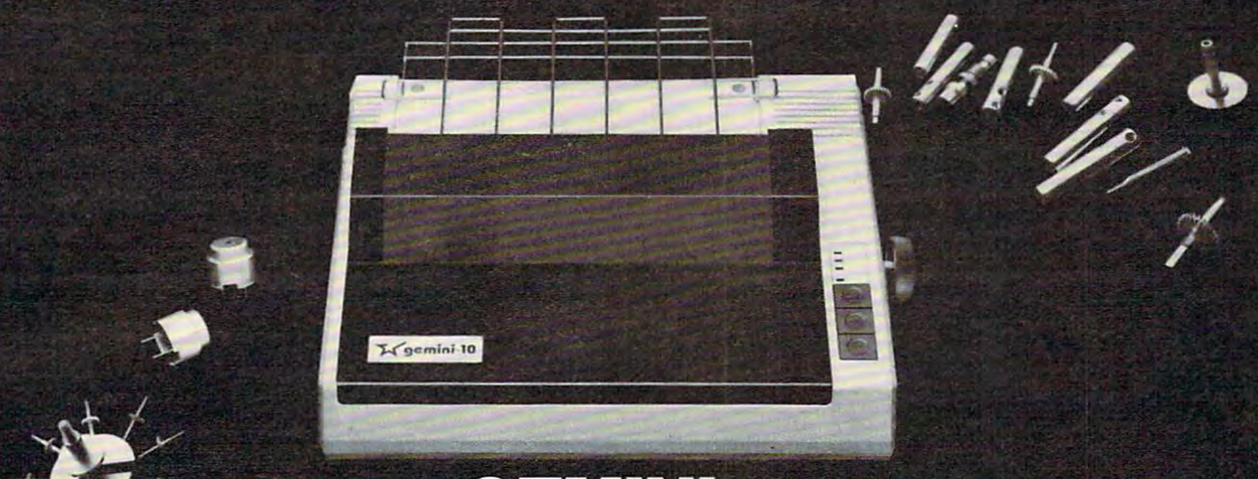
Never be at a loss for words again! Expressing yourself is easier, faster, and less expensive than ever before ... thanks to Wordcraft 20© from United Microware. Wordcraft 20© has all the features found on bigger systems costing a lot more — all the features you'll need to prepare "picture-perfect" documents, letters, personalized mailers, lists, and many other projects that have you juggling words.

Wordcraft 20© is easy to use; it won't confuse you. Your words will appear on the screen just as you've written them. This fully featured system lets you do what you need to do ... change a character, a word, or an entire block of copy . . . arrange and rearrange the text . . . set the tabs and margins . . . adjust the line lengths . . . move the copy up and down, left or right, so you see every word on the screen . . . before you print. Wordcraft 20© gives you personal word processing with professional results. Plug Wordcraft 20© into your VIC® today . . . see for yourself just what words will do. Available exclusively from your local UMI dealer.

