A Tape "EXEC" For Applesoft

The 6502 Resource Magazine
PET • Apple • Atari • OSI • KIM • SYM • AIM

Using The Monitor On The PET/CBM

COMPUTE!

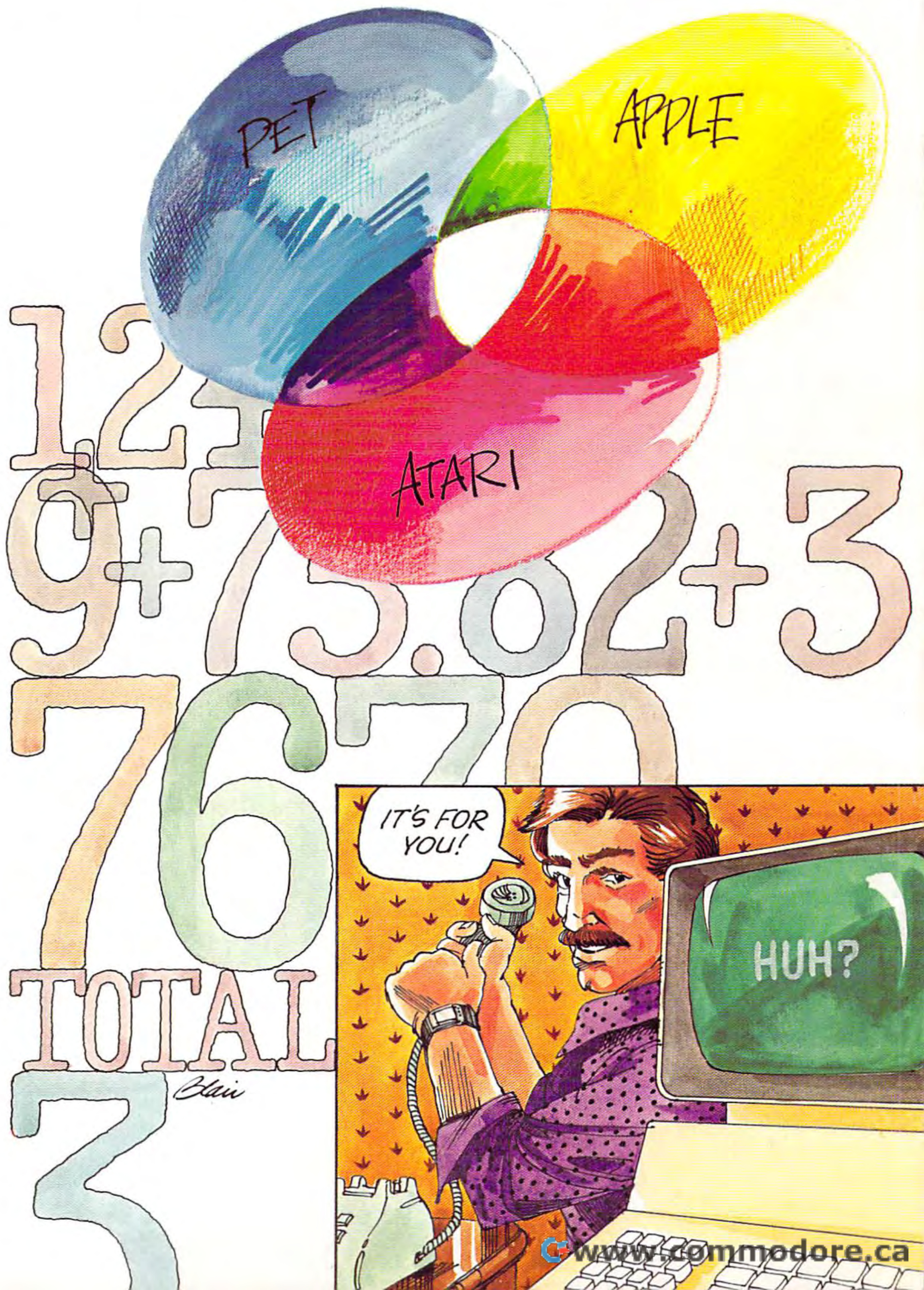
$2.50
September, 1981
Issue 16
Vol. 3, No. 9
63379

The Journal For Progressive Computing™

PET, APPLE, ATARI: On Speaking Terms
Converting BASIC Programs

The Column Calculator
A Screen Scrolling Calculator For Your Computer

What Is A Modem, And Why Do I Need One?
Telecommunications And Your Personal Computer: First In A Series



0 74470 63379 09

www.commodore.ca

Mountain Computer CPS MultiFunction Card™

The Only Interface Card You Need!

Connecting a Parallel Printer?

Epson®- Centronics®- IDS Paper Tiger® CPS handles all these printers and others with on-board intelligence to provide paging and other features found on no other card.

Connecting a Serial Printer?

Diablo®- Qume®- NEC - TI 800 Series® CPS handles these printers and others with standard RS-232 interface providing selection of baud rates, handshakes, paging, and more.

Connecting a Modem or Terminal?

Hayes Smartmodem®- Novation CAT®- M & R Pennywhistle® CPS handles these and others with full/half duplex operation, baud rate selection, and even a transparent terminal mode which includes a dual mode feature that permits printing of text to parallel printers while 'on-line' eliminating the need for special terminal software—and more.

Connecting with the Time?

The on-board calendar/clock provides real time and date information including day of week, day, month, year, hours, minutes, and seconds for any application requiring a time stamp—battery backed-up for over two years!

Connecting with The Source?

Used with a modem, CPS provides the connection to information utilities, such as The Source,® Dow Jones, and others. Additionally, CPS provides the connection to big-time electronic mail with programs such as Micro Courier® and Micro Telegram,® and other data transfer programs.

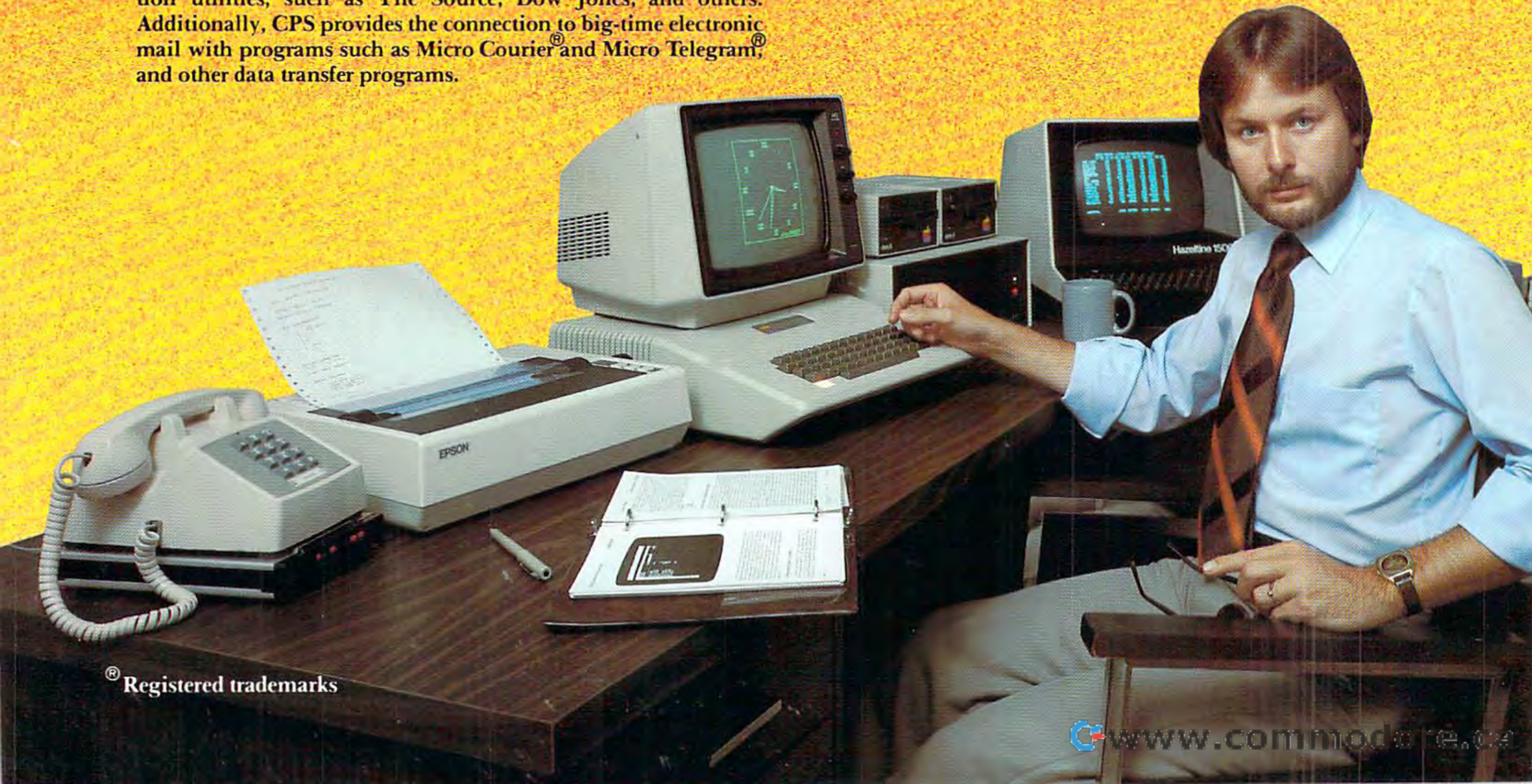
Connect with Easy Use!

CPS has no switches to set! All functions on the card are set from a user program. Menu driven screens set up your choice of all functions on CPS and store them on-board in CMOS RAM—battery backed-up (including the clock) for over two years! To change parameters, run the set-up program again—or use special commands from your keyboard. Furthermore, most existing software programs are immediately usable with CPS. *Phantom Slot Capability* permits assignment of CPS' functions to your software's pre-defined slots.

Connect with Your Apple® Dealer

Drop by your Apple dealer and see how the CPS MultiFunction Card provides the most comprehensive capabilities for RS-232C serial interface, parallel output, and real-time calendar/clock of any card available today—*all on one card*—at one low price—competitive with *any one* of the three or more single function cards that it replaces.