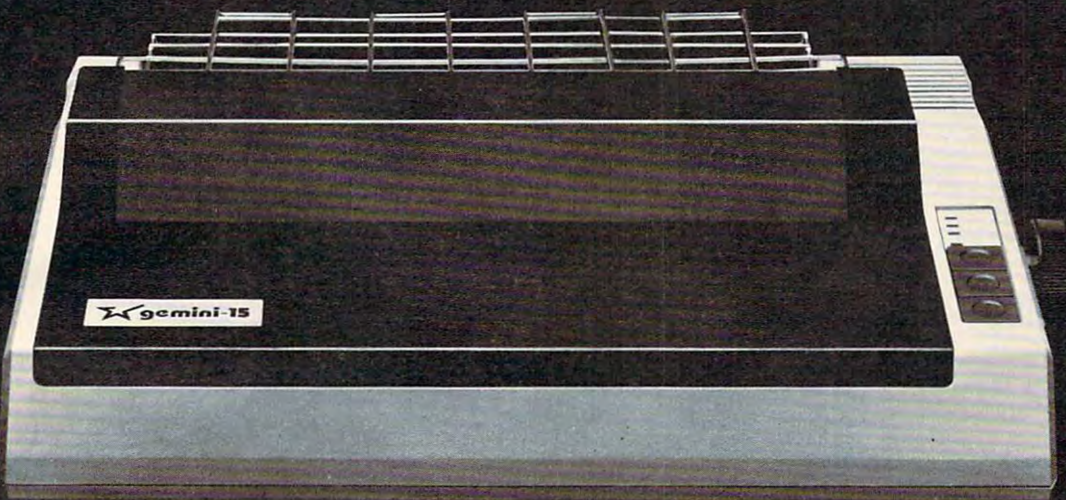
umi

United Microware Industries, Inc.
3503-C Temple Avenue
Pomona, CA 91768
(714) 594-1351

```
2Ø8Ø PRINT"{DOWN}TYPE OF DOORS"
2Ø9Ø PRINT"   1. WOOD"
21ØØ PRINT"   2. WOOD + STORM"
211Ø PRINT"   3. METAL URETHANE CORE
   "
212Ø PRINT"   4. METAL POLYSTYRENE C
   ORE"
213Ø INPUT T
214Ø R(I)=D(T)
215Ø Q(I)=A(I)*DT/R(I)
216Ø DW=138
217Ø PRINT"{DOWN}ARE DOORS WEATHERST
   RIPPED";
218Ø INPUT DW$
219Ø IF LEFT$(DW$,1)="Y" THEN DW=69
22ØØ IN(I)=Ø.Ø18*DT*DW*CD
221Ø RETURN
3ØØØ REM WALLS SUBROUTINE
3Ø1Ø I=3:HO=.17:HI=.68
3Ø2Ø PRINT"{CLEAR}{DOWN}TYPE OF WALL
    CONSTRUCTION"
3Ø3Ø PRINT"{DOWN}    1. BRICK VENEER"
3Ø4Ø PRINT "   2. STONE"
3Ø5Ø PRINT "   3. WOOD SHINGLES"
3Ø6Ø PRINT "   4. STUCCO"
3Ø7Ø PRINT "   5. MASONRY BLOCK"
3Ø8Ø PRINT "   6. LOG"
3Ø9Ø PRINT "   7. OTHER:"
31ØØ PRINT "      ENTER CALCULATED R
    VALUE DIRECTLY"
311Ø PRINT "      WHEN ASKED FOR INS
   ULATION R VALUE"
312Ø INPUT TY
313Ø ON TY GOTO 314Ø,315Ø,316Ø,317Ø,
   318Ø,319Ø,32ØØ
314Ø RM=.2*3.5:GOTO 321Ø:REM  BRICK
315Ø RM=.Ø8*5: GOTO 321Ø:REM  STONE
316Ø RM=.87:   GOTO 321Ø:REM  WOOD
317Ø RM=.2*2:  GOTO 321Ø:REM  STUCCO
318Ø RM=2:     GOTO 321Ø:REM  MASONR
   Y
319Ø RM=1.25*8:GOTO 321Ø:REM  LOG
32ØØ RM=Ø:REM                 OTHER
321Ø PRINT"   FOR LIST OF R VALUES F
   OR INSULATION"
322Ø PRINT"   ENTER -1 FOR INSULATIO
   N R VALUE"
323Ø PRINT"INSULATION R VALUE";
324Ø INPUT RI
325Ø IF RI<Ø THEN GOSUB 35ØØ:GOTO 32
   3Ø
326Ø R(I)=HO+RM+RI+HI:IF PK>1 THEN 3
   34Ø
327Ø PRINT"HOW MANY STORIES IN HOUSE
   ";
328Ø INPUT NT
329Ø PRINT"WHAT IS THE CEILING HEIGH
   T (FT)";
33ØØ INPUT CH
331Ø PRINT"WHAT IS TOTAL PERIMETER (