Mountain Computer
INCORPORATED
300 El Pueblo Road, Scotts Valley, CA 95066
(408) 438-6650 TWX: 910 598-4504



® Registered trademarks

www.commodore.ca

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

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

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

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

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

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

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

www.commodore.ca

Turn your Apple into the world's most versatile personal computer.

The SoftCard™ Solution. SoftCard turns your Apple into two computers. A Z-80 and a 6502. By adding a Z-80 microprocessor and CP/M to your Apple, SoftCard turns your Apple into a CP/M based machine. That means you can access the single largest body of microcomputer software in existence. Two computers in one. And, the advantages of both.

Plug and go. The SoftCard system starts with a Z-80 based circuit card. Just plug it into any slot (except 0) of your Apple. No modifications required. SoftCard supports most of your Apple peripherals, and, in 6502-mode, your Apple is still your Apple.

CP/M for your Apple. You get CP/M on disk with the SoftCard package. It's a powerful and simple-to-use operating system. It supports more software than any other microcomputer operating system. And that's the key to the versatility of the SoftCard/Apple.

BASIC included. A powerful tool, BASIC-80 is included in the SoftCard package. Running under CP/M, ANSI Standard BASIC-80 is the most powerful microcomputer BASIC available. It includes extensive disk I/O statements, error trapping, integer variables, 16-digit precision, extensive EDIT commands and string functions, high and low-res Apple graphics, PRINT USING, CHAIN and COMMON, plus many additional commands. And, it's a BASIC you can compile with Microsoft's BASIC Compiler.

More languages. With SoftCard and CP/M, you can add Microsoft's ANSI Standard COBOL, and FORTRAN, or Basic Compiler and Assembly Language Development System. All, more powerful tools for your Apple.

Seeing is believing. See the SoftCard in operation at your Microsoft or Apple dealer. We think you'll agree that the SoftCard turns your Apple into the world's most versatile personal computer.

Complete information? It's at your dealer's now. Or, we'll send it to you and include a dealer list. Write us. Call us.

SoftCard is a trademark of Microsoft. Apple II and Apple II Plus are registered trademarks of Apple Computer. Z-80 is a registered trademark of Zilog, Inc. CP/M is a registered trademark of Digital Research, Inc.

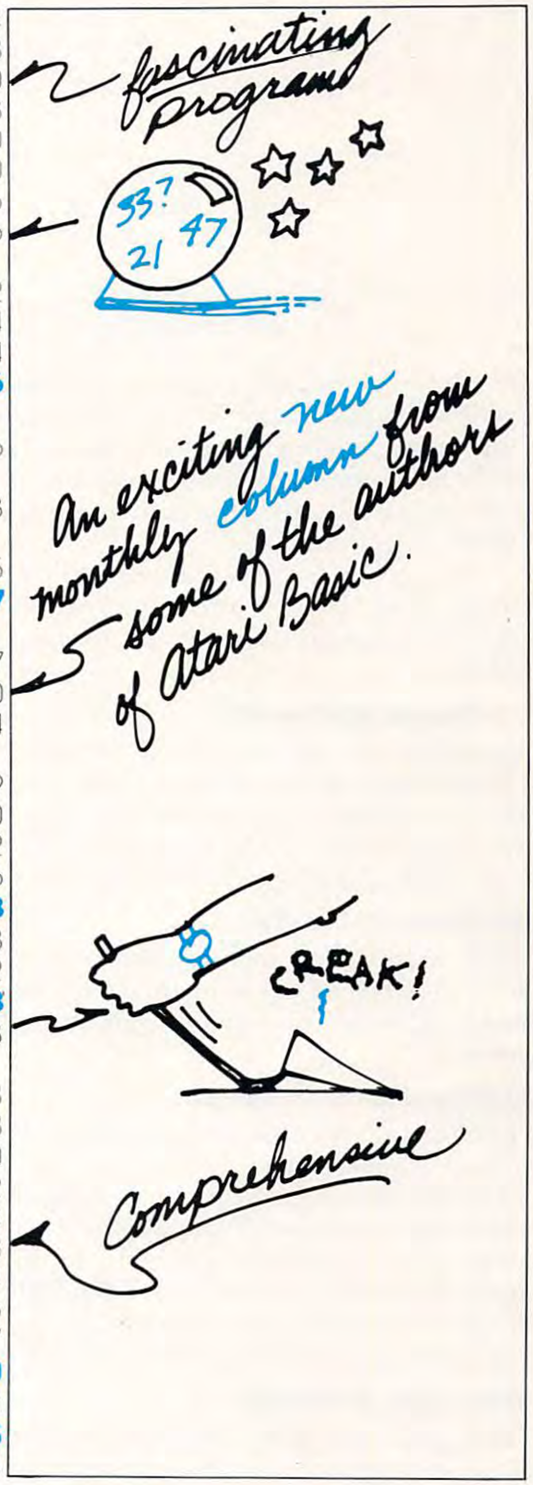
MICROSOFT
CONSUMER PRODUCTS

Microsoft Consumer Products, 400 108th Ave. N.E., Bellevue, WA 98004. (206) 454-1315

www.commodore.ca

Table of Contents

September, 1981, Vol. 3, No 8



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 200 East Bessemer Ave., Greensboro, NC 27401.

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. Controlled circulation postage paid at Greensboro, NC 27403. Application to mail at controlled circulation rates pending at Hickory, NC 28601. Entire contents copyright © 1981 by Small System Services, Inc. All Rights reserved. ISSN 0194-357X.

TOLL FREE Subscription Order Line
800-227-1617
In CA 800-772-3545
Please ask for Extension 401.

A Small System Services, Inc. Publication

COMPUTE!

www.commodore.ca

The Editor's notes...

Robert Lock, Editor/Publisher

THANKS! ... for your response with the Authors' Feedback cards. As you know, we've been slowly (but surely!) moving our production schedule up. We've been undergoing other expansion as well, and welcome Charles Brannon to our staff as Editorial Assistant.

Our general section has been tremendously enhanced this issue by the addition of in-house generated versions of programs for multiple machines.

Computers And Society

Regardless of your interests, don't miss David Thornburg's column this issue. The program presented is fascinating and intriguing. After you test it for a while, drop us a note. We'll put together some of your reactions in an issue later in the fall.

The Power Of Brevity

We've used David's introductory program to help define the rest of our issue this time. You'll find short, extremely powerful programs in the later articles.

And The Beauty Of Length

Our Atari readers will be happy to find what we feel is the most comprehensive Atari memory information ever published by a magazine. It's all embedded in the program titled SHOOT, and we're quite excited by the wealth of information. We had planned to hold it for ***COMPUTE!****'s First Book of Atari*, but Richard and Charles were too excited to wait and I deferred to their enthusiasm.

Writing For COMPUTE!

On page 54 you'll find our style sheet, with instructions and guidance for those of you interested in adding your contribution to **COMPUTE!** Needless to say, your contribution as readers is appreciated. We recently sent reader surveys out to 1,352 randomly selected subscribers. The response? Overwhelming. As of this writing, our return rate is approaching 70%. Thank you all for taking the time to answer, and a special thanks to the three of you who somehow, missing the postage paid return envelope, supplied your own.

Bit Copiers Revisited — A Resurgence?

Several months back we noted some of the problems associated with the wave of duplicating software coming into the marketplace. At that time we also discussed the needs the user/consumer regarding the right of back-up. The revised copyright law (amended December 12, 1980) clearly reinforces the right of the owner of a copy of a program to make "archival" or back-up copies. (See full text from last issue's editorial.)

If the software houses currently marketing copy-protected software don't move to establish clearly consumer-protective back-up rights, we would expect to see a growing, and quite legitimate market for programs such as Locksmith. We'd be interested in hearing of existing vendor approaches to the problem of user back-up. ©

Coming In October: COMPUTE! Overviews

COMPUTE! has a new idea in software reviews. We call it the Overview. The October **COMPUTE!** will feature a comparative review of two major word processors for the Atari: Letter Perfect and Text Wizard. **COMPUTE!**, in an effort to provide useful, objective reviews, has assembled panels of reviewers whose independent opinions will be merged into a single, large review, a **COMPUTE!** Overview.

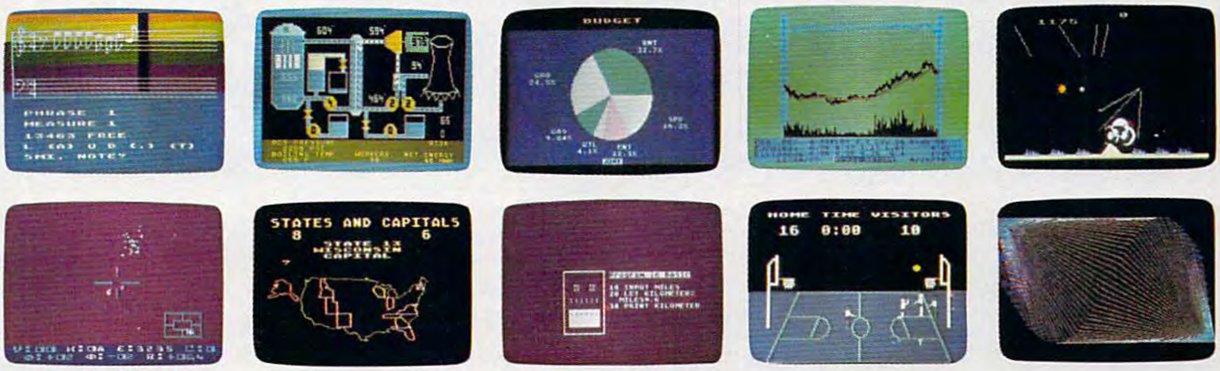
The panelists were selected for their special knowledge of the target environments of the software they will test and analyze: doctors will examine medical packages, lawyers legal software, and so forth. We hope that the new, multiple-reviewer Overview will offer the readers of **COMPUTE!** the most balanced and comprehensive analysis possible. We expect that **COMPUTE!** readers will then be able to make informed, cost-effective software purchases. Look for the Atari word processors Overview in the October **COMPUTE!**.

Kathleen Martinek,
Review Coordinator

www.commodore.ca

THE GRAPHIC DIFFERENCE

BETWEEN ATARI® COMPUTERS AND ALL OTHERS.