   FT)";
332Ø INPUT P
333Ø A(I)=NT*CH*P-A(1)-A(2)
334Ø Q(I)=A(I)*DT/R(I)
335Ø RETURN
35ØØ REM LIST OF INSULATION R VALUES
351Ø PRINT"{CLEAR}{DOWN}LIST OF INSU
   LATION R VALUES, WALLS"
352Ø PRINT"{DOWN}          NO INSULATI
   ON (AIR) = .94"
353Ø PRINT"     BATT INSULATION IN WA
   LL = 11"
354Ø PRINT"     HALF INCH ASPHALT BOA
   RD = 2.4"
355Ø PRINT"    1/2 IN GYPSUM OR PLAST
   ER = 1.39"
356Ø PRINT"     1/4 IN WOOD FIBER BOA
   RD = 1.12"
357Ø PRINT"       FIR OR PINE SHEATHI
   NG = 1.92"
358Ø PRINT"       3/4 IN PLYWOOD PANE
   LS = 1.88"
359Ø PRINT"              1/2 IN PLYWO
   OD = 1.57"
36ØØ PRINT:PRINT
361Ø RETURN
4ØØØ REM CEILING ROUTINE
4Ø1Ø I=4
4Ø2Ø HI=.61:HO=.61:IF PK>1 THEN 4Ø6Ø
4Ø3Ø PRINT"{CLEAR}{DOWN}WHAT IS TOTA
   L CEILING AREA"
4Ø4Ø PRINT"OF THE HOUSE";
4Ø5Ø INPUT A(I)
4Ø6Ø PRINT"HOW MANY INCHES OF INSULA
   TION IN CEILING";
4Ø7Ø INPUT CI
4Ø8Ø PRINT"TYPE OF INSULATING MATERI
   AL"
4Ø9Ø PRINT"{DOWN}  1. FIBERGLASS"
41ØØ PRINT "   2. MINERAL WOOL"
411Ø PRINT "   3. VERMICULITE OR PERL
   ITE"
412Ø PRINT "   4. CELLULOSE FIBER"
413Ø PRINT "   5. U-F FOAM{DOWN}"
414Ø INPUT T
415Ø RM=CI*IC(T)
416Ø R(I)=HO+RM+HI
417Ø Q(I)=A(I)*DT/R(I)
418Ø RETURN
5ØØØ REM FLOOR ROUTINE
5Ø1Ø I=5:IF PK>1 THEN 5Ø4Ø
5Ø2Ø PRINT"{CLEAR}{DOWN}WHAT IS TOTA
   L FLOOR AREA";
5Ø3Ø INPUT A(I)
5Ø4Ø PRINT"HOW MANY INS OF INSULATIO
   N IN FLOOR";
5Ø5Ø INPUT FI:IF PK>1 THEN 511Ø
5Ø6Ø PRINT"TYPE OF FOUNDATION"
5Ø7Ø PRINT"   1. OPEN CRAWLSPACE"
5Ø8Ø PRINT"   2. ENCLOSED CRAWLSPACE
     OR BASEMENT"
5Ø9Ø PRINT"   3. CONCRETE SLAB"
51ØØ INPUT TF
511Ø R(I)=HO+FI*3.1+RF(TF)+HI
512Ø Q(I)=A(I)*(DT-TC(TF))/R(I)
513Ø RETURN
52ØØ REM DUCTS
521Ø DI=.1
```



GEMINI–
FOR PRINTER VALUE THAT'S OUT OF THIS WORLD



Over thirty years of down-to-earth experience as a precision parts manufacturer has enabled Star to produce the Gemini series of dot matrix printers—a stellar combination of printer quality, flexibility, and reliability. And for a list price of nearly 25% less than the best selling competitor.

The Gemini 10 has a 10″ carriage and the Gemini 15 a 15½″ carriage. Plus, the Gemini 15 has the added capability of a bottom paper feed. In both models, Gemini quality means a print speed of 100 cps, high-resolution bit image and block graphics, and extra fast forms feed.

Gemini's flexibility is embodied in its diverse specialized printing capabilities such as super/sub script, underlining, backspacing, double strike mode and emphasized print mode. Another extraordinary standard feature is a 2.3K buffer. An additional 4K is optional. That's twice the memory of leading, comparable printers. And Gemini is compatible with most software packages that support the leading printers.

Gemini reliability is more than just a promise. It's as concrete as a 180 day warranty (90 days for ribbon and print head), a mean time between failure rate of 5 million lines, a print head life of over 100 million characters, and a 100% duty cycle that allows the Gemini to print continuously. Plus, prompt, nationwide service is readily available.

So if you're looking for an incredibly high-quality, low-cost printer that's out of this world, look to the manufacturer with its feet on the ground—Star and the Gemini 10, Gemini 15 dot matrix printers.