3.7 million reasons why the ATARI Personal Computer is something to see.
The display screen used with our computers is composed of 192 horizontal lines, each containing 320 dots. Delivering color and luminosity instructions to each dot for a second requires 3.7 million cycles... a lot of work for the normal 6502 processor.

That's why the ATARI computer has equipped its 6502 with its own electronic assistant. It's called ANTIC, and it handles all the display work, leaving the 6502 free to handle the rest. What this means to you is uncompromisingly spectacular display capabilities without loss of computer power needed to carry out the demands of your program.

That's a quality you just don't find in ordinary personal computers. And it's one of the reasons some computer experts say that ATARI computers are so far ahead of their time.

There's more... which is what you'd expect from ATARI.
Language. The ATARI Personal Computer uses several programming languages to give the user maximum control of its extraordinary capabilities. PILOT, Microsoft BASIC,* and ATARI BASIC are understood and spoken by the ATARI computer. You'll also find our Assembler Editor cartridge indispensable for machine language programming.

Sound. An ATARI computer has four sound generators, or voices, activated by a separate microchip. This leaves the principal microprocessor chips free to perform other tasks. And you can take full advantage of this capability which is designed for easy programming.

Change. ATARI Personal Computers have been designed to make change and expansion easy. The ATARI computer has a modular operating system* that can be easily replaced as new technology develops. If you need it, memory expansion requires no more than inserting additional RAM modules.* And the ATARI ROM cartridge system also makes it easy to change languages. In short, your ATARI computer won't be obsoleted by future developments... *because it already incorporates the future.*

Sharing. To learn more about the amazing capabilities of ATARI computers, visit your local computer store for a demonstration. Or send for our Technical User's Notes, intended for the serious programmer. They are only $27 and contain a lot more information about our computers' special capabilities than most companies could tell. See your ATARI dealer, or send $30 ($27 plus $3 postage and handling), payable to ATARI, to Technical User's Notes, c/o ATARI Customer Service, 1340 Bordeaux Avenue, Sunnyvale, CA. 94086.

*ATARI 800™ computer only.

ATARI®
Computers for people.™

©1981 Atari, Inc.

A Warner Communications Company

www.commodore.ca

Robert C. Lock, Publisher/Editor
Kathleen Martinek, Publication Assistant
Richard Mansfield, Assistant Editor

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

Contributing Editors
Charles Brannon, P.O. Box 6104, Greensboro, NC 27405
Marvin DeJong, Dept. of Mathematics -Physics, The School of the Ozarks Pt. Lookout, MD 65726
Eric Rehnke, 1067 Jadestone Lane, Corona, CA 91720
David Thornburg, P.O. Box 1317, Los Altos, CA 94022

Georgia Papadopoulos, Art Director/ Production Manager
Terry Cash, Typesetting/Production Assistant
Harry Blair, Director, Advertising and Promotion
Joretta Klepfer, Manager, North American Retail Sales
Alice S. Wolfe, Manager, Foreign Retail Sales
Bonnie Valentino, Circulation Assistant
Sonja Whitesell, Office Assistant
Dai Rees, Shipping Department

Subscription Information
(12 Issue Year):
COMPUTE! Circulation Dept.
P.O. Box 5406
Greensboro, NC 27403 USA
U.S. $20.00
Canada $25.00 (U.S. funds)
Europe: Surface Subscription, $25.00 (U.S. funds)

TOLL FREE
Subscription
Order Line
800-227-1617
In CA 800-772-3545
Please ask for Extension 401.

Canadian Retail Dealers should contact:
Micron Distributing
409 Queen Street West
Toronto, Ontario M5V 2A5
(416) 361-0609

Advertising Sales

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

Southern California, Arizona, New Mexico
Jules E. Thompson, Inc.
2560 Via Tejon
Palos Verdes Estates, CA 90274
213 378-8361
Jo Ann Sullivan

Northern California, Pacific Northwest, Rocky Mountain States, Texas, Oklahoma
Jules E. Thompson, Inc.
1290 Howard Avenue, #303
Burlingame, CA 94010
In Texas or Oklahoma call our Houston number: 713 731-2605
Elsewhere: 408 354-5553 or 415-348-8222
Phoebe Thompson

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

New England, New York State
The Gittelman Company
Statler Office Building
Suite 582
20 Providence Street
Boston, MA 02110
617 451-0822
Joan Donahue

New York City Metro Area, Mid-Atlantic and Southeastern States:
Local Numbers:
New York 212-567-6717
Atlanta 404 523-1252
The Gittelman Company
Summit Office Centre
7266 Summit Avenue
Fort Washington, PA 19034
215 646-5700
Doug Johnson

If you're in the **Midwest, we're now represented by GB & Associates.** You can reach them through the office of:
GB & Associates
P.O. Box 335
Libertyville, IL 60048
312 362-1821
Gordon Benson

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

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

Telephone: (919) 275-9809

Authors of manuscripts warrant that all materials submitted to COMPUTE! are original materials with full ownership rights resident in said authors. By submitting articles to COMPUTE!, authors acknowledge that such materials, upon acceptance for publication, become the exclusive property of Small System Services, Inc. No portion of this magazine may be reproduced in any form without written permission from the publisher. Entire contents copyright © 1981, Small System Services, Inc. Programs developed and submitted by authors remain their property, with the exception that COMPUTE. reserves the right to reprint the material, as originally published in COMPUTE!, in future publications. Unsolicited materials not accepted for publication in COMPUTE! will be returned if author provides a self-addressed, stamped envelope. Program listings should be provided in printed form (new ribbon) as well as machine readable form. Articles should be furnished as typed copy (upper and lower case, please) with double spacing. Each page of your article should bear the title of the article, date and name of the author. COMPUTE! assumes no liability for errors in articles or advertisements. Opinions expressed by authors are not necessarily those of COMPUTE!.
PET is a trademark of Commodore Business Machines, Inc.
Apple is a trademark of Apple Computer Company.
ATARI is a trademark of Atari, Inc.

Apple World

3-D ANIMATED COLOR GRAPHICS
Written in machine code
The program made famous on national T.V.!
by Paul Lutus

APPLE WORLD turns your Apple into a sophisticated graphics system capable of creating animated three-dimensional color images, projecting them in true perspective on the screen, rotate them, move them closer, further away, and many other exciting and imaginative things

Draws objects with 65,000 points per side.

A powerful screen-oriented text editor is included to facilitate image fomation. This program was recently featured on Tom Snyder's Prime Time Saturday TV Show and is now available for sale.

APPLE WORLD'S powerful editor is so easy to use that children will love it. You can now "sketch" your dream house, boat, car, or fantasy empire. Then view it as it would be seen from 10,000 feet, or you can ZOOM in until the screen is filled with a doorknob. You could then go inside and move from room to room examining furniture placement as your screen rotates within the room. Images or specific parts of images can easily be saved to disk or printer

Does all this sound like science fiction?
You won't think so after you have visited **Apple World.**

INTRODUCTORY PRICE $59.95
36 page manual included
For 48K Apple II or Plus with Disk

3—D Supergraphics

& 3-D GAME DEVELOPMENT SYSTEM IN COLOR
by Paul Lutus

Watch colorful butterflys, birds, fly across your Apple or Atari screen with true 3 dimensional perspective. Have rocket ships fly out at you in this incredible high speed graphics package. 3-D SUPERGRAPHICS™ is a 6502 machine language program that will interface to your Basic or machine language programs or games using simple "DOS-like" commands.

Features include:
- Simple image entry through editor
- Objects up to 256 points per side
- Uses all hi-res colors
- Allows mixed colored text & graphics for prompts and captions
- Translates on 3 axes
- Individual axis scales
- 21 different commands
- Rotate object 1.4° to 360° increments at machine speeds

FOR 48K APPLE II OR PLUS WITH DISK II $39.95 FOR DISK

FOR ATARI 800 WITH 40K MEMORY (DISK OPTIONAL)
$39.95 FOR TAPE

OTHER SOFTWARE

APPLE COMPUTERS

Super Space Wars	$ 9.95
States & Capitals	9.95
Moving Point Average	19.95
Stock Options	24.95
Finance	12.95
Bonds	12.95

COMMODORE PET

Stock Options	24.95
Finance	12.95
Bonds	12.95
Stock Analyzer	22.95
Mortgage	14.95
Space Intruders ("Best Game of 1979")	19.95
Jury/Hostage	9.95
Kentucky Derby/ Roulette	9.95
Alien I.Q./Tank	9.95
Submarine Attack	9.95
Battle of Midway	7.95
Laser Tank Battle	9.95
Swarm	14.95
Baseball	9.95
Super Startrek	14.95
PET Music Box	29.95
Music Composition System	19.95
Pearl Harbor Adventure	14.95
Super Gomoku	9.95

www.commodore.ca

Relational Query System For Management

REQUEST™

DATABASES: You've Heard The Hype Before...

The Truth IS... REQUEST DELIVERS!

DATABASE MAINTENANCE—

- Uses sophisticated screen formatting & data entry, like on IBM 3270's!
- Generates it's own screens automatically!
- Handles records up to 4K in length, using multiple screen "Pages"!
- Automatic data compression for increased disk capacity
- Uses Superkram (See below) access method for incredibly fast access, LESS THAN .2 SECONDS FOR A RECORD!
- Automatic index creation/maintenance
- Automatic maintenance capabilities
- "Goof-Proof" error handling
- Input can come from VISICALC™ or SOURCE™

DATABASE SELECTION—

- Uses screen masks to form query
- Provides extensive search capabilities
- Search arguments can include arithmetic/boolean functions, multi-field comparisons
- Queries can generate input for automatic database maintenance
- Queries can be stored in "Query Library" and executed from menu on demand
- Any number of fields can be queried concurrently
- Query output can be routed to disk, CRT report formatter, VISICALC™ or SOURCE™

ONLY $225

DATABASE REPORTING—

- Automatic headlines
- Automatic field editing
- Report fields can be calculated, sub-totaled & cross-footed in any manner desired.
- Optional counter breaks may be set
- Automatic grand totals
- Automatic statistics

REQUIREMENTS

Superkram (see below) and: Commodore Pet 32K (40 or 80 col.) and 2040/4040/8050 disk OR Apple II 48K with Applesoft or language system and 2 disk drives or CORVUS.