star® MICRONICS • INC

MAKING A NAME FOR OURSELVES

1120 Empire Central Place, Suite 216, Dallas, TX 75247
For more information, please call Bob Hazzard, Vice President, at (214) 631-8560.

```
522Ø IF TF=3 THEN KD=3:RETURN
523Ø PRINT"{DOWN}IS YOUR DUCTWORK IN
   SULATED";
524Ø INPUT D$:IF PK>1 THEN 531Ø
525Ø PRINT"{DOWN}LOCATION OF HEAT DU
   CTS:"
526Ø PRINT"    1. ATTIC OR CRAWLSPAC
   E"
527Ø PRINT"    2. UNCONDITIONED BASE
   MENT"
528Ø PRINT"    3. IN SLAB FLOOR"
529Ø PRINT"    4. INSIDE CONDITIONED
     SPACE"
530Ø INPUT KD
531Ø RETURN
6ØØØ REM WRITE A REPORT
6Ø1Ø PRINT"{CLEAR}","HEAT LOSS EVALU
   ATION"
6Ø2Ø PRINT:PRINT:TQ=Ø
6Ø3Ø PRINT" ITEM"," AREA"," R-VALUE"
   ,"HEAT LOSS"
6Ø4Ø PRINT ," SQ.FT.",," BTU/HR":PRI
   NT
6Ø5Ø FOR I=1 TO 5
6Ø6Ø A(I)=INT(A(I)*1ØØ+.5)/1ØØ
6Ø7Ø R(I)=INT(R(I)*1ØØ+.5)/1ØØ
6Ø8Ø Q(I)=INT(Q(I)+.5)
6Ø9Ø PRINT N$(I),A(I),R(I),Q(I)
61ØØ TA=TA+A(I):TQ=TQ+Q(I)
611Ø NEXT I
612Ø REM PRINT INFILTRATION LOSS
613Ø PRINT"INFILTRATION",,INT((IN(1)
   +IN(2))/2+.5)
614Ø TQ=TQ+(IN(1)+IN(2))/2
615Ø REM CALCULATE DUCT LOSS
616Ø X=TQ/(A(5)*CH*NT):J=3:K=3
617Ø IF X<45 THEN K=2
618Ø IF X<35 THEN K=1
619Ø DI=.15+.Ø5*(3-K)
62ØØ IF LEFT$(D$,1)="N" AND KD<2 THE
   N 624Ø
62Ø5 IF KD>2 THEN DI=Ø:GOTO 624Ø
621Ø IF OT<15 THEN J=2
622Ø IF OT<Ø THEN J=1
623Ø DI=DM(KD,J,K)
624Ø PRINT"DUCT LOSS",,,INT(DI*TQ+.5)
625Ø TQ=TQ+TQ*DI
626Ø PRINT ,"@@@@@@@@",,"@@@@@@@@"
627Ø PRINT"  TOTAL",INT(TA),,INT(TQ)
628Ø PRINT:PRINT
629Ø PRINT"DESIGN CONDITIONS:"
63ØØ PRINT"   OUTSIDE DESIGN TEMP";O
   T
631Ø PRINT"    INSIDE DESIGN TEMP";I
   T
632Ø PRINT"TEMPERATURE DIFFERENCE";D
   T
633Ø RETURN
7ØØØ REM FIND SAVINGS USING DEGREE-D
   AYS
7Ø1Ø DD=2961:DD$="ATLANTA GA"
7Ø12 E1=INT(Q1*DD*24)
7Ø14 E2=INT(Q2*DD*24)
7Ø3Ø PRINT"{CLEAR}TYPE OF HEATING FU
   EL USED"
7Ø4Ø PRINT"  1. ELECTRICITY"
7Ø5Ø PRINT"  2. NATURAL GAS"
7Ø6Ø PRINT"  3. FUEL OIL"
7Ø7Ø INPUT FT:PC=.55
7Ø8Ø ON FT GOTO 71ØØ,72ØØ,73ØØ
7Ø9Ø GOTO 7Ø3Ø
71ØØ REM ELECTRICITY
711Ø PRINT"IS HEATING UNIT A HEAT PU
   MP";
712Ø INPUT HP$:ER=3413
713Ø IF LEFT$(HP$,1)<>"Y" THEN 715Ø
714Ø INPUT"ENTER EER OF HEAT PUMP";E
   R:ER=ER*1ØØØ
715Ø INPUT"AVERAGE $ COST PER KWH";C
   O:FU$="KWH"
716Ø E1=INT(E1/ER+.5)
7165 M1=E1*CO
717Ø E2=INT(E2/ER+.5)
7175 M2=E2*CO
718Ø MS=M1-M2
719Ø GOTO 74ØØ
72ØØ REM NATURAL GAS
721Ø INPUT"AVERAGE $ COST PER THERM
   OF NATURAL GAS";CO
722Ø E1=INT(E1/(1Ø3ØØØ*PC)+.5)
7225 M1=E1*CO
723Ø E2=INT(E2/(1Ø3ØØØ*PC)+.5)
7235 M2=E2*CO
724Ø MS=M1-M2
725Ø FU$="THERMS":GOTO 74ØØ
73ØØ REM FUEL OIL
731Ø INPUT"AVERAGE $ COST PER GALLON
    OF FUEL OIL";CO
732Ø E1=INT(E1/(138ØØØ*PC)+.5)
7325 M1=E1*CO
733Ø E2=INT(E2/(138ØØØ*PC)+.5)
7335 M2=E2*CO
734Ø MS=M1-M2:FU$="GALLONS"
74ØØ REM GIVE RESULTS
741Ø M1=INT(M1*1ØØ)/1ØØ
742Ø M2=INT(M2*1ØØ)/1ØØ
743Ø MS=INT(MS*1ØØ)/1ØØ
744Ø INPUT"{DOWN}TOTAL $ COST OF YOU
   R IMPROVEMENTS";CI
745Ø PB=INT(CI/MS*1ØØØ)/1ØØØ
746Ø REM REPORT SAVINGS AND PAYBACK
747Ø PRINT"{CLEAR}","ANALYSIS OF IMP
   ROVEMENTS"
748Ø PRINT:PRINT
749Ø PRINT,,"ENERGY NEEDED"
75ØØ PRINT"ORIGINAL HOUSE",E1;FU$
751Ø PRINT"IMPROVED HOUSE",E2;FU$
752Ø PRINT,,"@@@@@@@@@"
753Ø PRINT,"SAVINGS",E1-E2;FU$
754Ø PRINT
755Ø PRINT,,"OPER. COSTS"
756Ø PRINT"ORIGINAL HOUSE","$";M1
757Ø PRINT"IMPROVED HOUSE","$";M2
758Ø PRINT,,"@@@@@@@@@"
759Ø PRINT,"SAVINGS","$";MS
76ØØ PRINT:PRINT,"PAYBACK",PB;"YEARS
   "
761Ø PRINT:PRINT
```

```
7620 PRINT"ABOVE IS BASED ON ONE YEA
    R OF OPERATION"
7630 PRINT"IN ";DD$
7640 RETURN
8000 REM DRAW HOUSE
8010 PRINTCHR$(142):PRINT:PRINT:PRIN
    T
8020 PRINTSPC(8);"      ,;"
8030 PRINTSPC(8);"{REV})          _{
    OFF}"
8040 PRINTSPC(8);"T,;,;,;,;,;Y
8050 PRINTSPC(8);"T<><><><><>*{REV}
        _{OFF}"
8060 PRINTSPC(8);"T,;,;,;,;,;Y,""";Y
    "
8070 PRINTSPC(8);"T<><>{REV}!{OFF}!<
    ><>Y{REV}!   {OFF}!Y"
8080 PRINTSPC(7);"##################
    ###"
8090 RETURN
```