SUPER KRAM™

Now With Multi-Key Capabilities For Apple & Pet

by Ken Germann

Since KRAM™ was introduced in 1979 it has fast become known as the quickest and most powerful access method for serious Apple and Pet users. Now, after hundreds of requests we have added MULTI-KEY, MULTI-INDEX, functions, as well as increasing processing speed.

IBM/370 users have VSAM (Virtual Storage Access Method) to provide fast, flexible keyed-access to their data. Now SUPER KRAM (Keyed Random Access Method), from United Software of America, gives Apple and Pet users the same flexibility, substantially increasing the processing power of the Apple and Pet.

Until SUPER KRAM the only "random access" capability in the Apple and Pet consisted of a crude form of "relative record" processing. While this is usable for very simple applications, it falls far short of the needs of today's business and analytical applications. Using SUPER KRAM records may be processed by any one of multiple "Key" values, which may consist of any kind of data: numbers, letters, special characters, etc. Even Apples's long-awaited DOS 3.3 doesn't have anything like this!!

KRAM™ 2.0 Only $99.95

SUPER KRAM™ Only $175

KRAM™ 2.0 Regular Features

- Written in 6502 machine code Basic compatible
- Create/Open a dataset
- Put record by key
- Add & delete records by key
- Get any record by Full/Partial key
- Access by any key in as little as .2 sec. (.1 sec. with Corvus disk)
- Supports multiple disks
- Read next or previous record
- Dynamic space allocation
- Dynamic space reclamation
- Dynamic index compression
- Files never need reorganization
- Compatible with language systems

NEW IMPROVED KRAM™ 2.05

ATTENTION-EXISTING KRAM USERS,
Send $15 with original disk and ROM to United Software for improved version of Kram.

SUPER KRAM'S™ Added Features

- MULTIKEY SUPPORT — Allowing simultaneous access to a KRAM file by more than one key field.
- HI-SPEED READ — This feature allows increased I/O speed up to 60% faster during processing of SUPER KRAM read next, read previous, put and delete requests.
- IMPROVED INDEX ARCHITECTURE — Allowing faster index searchers and more efficient disk space utilization.
- INTEGRATED BASIC COMMANDS — Allowing SUPER KRAM™ commands to be coded in-line with Basic, providing easier usage of KRAM than ever before.
- USER-SPECIFIABLE BUFFER POOL — Allowing the user to specify how many KRAM files are allowed open at one time; will support any number of KRAM files.
- LOGICAL RECORDS (KEYS MAY BE NON-UNIQUE) — Records added to the KRAM files are immediately accessible by any of the defined keys for the file (Automatic Upgrade).
- KRAM 2.0 files are totally compatible with SUPER KRAM

USA UNITED SOFTWARE OF AMERICA

750 3RD Avenue,
New York NY 10017
(212) 682-0347 ***Telex 640055***

Look for the RED-WHITE-BLUE United Software Display at your local computer dealer, or send check or moneyorder, plus $3.00 shipping to:

DEALER INQUIRIES INVITED

REQUEST & KRAM are trade marks of United Software of America

www.commodore.ca

Ask The Readers

Robert Lock, Richard Mansfield
And Readers

Thanks to the many **COMPUTE!** readers who sent in answers, this month's column contains both new questions and answers to questions raised in previous issues.

Answers:

*"I am writing in response to the question posed by "A Reader" about how to modify 'Index' by David Wilcox to work on the new 4.0 CBM machines (**COMPUTE!** #14). I use this program extensively on my C-30 cassettes and, when I got a new CBM machine recently, I had the same problem. The fix only requires two changes on line 210 of the 'Index' program. Line 210 checks the #1 cassette status switch with a PEEK(519) and sets it and the cassette motor control register so the motor is off. The problem with the new 4.0 machines is that the cassette status switch has been moved. The only change needed to fix 'Index' for new machines is to change the PEEK(519) and POKE 519,52 on line 210 to PEEK(249) and POKE 249,52 respectively." David Swaim*

*"I would like first to respond to Tracy Principio's question about full screen graphics on the ATARI: creating any soft of graphics display in machine language requires rewriting the display list, which unfortunately, is a topic which is beyond the scope of my letter. (Boy, that sounds like a cop out, doesn't it?) I would refer Tracy to issue #6 of **COMPUTE!** page 71 for an excellent article on the subject.*

Another question is particularly bothersome to me because I own an APPLE myself and I don't believe it is possible to make the "mistake" of buying an APPLE. Several things can be done to reduce radio frequency (rf) interference on APPLEs. Most interference comes from the use of rf modulators. Often, using a video monitor with a short, shielded cable will cure the problem." Erann Gat

Questions:

"I recently got a Commodore VIC-20 computer and I have been absolutely delighted with it (even with the limitation of a twenty-two character line). However, I ran into a problem that perhaps somebody can help me with.

I have a program that executes in two phases. Phase I is 'saved' on a cassette tape and is followed immediately with Phase II which has also been 'saved.' During execution, Phase I completely finishes and at the end, the last instruction executed is:

9999 LOAD "PHASE II"

Everything seems to work alright except that it doesn't completely load the second phase and I end up with 'undefined errors' because of the truncated program. Yet, if I load the second phase manually, (i.e., by entering the immediate command, " 'LOAD PHASE II' "), it will load successfully.

Does anybody know what's wrong? I have tried everything but cannot get it to work. Please help!" Stanley Berlin

When one program "overlays" another, the first program must be longer than the second. When you ask for a ?FRE(1), the number must be larger for the program which calls a second program in.

"Can anyone tell me where to locate the producers or a copy of the manual for Altair 8800b Microcomputer System operating under Altair's Revision 4.1 of their Disk Extended BASIC?" Reinaldo Jiminez

"I have an interesting question to raise. As we have all come to accept by now, there isn't going to be a next generation 6502; a 6516, a 6509, or whatever you want to call it. There are some who feel that it will be around for a long time to come. Others think it will fade rapidly in the face of newer machines that have finally begun to emulate some of its advanced features, and outperform it.

The question is, then, should 6502 fans go down with their ship, or hop a ride on another? If the choice is the latter, is there a better alternative than the 6809? Already there is at least one 6809 card available for the APPLE II. The TRS-80 model III uses the 6809. Commodore is making a 6809 card available for its new mini-mainframe computers. Synertek is now offering a plug-in module to replace the 6502 in the SYM board, complete with the SYM's beautiful operating system.

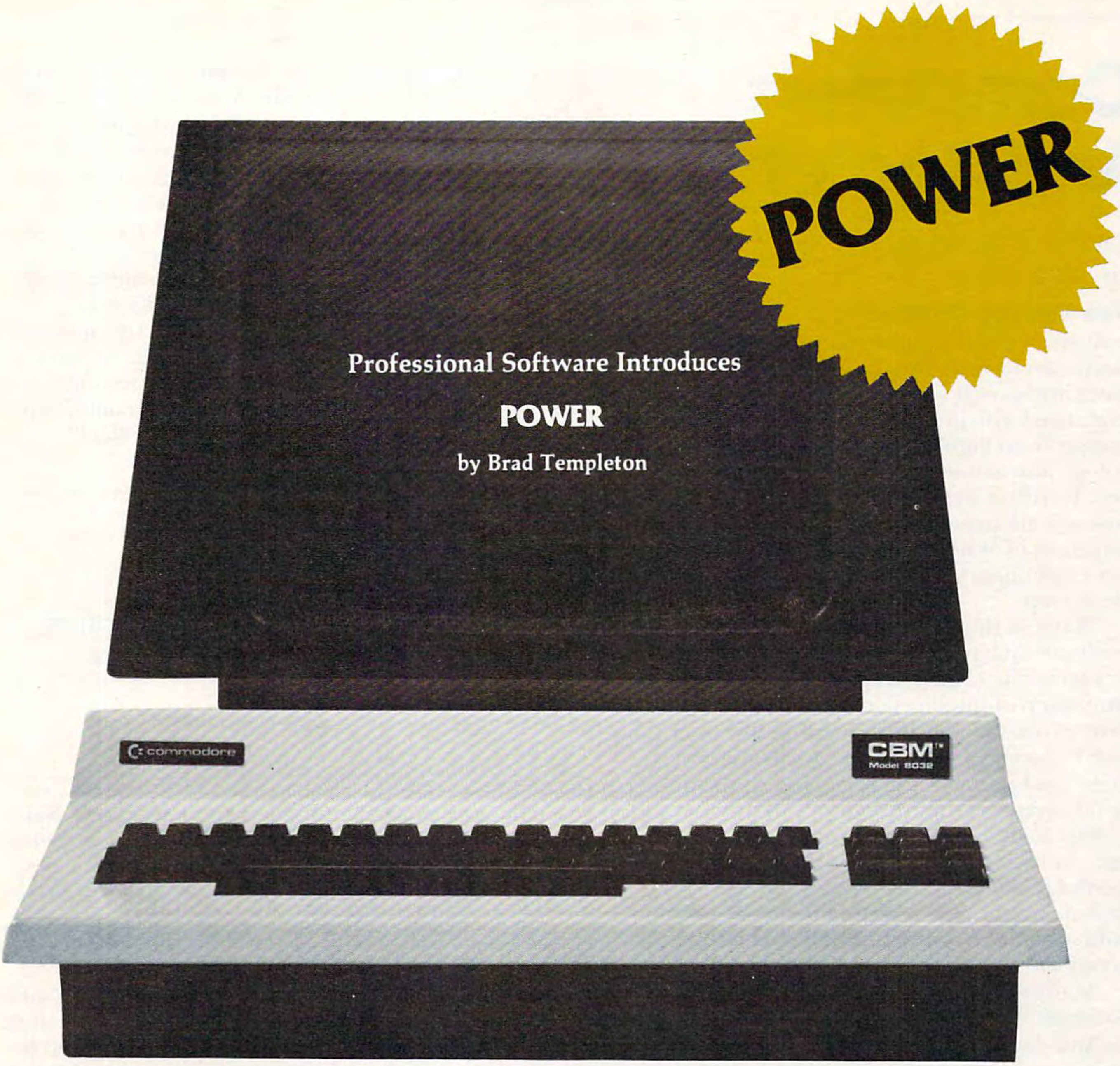
For those using assembly language, the change from 6502 to 6809 is not that traumatic, since both descend from the 6800. In some ways, 6502 users will adapt easier than 6800 users to the 6809, since they are already used to indirect addressing. For those using BASIC or another High Level Language (HLL), the change is painless, since the HLL is transparent to the processor anyway.

For both kinds of users there are some definite performance advantages to the 6809. The trend in hardware is to build processors that can more directly handle HLLs. While the 6809 is still rather conventional, its second "user" stack provides a significant edge over the 6502 for implementing threaded languages like FORTH.

*The next logical question is: should **COMPUTE!** expand its horizons and begin to provide information and articles about the 6809 and 6809-based systems? That's a tough one. Perhaps the readers should be allowed to respond and say what they think. The time to start considering this question is now." Eugene M. Zumchak*

We're not necessarily prepared to accept the point that there will be no second generation 6502; nonetheless Gene Zumchak, a **COMPUTE!** columnist, raises an important issue here. Let us know what you think. Anyone voting for the Motorola 68000? ©

www.commodore.ca



ADD POWER TO YOUR COMMODORE COMPUTER

$89.95

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

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

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

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

www.commodore.ca

Computers And Society

David D. Thornburg
Los Altos, CA

A Few Thoughts On Thinking ...

As readers of this column may have noticed, I have thus far avoided writing about machine intelligence. There are several reasons. For one thing, this topic is discussed with great regularity in other magazines ranging from highly technical computer and psychology journals to mass market magazines such as *Time*. Hardly a week goes by, it seems, without some solemn pronouncement emanating from the University of Wherever that the medium for the next evolutionary step in intelligence will be the silicon chip.

Many of those who forecast this extraordinary development have been making the same prediction for years. The controversy surrounding the mechanization of intelligence is not new. In fact, one of the earliest major discussions on this topic took place between Charles Babbage (inventor of the Analytical Engine — the precursor of the modern digital computer) and Ada Byron (Lady Lovelace) in 1842. Lady Lovelace worked closely with Babbage, and became the world's first "systems programmer." Babbage was of the opinion that his machine would have a feeling for numbers, that it could someday be made to think. Ada Bryon disagreed with him most strongly on this issue.

Many of you are probably saying to yourselves, "So what?" After all, the philosophical issue of machine cognition doesn't affect the utility of computers, so the "mechanized brain" controversy doesn't have much practical impact on anyone.

Well, maybe it doesn't, and then again maybe it does. One of the characteristics of human beings which separates us from toadstools is our ability to question the origins and nature of our own existence. The fervor with which people agree (or disagree) with the idea that computers can be made to think suggests that, regardless of practical issues, this philosophical question is quite important to many people.

I have two reasons for spending some time on this subject this month. First, it was my pleasure to be a panelist on this topic at the National Computer Conference held in Chicago last May. The spirited debate between David Ahl, Ernest Kent, and myself gave me much to think about. Second, I recently read a short story which resulted in a computer program which raises some interesting questions on the nature of intelligence — human or otherwise.

The question, "Can Machines Think?" was the topic of a panel chaired by Abby Gelles at the Personal Computing Festival held in conjunction with the NCC in Chicago. David Ahl, publisher of *Creative Computing*, presented the view that machines couldn't think because the richness and depth of human experience was too great for any machine to handle. Basically, the absence of a social, historical, and evolutionary context creates fundamental limitations on the capability of mechanized thought. Furthermore, Ahl suggested that thought and "feeling" are related. To have a conscious thought first requires consciousness. It is as ridiculous to suggest that a machine could "think" as it is to suggest that a machine could "feel" loneliness or love.

The controversy surrounding the mechanization of intelligence is not new.

The second speaker on the panel was Ernest Kent, a professor of psychology and psychopharmacology at the University of Illinois (Chicago). Kent's view (which is also expressed in his fine book *The Brains of Men and Machines* (Byte Books, 1981)) is that the brain is a machine, and that the mind and the brain are one and the same thing. Kent's book presents a model of the human brain which is quite understandable to computer-literate readers, since it deals with an electronic computer simulation for the brain's activity. Even if one rejects the idea that computers can be made to exhibit conscious thought, there is merit in exploring the limits of mechanized models of the brain. After all, our knowledge of the aerodynamics of bird's wings helped man to build flying machines, even though these machines do not fly the way birds do.

In opposition to David Ahl, Kent suggested that there was much evidence for the mind and the brain being one and the same thing. For example, electrical measurements of brain activity can be traced to specific thoughts. Furthermore, Kent said that feelings were not as mysterious as Ahl would have us believe. He said that he knew just where to probe in Ahl's brain to elicit a "feeling," and that, if the probe signal were strong enough, he would never experience that feeling again.

While I found much of Kent's work on a model for the brain quite fascinating — especially his idea that the model consists of many millions of processors all highly interconnected with thousands of

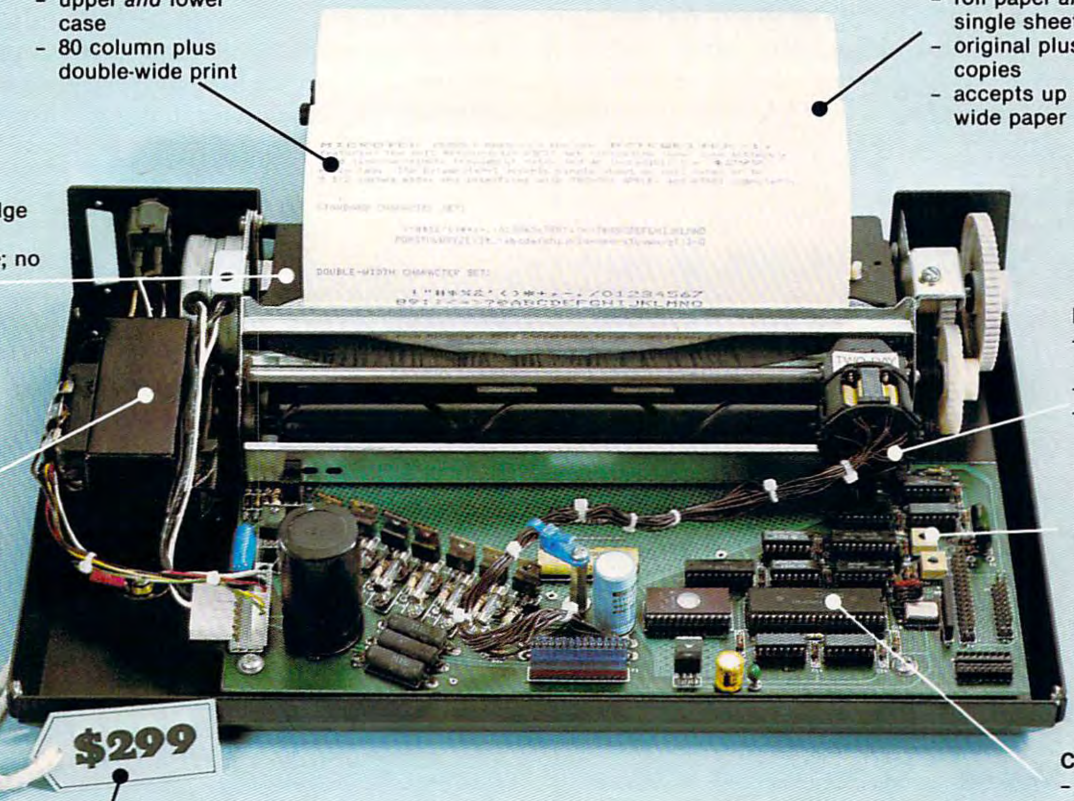
www.commodore.ca

TECHNOLOGY!
Who says it has to be expensive?

Character Set
- upper *and* lower case
- 80 column plus double-wide print

Paper
- roll paper *and* single sheet
- original plus 3 copies
- accepts up to 9½″ wide paper

Ribbon
- long life cartridge ribbon $9.95
- easy to change; no mess



Printhead
- 100 million characters $29.95
- disposable
- 7 wire bi-directional impact matrix

Transformer
- heavy duty power transformer for longer life

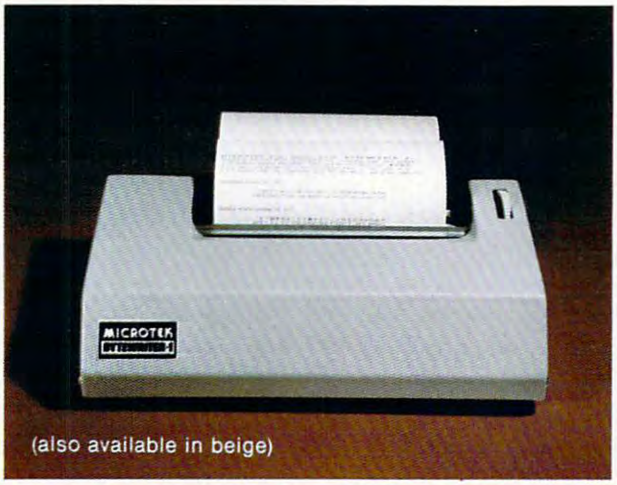
Trimpot
- for fine adjusting of vertical alignment

$299

Controller
- state of the art single chip 8039 microprocessor

Price
- high quality printer for $299.

BYTEWRITER-1



(also available in beige)

With a highly engineered circuit board and software program, backed by a rugged mechanism, the Bytewriter-1 makes no compromise with quality.

Call us for more information today — and be sure to ask us about our 30-day money-back guarantee.

MICROTEK inc.

Microtek, Inc.
9514 Chesapeake Drive
San Diego, CA 92123
Tel. 714-278-0633
Outside Calif. call toll free: 800-854-1081
TWX. 910-335-1269

www.commodore.ca

their neighbors — his suggested connection between the brain and the mind bothered me.

In my talk, I suggested that someone who had no prior knowledge of radios could study one and draw a most interesting conclusion. To someone who knew nothing about radio transmission, the radio appears to have two parts — its physical embodiment, and the programming which comes out of the loudspeaker. After careful study, our naive person has decided that the radio and its programming are one and the same thing. For example, the music coming out of the loudspeaker can be traced throughout the radio as it is being played. Furthermore, our new radio expert may have discovered that a probe signal in certain areas of the radio will elicit a response and that, if the signal is large enough, the radio will never have that response again.