Program 2:

Make these changes in Program 1 for the Apple II.

```
100  HOME : VTAB 2: PRINT "  HOME
      ENERGY PROGRAM"
3510  HOME : PRINT "LIST OF INSUL
     ATION R VALUES, WALLS"
3520  PRINT : PRINT "        NO INS
     ULATION
4030  HOME : PRINT "WHAT IS TOTAL
      CEILING AREA
4090  PRINT : PRINT "  1. FIBERGL
     ASS
4130  PRINT : PRINT "  5. U-F FOA
     M": PRINT
5020  HOME : PRINT "WHAT IS TOTAL
      FLOOR AREA?";
5230  PRINT : PRINT "IS YOUR DUCT
     WORK INSULATED?";
5250  PRINT "LOCATION OF HEAT DUC
     TS:
6010  HOME : PRINT "HEAT LOSS EVA
     LUATION"
7440  PRINT : INPUT "TOTAL $ COST
      OF YOUR IMPROVEMENTS";CI
7470  HOME : PRINT "ANALYSIS OF I
     MPROVEMENTS"
7520  PRINT ,,"---------"
7580  PRINT ,,"---------"
8000  RETURN
8010  - 8090"DELETE"
```

Program 3: Atari Version

```
100 POKE 82,0:PRINT "{CLEAR}{2 DOWN}
     HOME ENERGY PROGRAM"
110 PRINT :PRINT
150 GOSUB 8000
170 OPEN #1,4,0,"K:"
180 DIM A(6),Q(6),R(6),RW(4,3),D(4),I
    W(2,3),S(10)
190 DIM RF(3),TC(3),N$(5*10),IC(5),DM
    (2,15),IN(2),A$(1),NL(5)
191 DIM WW$(1),DW$(1),D$(1),DD$(20),H
    P$(1),FU$(10)
200 REM WINDOW R VALUES
210 DATA 1.01,2.22,1.815,3.155
220 DATA .909,1.667,1.437,2.137
230 DATA .909,2,1.724,2.564
240 REM DOOR R VALUES
250 DATA .41,.75,.95,1.1
260 REM FLOOR R VALUES AND TEMP CORR
270 DATA 3.2,0,3.2,30,1.23,0
280 REM CEILING INSULATION R PER INCH
290 DATA 3.5,3,2.5,4.5,5.5
300 N$(1)="WINDOWS":N$(11)="DOORS":N$
    (21)="WALLS"
310 N$(31)="CEILING":N$(41)="FLOOR *"
315 NL(1)=7:NL(2)=5:NL(3)=5:NL(4)=6:N
    L(5)=7
320 REM DUCT MULTIPLIERS
330 DATA .2,.15,.1,.15,.1,.05,.1,.05,
    .05
340 DATA .2,.15,.1,.1,.1,.05,.05,.05,
    .05
350 REM AIR CHANGES PER FOOT OF CRACK
360 DATA 39,74,52,24,32,33
370 REM READ WINDOW R VALUES
380 FOR F=1 TO 3
390 FOR G=1 TO 4
400 READ TEMP:RW(G,F)=TEMP
410 NEXT G:NEXT F
420 REM READ DOOR R VALUES
430 FOR I=1 TO 4:READ TEMP:D(I)=TEMP:
    NEXT I
440 REM READ FLOOR R VAL AND TEMP COR
    R
450 FOR I=1 TO 3:READ TEMP:RF(I)=TEMP
    :READ TEMP:TC(I)=TEMP:NEXT I
460 REM READ INSULATION R PER INCH
470 FOR I=1 TO 5:READ TEMP:IC(I)=TEMP
    :NEXT I
480 REM READ DUCT MULTIPLIERS
490 FOR KD=1 TO 2
500 FOR K=1 TO 3
510 FOR J=1 TO 3
520 READ TEMP:DM(KD,J+K*4)=TEMP
530 NEXT J:NEXT K:NEXT KD
540 REM READ AIR CHANGES FOR INFILTRA