I am not suggesting that the radio/program brain/mind analogy is perfect. After all, we can isolate the radio from its programming in a special room called a Faraday cage. But my point is that just because "thoughts" can be traced in that portion of the brain which we can model, we still have no proof that the mind and the brain are one.

... we still have no proof that the mind and the brain are one ...

Twenty Questions: Atari Version

```
100 REM *20 QUESTIONS
105 DIM A$(40),B$(6)
110 PRINT "}WELCOME TO THE GAME OF TWENT
Y"
120 PRINT "QUESTIONS.  BY ASKING QUESTIO
NS WHICH"
130 PRINT "HAVE YES OR NO ANSWERS, TRY T
O GUESS"
140 PRINT "THE OBJECT WHICH HAS BEEN SEL
ECTED."
150 PRINT
160 PRINT "BE SURE TO END EACH QUESTION
WITH A"
170 PRINT "QUESTION MARK."
180 PRINT
190 PRINT
195 B$="AEIOUY"
200 C=0
210 REM *ROUND
220 C=C+1
230 REM *QUESTION
240 PRINT "ENTER QUESTION #";C
250 INPUT A$
260 IF A$(LEN(A$))="?" THEN 290
270 PRINT "THAT ISN'T A QUESTION.  PLEAS
E ASK A  QUESTION."
280 GOTO 230
290 YES=0:NO=1
300 FOR I=1 TO 6
310 IF A$(LEN(A$)-1,LEN(A$)-1)=B$(I,I) T
HEN YES=1:NO=0
320 NEXT I
330 FOR PAUSE=1 TO 50*RND(0):NEXT PAUSE
340 IF YES THEN PRINT "YES"
350 IF NO THEN PRINT "NO"
360 PRINT
370 IF C<20 THEN 210
380 PRINT "END OF TWENTY QUESTIONS."
390 PRINT "PRESS RETURN TO START AGAIN."

400 INPUT A$
410 RUN
420 END
```

As usual, this panel didn't resolve anything; but did raise some challenging issues. I would have been happy to let the topic die at this point had I not received another interesting book, *Tales of the Marvelous Machine, 35 Stories of Computing*, edited by Robert Taylor and Burchenal Green (Creative Computing Press, 1980). This book is a collection of short stories, some of which first appeared in Creative Computing. While much of this book is very interesting, I was particularly taken by the story "XX?S" by Brian McCue. In this story, a computer science teacher is asked to run a program which plays the game of "twenty questions." In an effort to find the object chosen by the computer, the teacher asked quite detailed questions which the computer answered with a YES or a NO, as appropriate. As the play continued, the teacher became intrigued with the apparent cognitive skills being displayed by the machine. The computer was able to answer a complex question like: WAS THE OBJECT INVENTED PRIOR TO THE YEAR OF OUR LORD MCX. The machine's ability to respond, NO, startled the teacher. Even the use of Roman numerals and elaborate dating schemes couldn't throw the computer off track. Finally, after finding the correct answer (The Wright Brother's airplane), he tried unsuccessfully to list the program. What he *did* discover was that the program was only one disk sector long.

This seemed most strange, since programs which purport to have some understanding of natural language are typically too large to run on any microcomputer.

After finding the key to the solution, I wrote my own version of this program. A typical run is

www.commodore.ca

SOFT ROM

- 4096 BYTES OF SOFT ROM
- STORE MACHINE CODE SOFTWARE BEYOND THE BOUNDARIES OF BASIC
- WRITE PROTECT RAM WITH A FLIP OF A SWITCH
- STORE VARIABLES OR INDICES OUTSIDE OF BASIC
- SOLVE THE CONFLICTING ROM PROBLEM BY SOFT-LOADING THE APPROPRIATE ROM IMAGES

The SOFT ROM is compatible with any large keyboard PET/CBM or similar 2532 EPROM systems. It may be placed in any ROM socket to give the user room for machine code. If the SOFT ROM is placed in an occupied ROM socket, the user can transfer the PET/CBM ROM into the on-board ROM socket and select between ROM and RAM to manipulate the Commodore operating system.

Since the SOFT ROM places write protectable RAM into any of the computer's ROM sockets, it is ideally suited to use as a development tool to test ROM or EPROM based software systems before they are burned in.

Examples of software presently available for the SOFT ROM includes BASIC AID, UNIVERSAL WEDGE, SUPERMON, EXTRAMON, USER PORT PRINTER (Centronics parallel) and a buffered BACKGROUND PRINTER routine.

Installation is a simple plug-in into any available ROM socket.

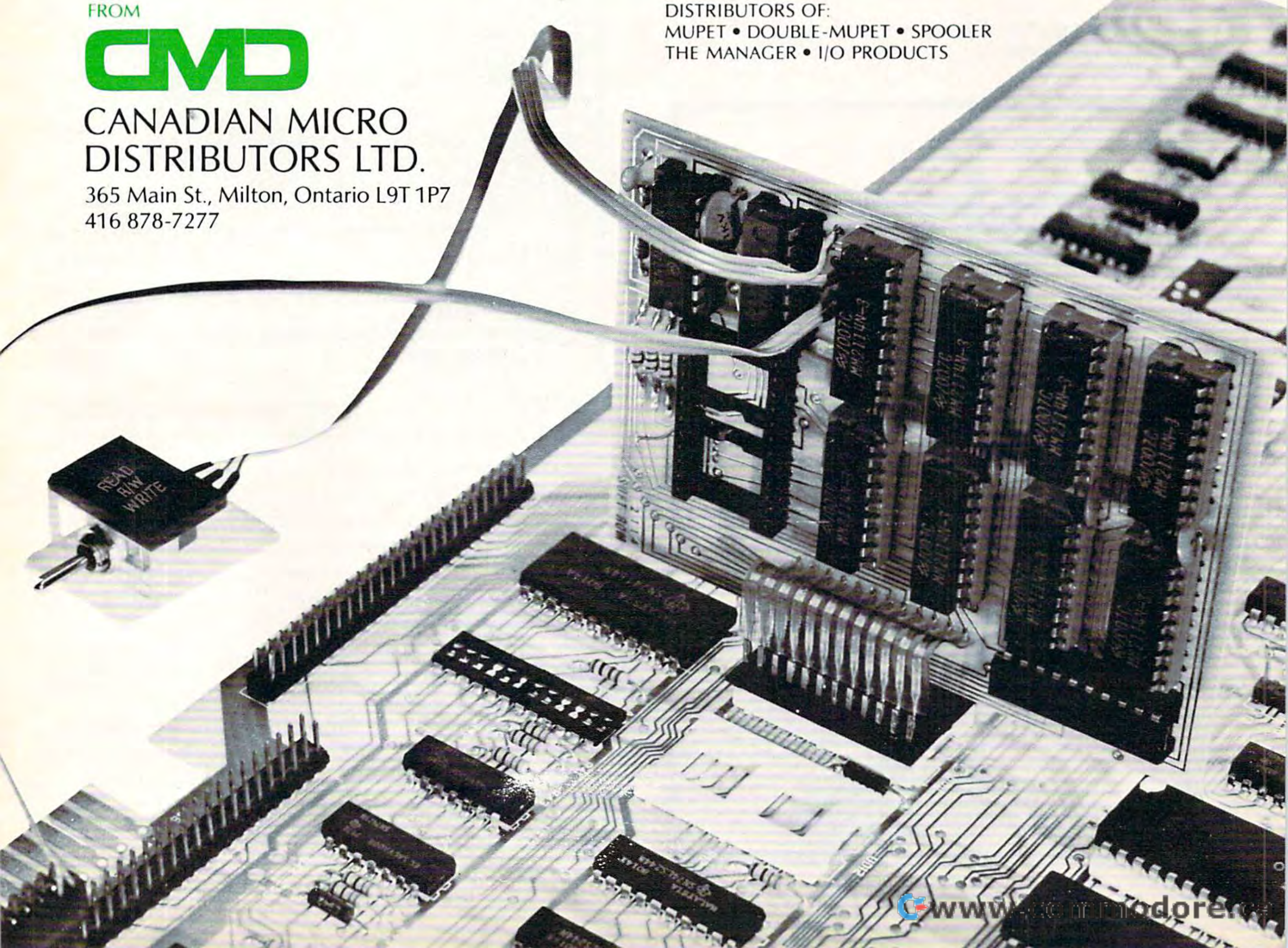
$129.00

FROM

CMD

CANADIAN MICRO DISTRIBUTORS LTD.
365 Main St., Milton, Ontario L9T 1P7
416 878-7277

DISTRIBUTORS OF:
MUPET • DOUBLE-MUPET • SPOOLER
THE MANAGER • I/O PRODUCTS



shown below:

ENTER QUESTION 1
Is it mineral?
NO

ENTER QUESTION 2
Is it vegetable?
YES

ENTER QUESTION 3
Does it grow under the ground?
NO

ENTER QUESTION 4
How about above?
YES

ENTER QUESTION 5
Are you sure?
YES

ENTER QUESTION 6
OK, is it coniferous?
NO

ENTER QUESTION 7
Does it grow on a tree?
YES

ENTER QUESTION 8
Is it green?
NO

ENTER QUESTION 9
Are people likely to make juice from it?
NO

ENTER QUESTION 10
Is it a black olive?
YES

As you can see, the right answer was found in only ten tries. The apparent language-understanding ability of this program has startled many of the people to whom I have shown it. For example, most computer scientists know that it is non-trivial to have a computer figure out that our question 4 is a modification of question 3. Some people have felt that, except for the occasional lengthy pauses between question and answer, they couldn't tell if they were playing against a computer or a human player.

By what magic does one generate artificial intelligence in such a tiny program?

By not doing anything of the sort!

The following is the complete listing (in Atari PILOT) of the program I wrote:

```
*20 QUESTIONS
T:WELCOME TO THE GAME OF TWENTY
  QUESTIONS. BY ASKING/
T:QUESTIONS WHICH HAVE YES OR NO
  ANSWERS, TRY TO GUESS THE/
T:OBJECT WHICH HAS BEEN SELECTED.
T:
T:BE SURE TO END EACH QUESTION WITH A
  QUESTION MARK.
T:
T:
C:#C=0
*ROUND
C:#C=#C+1
*QUESTION
T:ENTER QUESTION #C
A:
M:?
TN:THAT ISN'T A QUESTION. PLEASE ASK A
  QUESTION.
JN:*QUESTION
M:A?,E?,I?,O?,U?,Y?
PA:?  150 [random program delay of up to 149 "jiffies"]
TY:YES
TN:NO
T:
J(#C<20):*ROUND
T:END OF TWENTY QUESTIONS. PRESS RETURN
  TO START AGAIN.
A:
J:*20QUESTIONS
E:
```

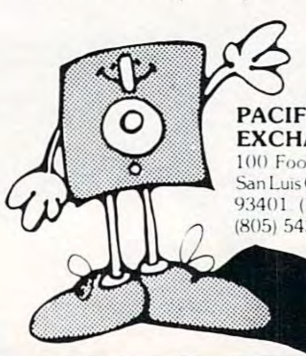
As you can see, the program is quite simple. Each question is first examined for a question mark, and is then examined (by the match command in the 17th line) to see if it ends in any of the letters A, E, I, O, U, or Y. After a random pause (the machine's "thinking" time) the answer YES is printed if a match was found. Otherwise the computer prints the word NO. (If you convert this program to BASIC, you might just want to check for words ending in E, as McCue did in his story.)

Once people see how simple this program is, they realize that they *were* playing twenty questions with an intelligent being — themselves. After all, if you asked if the object was a person and the computer said NO, you wouldn't be very inclined to ask if it was a person named Dave (to which the com-

wabash®

When it comes to Flexible Disks, nobody does it better than Wabash.

MasterCard, Visa Accepted.
Call Free: (800) 235-4137



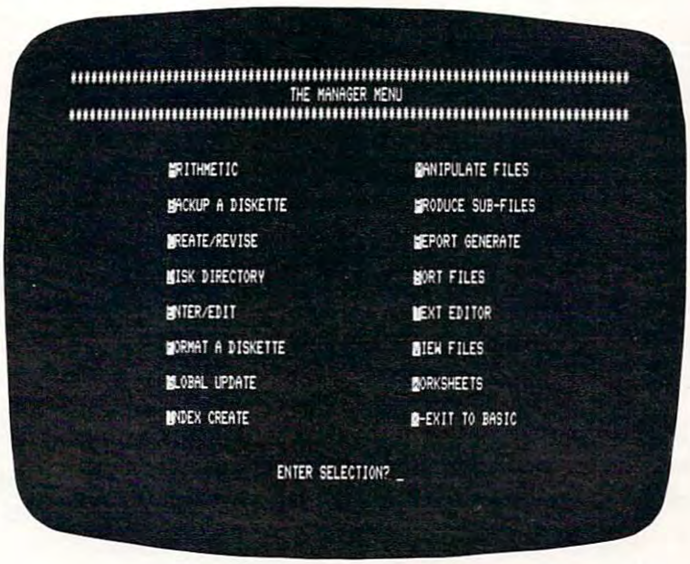
PACIFIC EXCHANGES
100 Foothill Blvd
San Luis Obispo, CA
93401 (In Cal call
(805) 543-1037)

www.commodore.ca

THE MANAGER

The first truly user-friendly Database Management System available at reasonable cost.

This suite of programs is ideally suited for both the businessman and programmer, for use with the CBM 8032.



For the Business User

- Uses Menu Options – no programming experience needed.
- Lets you enter data in the form you wish, then lets you recall it using any search criteria.
- Performs predefined calculations on the record in realtime as record is displayed on the screen.
- Reports can be produced using any search criteria and/or arithmetic functions.
- Useful applications can be developed quickly.

Typical Applications include –

- Inventory Control
- Mailing Lists
- Accounting systems
- Personnel
- Costing
- Gathering test data
- Budgeting
- Scheduling
- Examples of use included on disk supplied.

As Programmers Tool

- Uses standard PET ASCII files.
- Software interface is in Basic and available to the programmer.
- No special disk formatting so that word processing or other programs can be stored on the same disk.
- No ROM Based Security thus no need to open CPU.
- Fast 'n' key Sort/Merge included.
- Full realtime intra & inter record arithmetic performed on the screen as record is displayed.
- Professional software support including unique security available.

TRY IT!
IF YOU ARE
NOT SATISFIED WITHIN 30 DAYS
WE WILL RETURN YOUR MONEY

$250.00

CMD CANADIAN MICRO DISTRIBUTORS LTD.

365 Main St., Milton, Ontario L9T 1P7
416 878-7277

DISTRIBUTORS OF:
MUPET • DOUBLE-MUPET • SPOOLER
THE MANAGER • I/O PRODUCTS

www.commodore.ca

puter would, of course, say YES). The "sense" of this program perfectly matches that of the player. It is *your* knowledge of objects such as trees, and that olives grow on trees, and that olives can be black, which makes the game work. The computer doesn't *know* anything.

The reason this is relevant is because some people who have seen programs of this type, and programs which produce computer generated poetry, feel that this is an example of mechanized "creativity," and thus an example of artificial intelligence.

There is not much effort required to write a program which generates certain poetic forms, choosing words from the appropriate parts of speech at the right time, *etc*. But, if the result is "poetry," it is only because the human reader has decided that it is. It is the stimulation of the reader's feelings by the computer-generated text strings which gives life to a poem.

As for my personal feelings on the likelihood of there ever being a "thinking" computer, I have to agree with Ada Bryon who, in a note to Charles Babbage, writes:

> It is desirable to guard against the possibility of exaggerated ideas that might arise as to the powers of the Analytical Engine. In considering any new subject, there is frequently a tendency, first, to overrate what we find to be already interesting or remarkable, and secondly, by a sort of natural reaction, to undervalue the true state of the case when we discover that our notions have surpassed those that were really tenable. The Analytical Engine has no pretentions whatever to originate anything. It can do whatever we know how to order it to perform. It can follow analyses; but it has no power of anticipating any analytical relations or truths. Its province is to assist us in making available what we are already acquainted with.

Now *that's* something to think about!

DTACK 10

SIMPLE Motorola 68000 systems exist; you can attach one to your Apple II® or Pet/CBM® for about $600. The resulting system is the world's least expensive 68000 software development system (IF you already have an Apple II® or a Pet/CBM®) because we give away a cross assembler with our board.

We have a four function 68000 floating point package, with a logarithm routine, in microsoft Apple II® /Pet® format. The log routine executes in less than 2 milliseconds. This package is available in *SOURCE* form for only $10 plus 50 cents postage (Calif. residents add 6%).

A newsletter covering *SIMPLE* 68000 systems and attached processors is available from us for $15/6 issues. Send $2.50 for the first issue.

DTACK GROUNDED, Dept. 105
1415 E. McFadden, St. F
SANTA ANA, CA 92705

Apple II® is a trademark of the Apple Computer Company. Pet® and CBM® are trademarks of Commodore Business Machines.

Twenty Questions: Microsoft Version (Pet, Apple, etc.)

```
100 REM *TWENTY QUESTIONS
110 PRINT CHR$(147);"WELCOME TO THE GAME
115 PRINT"OF TWENTY QUESTIONS. BY
120 PRINT "ASKING QUESTIONS WHICH HAVE
130 PRINT "YES OR NO ANSWERS, TRY  TO
140 PRINT "GUESS THE OBJECT WHICH HAS
150 PRINT "BEEN SELECTED."
155 PRINT
160 PRINT "BE SURE TO END EACH QUESTION WITH A
170 PRINT "QUESTION MARK."
180 PRINT
190 PRINT
195 B$="AEIOUY"
200 C=0
210 REM *ROUND
220 C=C+1
230 REM *QUESTION
240 PRINT "ENTER QUESTIONS #";C
250 INPUT A$
260 IF RIGHT$(A$,1)="?" THEN 290
270 PRINT "THAT ISN'T A QUESTION.
275 PRINT "PLEASE ASK A QUESTION."
280 GOTO 230
290 YES=0:NO=1
300 FOR I=1 TO 6
310 IF MID$(A$,LEN(A$)-1,1)=MID$(B$,I,1) THEN YES=1:NO=0
320 NEXT I
330 FOR PAUSE=1 TO 50*RND(1):NEXT PAUSE
340 IF YES THEN PRINT "YES"
350 IF NO THEN PRINT "NO"
360 PRINT
370 IF C<20 THEN 210
380 PRINT "END OF TWENTY QUESTIONS."
390 PRINT "PRESS RETURN TO START AGAIN."
400 GET A$:IF A$="" THEN 400
410 RUN
420 END
READY.
```

www.commodore.ca

ANNOUNCING A REVOLUTION IN THE COST OF PROFESSIONAL SOFTWARE

VISACCOUNT is a fully integrated business and accounting system designed for use in small businesses. VISACCOUNT is extremely comprehensive and professional, yet it is very easy to use. The system is controlled from a series of interconnected menus permitting user-friendly operation. Everything you need to set-up and operate the system is provided with the VISACCOUNT package. Experts have estimated the development costs for a fully integrated software system ranges between $7,200 and $22,000.† When you buy software the developer has to recapture this expense. Computer Services Corporation of America is selling its software with a view that volume sales can almost negate this development cost.

OUR GUARANTEE — Buy both our software and that of our competitors (who will no doubt charge several times our price because they need to recapture their development cost). Compare the two systems and we know you'll return theirs (make sure they'll let you return their software). If you decide not to keep our system, then return it within 45 days for a full refund. Once you've used our system we're confident you'll be delighted.

†Microcomputers for Business. Applications. 1979

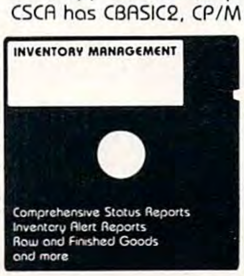
VISACCOUNT

What You Receive

- Nine 5¼" double density disks (or six 8" single density disks)
- Easy-to-use operator's manual (over 200 pages)
- Self-study guide on bookkeeping and accounting (over 180 pages)
- Cassette based instruction program on set-up and operation

Available for Apple*, TRS-80, and most others

*The Apple version requires the Microsoft Z80 softcard. CSCA has CBASIC2, CP/M and Microsfot Z80 softcard in stock.



EXTRA: MAILING LIST PROGRAM

Features

Menu Driven: The entire system runs from a single master menu which accesses numerous subsidiary menus, when needed, to perform the full spectrum of business and accounting functions.

Self-Documenting: All the information needed to use the sytem is provided in an easy to self-study format.