    TION
550 FOR I=1 TO 2
560 FOR J=1 TO 3
570 READ TEMP:IW(I,J)=TEMP
580 NEXT J:NEXT I
590 REM INSIDE DESIGN TEMPERATURE
600 IT=75:PK=1
601 ? :? :? "Press RETURN to begin:";
605 GET #1,A
610 PRINT "{CLEAR}{DOWN}WINTER OUTSID
    E DESIGN TEMPERATURE";
620 INPUT OT
630 DT=IT-OT
640 GOSUB 1000:REM WINDOWS
650 GOSUB 2000:REM DOORS
660 GOSUB 3000:REM WALLS
670 GOSUB 4000:REM CEILING
680 GOSUB 5000:REM FLOOR
690 GOSUB 5200:REM DUCTS
700 GOSUB 6000:REM REPORT RESULTS
710 Q1=TQ/DT
720 PRINT "DO YOU WISH TO MAKE IMPROV
    EMENTS?:";
730 GET #1,A:A$=CHR$(A)
740 K=2:IF A$="N" THEN 999
750 PRINT "{CLEAR}{2 DOWN}DO YOU WISH
     TO IMPROVE WINDOWS";:INPUT A$
760 IF A$="Y" THEN GOSUB 1000
770 PRINT "{CLEAR}{2 DOWN}DO YOU WISH
     TO IMPROVE DOORS";:INPUT A$
780 IF A$="Y" THEN GOSUB 2000
790 PRINT "{CLEAR}{2 DOWN}DO YOU WISH
```

```
     TO IMPROVE WALLS";:INPUT A$
800 IF A$="Y" THEN GOSUB 3000
810 PRINT "{CLEAR}{2 DOWN}DO YOU WISH
     TO IMPROVE CEILING";:INPUT A$
820 IF A$="Y" THEN GOSUB 4000
830 PRINT "{CLEAR}{2 DOWN}DO YOU WISH
     TO IMPROVE FLOOR";:INPUT A$
840 IF A$="Y" THEN GOSUB 5000
850 PRINT "{CLEAR}{2 DOWN}DO YOU WISH
     TO IMPROVE DUCTS";:INPUT A$
860 IF A$="Y" THEN GOSUB 5200
870 GOSUB 6000:REM REPORT RESULTS
880 Q2=TQ/DT
890 PRINT :PRINT "HIT RETURN TO GET S
    AVINGS"
900 GET #1,A
910 GOSUB 7000:REM CALCULATE A YEAR O
    F SAVINGS
999 END
1000 REM WINDOW SUBROUTINE
1010 I=1:IF PK>1 THEN 1040
1020 PRINT "{CLEAR}{DOWN}HOW MANY DIF
     FERENT TYPES OF WINDOWS";
1030 INPUT NX
1040 IX=1:CW=0:A(I)=0:Q(I)=0
1050 PRINT "{DOWN} ARE WINDOWS WEATHE
     RSTRIPPED";
1060 INPUT WW$
1070 IF WW$="Y" THEN IX=2
1080 FOR J=1 TO NX
1090 PRINT "SIZE ";J:IF PK>1 THEN 116
     0
1100 PRINT "NUMBER OF WINDOWS";
1110 INPUT NW
1120 PRINT "SIZE OF WINDOWS (H,W) FT"
     ;
1130 INPUT H,W
1140 S(J)=H*W*NW
1150 CW=CW+(H+W)*NW
1160 A(I)=A(I)+S(J)
1170 PRINT "TYPE OF WINDOWS"
1180 PRINT "{3 SPACES}1. SINGLE GLASS
     "
1190 PRINT "{3 SPACES}2. SINGLE + STO
     RM"
1200 PRINT "{3 SPACES}3. DOUBLE PANE"
1210 PRINT "{3 SPACES}4. TRIPLE (DOUB
     LE + STORM)"
1220 INPUT G
1230 PRINT "TYPE OF WINDOW FRAME"
1240 PRINT "{3 SPACES}1. WOOD"
1250 PRINT "{3 SPACES}2. METAL OR JAL
     OUSE"
1260 PRINT "{3 SPACES}3. FIXED"
1270 INPUT F
1280 RM=RW(G,F)
1290 Q(I)=Q(I)+S(J)*DT/RM
1300 R(I)=RM
1310 PRINT "{CLEAR}{DOWN}";
1320 NEXT J
1330 IN(I)=0.018*DT*IW(IX,F)*CW
1340 RETURN
2000 REM DOORS SUBROUTINE
2010 I=2:IF PK>1 THEN 2080
2020 PRINT "{CLEAR}{DOWN}NUMBER OF DO
     ORS";
2030 INPUT N
2040 PRINT "SIZE OF DOORS (H,W) FT";
2050 INPUT H,W
2060 A(I)=H*W*N
2070 CD=(H+W)*N
2080 PRINT "{DOWN}TYPE OF DOORS"
2090 PRINT "{3 SPACES}1. WOOD"
2100 PRINT "{3 SPACES}2. WOOD + STORM
     "
2110 PRINT "{3 SPACES}3. METAL URETHA
     NE CORE"
2120 PRINT "{3 SPACES}4. METAL POLYST
     YRENE CORE"
2130 INPUT T
2140 R(I)=D(T)
2150 Q(I)=A(I)*DT/R(I)
2160 DW=138
2170 PRINT "{DOWN}ARE DOORS WEATHERST
     RIPPED";
2180 INPUT DW$
2190 IF DW$="Y" THEN DW=69
2200 IN(I)=0.018*DT*DW*CD
2210 RETURN
3000 REM WALLS SUBROUTINE
3010 I=3:HO=0.17:HI=0.68
3020 PRINT "{CLEAR}{DOWN}TYPE OF WALL
      CONSTRUCTION"
3030 PRINT "{DOWN}{3 SPACES}1. BRICK
     VENEER"
3040 PRINT "{3 SPACES}2. STONE"
3050 PRINT "{3 SPACES}3. WOOD SHINGLE
     S"
3060 PRINT "{3 SPACES}4. STUCCO"
3070 PRINT "{3 SPACES}5. MASONRY BLOC
     K"
3080 PRINT "{3 SPACES}6. LOG"
3090 PRINT "{3 SPACES}7. OTHER:"
3100 PRINT "{6 SPACES}ENTER CALCULATE
     D R VALUE DIRECTLY"
3110 PRINT "{6 SPACES}WHEN ASKED FOR
     INSULATION R VALUE"
3120 INPUT TY
3130 ON TY GOTO 3140,3150,3160,3170,3
     180,3190,3200
3140 RM=0.2*3.5:GOTO 3210:REM BRICK
3150 RM=0.08*5:GOTO 3210:REM STONE
3160 RM=0.87:GOTO 3210:REM WOOD
3170 RM=0.2*2:GOTO 3210:REM STUCCO
3180 RM=2:GOTO 3210:REM MASONRY
3190 RM=1.25*8:GOTO 3210:REM LOG
3200 RM=0:REM OTHER
3210 PRINT "{3 SPACES}FOR LIST OF R V
     ALUES FOR INSULATION"
3220 PRINT "{3 SPACES}ENTER -1 FOR IN
     SULATION R VALUE"
3230 PRINT "INSULATION R VALUE";
3240 INPUT RI
3250 IF RI<0 THEN GOSUB 3500:GOTO 323
     0
3260 R(I)=HO+RM+RI+HI:IF PK>1 THEN 33
     40
3270 PRINT "HOW MANY STORIES IN HOUSE
     ";
3280 INPUT NT
3290 PRINT "WHAT IS THE CEILING HEIGH
     T (FT)";
3300 INPUT CH
3310 PRINT "WHAT IS TOTAL PERIMETER (
     FT)";
3320 INPUT P
3330 A(I)=NT*CH*P-A(1)-A(2)
3340 Q(I)=A(I)*DT/R(I)
3350 RETURN
3500 REM LIST OF INSULATION R VALUES
3510 PRINT "{CLEAR}{DOWN}LIST OF INSU
     LATION R VALUES, WALLS"
3520 PRINT "{DOWN}{8 SPACES}NO INSULA
     TION (AIR) = .94"
3530 PRINT "{4 SPACES}BATT INSULATION
      IN WALL = 11"
```