Requirements: 48K CBASIC2
2 DISK DRIVES CP/M

Send $159 for the VISACCOUNT system

COMPUTER SERVICES CORPORATION of AMERICA
332 East 30th Street New York, New York 10016
Order Toll Free 1-800-221-2486
In New York and Technical Number 1-212-685-0090

Name ____________________

Address ____________________

City/State/Zip ____________________

☐ Master Charge ☐ Visa ☐ American Express

No. ____________________ Expires ____________

Your System ____________________

Disk Size ☐ 5¼" double density ☐ 8" single density

© 1981 Computer Services Corporation of America

www.commodore.ca

The Beginner's Page

Richard Mansfield
Assistant Editor

Initialization

As you begin to develop a feel for programming, you will probably notice that there are four parts to most programs: initialization, main loop, subroutines, and data. In the last issue (**COMPUTE!** #14) we examined subroutines. Let's take a look at the other three parts.

If you picked several programs at random, spread them out on a table, and compare them, you would probably see certain similarities. Programs usually require some preliminary setups: variables need to be defined or DIMensioned, REM statements document (explain) what the program is going to do, the screen needs to be cleared, and so on. Before getting on with its main task (the main loop), a program will frequently need to perform preliminary jobs. This first part of a program is called *initialization*. Here is a little program which prints the names and addresses of people to whom you want to send Christmas cards. it will illustrate the four divisions of computer programs:

```
10 ADDRESSES=3                              (Initialization)
20 FOR I=1 TO ADDRESSES                     ---------------
30 GOSUB 1000                               (Main Loop)
40 GOSUB 2000
50 NEXT I
60 END      ----------------------------------------------
1000 READ NAME$ (I)
1010 PRINT NAME$ (I)
1020 RETURN                                 (Subroutines)
2000 GET K$
2010 IF K$="" THEN GOTO 2000
2020 RETURN  -----------------------------------------------
5000 DATA MYRTLE FACE / 121 TYRONE PIKE /
     NEW YORK NY 10020
5010 DATA CARL MENEFEE / 36 HAWERD ST. /
     ALAMEDA CA 92171
5020 DATA FELICE MONTREAL / 15 ACE ST. /
     RAMADA IL 80221
```

(For the Atari, add 15 DIM NAME$(80) and change lines: 1000 READ NAME$ 1010 PRINT NAME$).

Some aspects of initialization are obvious: if the program is designed to organize your Christmas card mailing list, it will need to know the total number of addresses it has on file. Part of the initialization can involve the definition of a variable which "knows" this total (line 10). Program initialization includes putting necessary information into variables, clearing the screen, defining sound, color, format (how the video or printed output will appear), and so on.

Main Loop

Often the largest section, and certainly the heart of every program, is the main loop. Like a business executive, the main loop has a job to do and sees that the assistants (the subroutines) perform their tasks correctly and in the correct order. The main loop moves down its list of subroutine jobs until the primary goal is achieved. In our example program (Christmas Card List), the primary goal is a screen display of addresses to be handwritten on the envelopes. To achieve this, the program must: 1) Read a name from the DATA table, 2) print the name and address on the screen, 3) wait until you press any key to allow it to continue, and 4) loop back to job number one.

These four jobs can be thought of as subroutines which are governed by the main routine (main loop). This executive routine, ordering and supervising its several subroutines, is often a true "loop," but need not be. That is, it often does its job over and over, looping (cycling) through the subroutines each time. Some programs only do one thing after another down a list. They do not loop. But looping is one of our major programmer's tools and most tasks lend themselves to the loop structure.

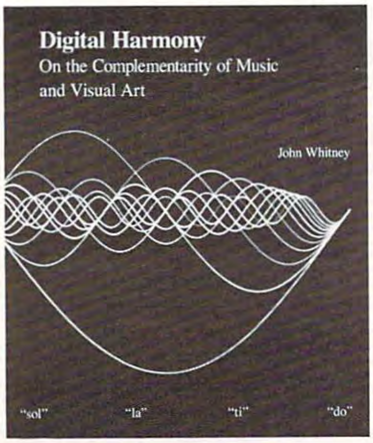
Data

Lists, arrays, tables, data base, data, DATA statements, fields, records — each of these terms are taking on specific meanings as computer jargon slowly evolves into a vocabulary of fixed meanings. But at this point in the creation of a universal terminology for computing, words are somewhat imprecise. The terms above, nonetheless, have something in common — all refer to *raw information*. The accepted word is *data*. Raw information is to a computer what raw materials are to a factory. Visualize a paper factory. Imagine that somewhere in the building (or stored nearby) is a pile of logs to be processed by the factory. At the other end of the building, the finished product, paper plates, drop onto trucks. In the same way, computers *process* information. Somewhere in the program (or stored nearby on tape or disk files) is a pile of raw information — names and addresses in our example program.

The data are *input* into the program and are processed into some computer-made product. This product is them *output* somewhere. Our Christmas program sends its product to the screen, but it could easily output to a printer which would address the cards for us. In any case, data sits in piles waiting to be computed. One of the advantages which information processing has over log processing is the ease with which new products can be generated. Inserting an alphabetizing subroutine into the Christmas program, and making a slight change to the main loop, will result in an alphabetized product. Likewise, slight changes would transform the Christmas Card Address Factory into a birthday list, a telephone directory, a record of gifts sent and received, etc.. ©

www.commodore.ca

A New Frontier Between Sight and Sound...



Digital Harmony by John Whitney

On the Conplementarity of Music and Visual Art

BYTE Books is pleased to offer *Digital Harmony*, a major new work by John Whitney, whose film art, known around the world, has been an influence on technological art and cinema special effects from *2001* to *Star Wars*. His book explores a special union of music and computer graphics and defines a new frontier between sight and sound, integrating the two to create a new art form.

Whitney shows what can be done with small computers, spells out a thorough theoretical background, and includes listings and programs for those interested in joining in the exploration of his unified field theory of art. *Digital Harmony* lays the foundation for an audio-visual art made possible by computers. It is must reading for all enthusiasts with interests in art, music, video, film, computers, education, artificial intelligence, psychology, and futurology.

John Whitney is on the Faculty of the Department of Art at the University of California, Los Angeles.

ISBN 0-07-070015-X
240 pages; hardcover
$21.95

"Digital Harmony is a peek into the future when computerization will bridge the gap between art, science, and self-understanding. I loved it."

Seymour Papert
Author, Professor
Massachusetts Institute of Technology

"Here in this beautiful book John Whitney marries art to modern technology. Scenes that previously we would have strained to imagine, Whitney can now show us."

Jearl Walker
Scientific American

BYTE BOOKS 70 Main Street Peterborough, NH 03458

Surprising Symmetries in Design and Letterforms...

Inversions:

A Catalog of Calligraphic Cartwheels
by Scott Kim
Foreword by Douglas Hofstadter
Backword by Jef Raskin

Illusion..calligraphy..visual magic—Scott Kim's new book delights the eye and enchants the mind. Filled with intriguing designs, words that read the same right-side-up and upside-down, words within words and unexpected symmetries, these compositions create a fresh way to look at the alphabet. The text includes the visual principles of symmetry, lettering and problem solving that are basic to these images. The author also draws parallels to related exercises in perception in such diverse areas as art, music, wordplay and mathematics. Scott Kim's original inversion designs first appeared in *Omni* magazine, inspiring an overwhelming reader response. An irresistible challenge, invertible writing apeals to everyone who loves beauty in mathematics and design.

Scott Kim is a doctoral student in Computer Science at Stanford University and is a concert pianist and composer.

ISBN 0-07-034546-5
128 pages; softcover
$8.95

"Scott Kim has perfected a personal art form—one with grace, elegance, subtlety, and surprises."

Douglas Hofstadter, author
Godel, Escher, Bach: an Eternal Golden Braid

"Scott Kim's *Inversions*...is one of the most astonishing and delightful books ever printed."

Martin Gardner
Scientific American

Order by mail or phone. Call toll free **800-258-5420**

Please send: ____DIGITAL HARMONY $21.95
____INVERSIONS $8.95

C981

Add 75¢ per book to cover postage and handling
Check enclosed in the amount of $________
VISA MasterCard

Card # __________ Expires ______

Name __________

Street __________

City __________ State ____ Zip ____

www.commodore.ca

What Is A MODEM, And Why Do I Need One?

Michael E. Day
West Linn OR

Getting The Information Across

MODEM is a descriptive acronym for the device that the name is applied to. MODEM stands for MOdular-DEModulator. A MODEM is used to transmit digital information such as that used by computers to a remote location. Remote in this case meaning a device not directly connected to a computer, as the remote connection can be anywhere from 6 inches away to 6 billion miles away.

Digital information, in its own controlled environment has the highest form of redundant reliability of any means of signal processing. This is because, since the signals consist of 1's and 0's (on or off), any interference which might cause a slight change of signal level so that it becomes "slightly on" is ignored, as the level is still not above the point where a 1 is assumed. This differs from an analog system where any interference is propagated through the entire system.

Since the computer only recognizes a 1 or a 0, there can be no percentage errors encountered in the signal. The information is either right or it is wrong.

An error can be detected by adding an additional piece of information to the transmission. This is called parity. Under parity control the digital information is divided up into small groups (normally on a per-character basis), and the number of 1 bits in the group are added up. In the even parity system, if there were an even number of 1 bits, then an additional 1 bit is added to the group, otherwise a 0 bit is added. In odd parity a 1 bit is added if there were an odd number of bits and a zero otherwise.

At the receiving end the receiver adds up the bits and compares its answer to the parity bit that was sent, and if there was a difference, it flags the receiving device that an error has occurred.

The parity method will catch the loss of single bits. However if more than one bit is lost, it will not always catch it. So, for large blocks of information a checksum is sometimes used. In checksumming, as each character is received it is added to the sum of the previous characters with the final sum being transmitted after the last character in the block has been transmitted. The receiver then compares this with its own sum and flags the receiving device that an error has occurred if there is a difference between the two. This type of error checking will catch about 99% of the errors encountered.

This is fine if an occasional error can be accepted. Sometimes this is not the case and every possible error must be detected to ensure the highest possible reliability. This might be required in a binary coded program being transmitted. In this case a CRC check is used. CRC (Cyclic Redundancy Check) is a special coding scheme that is different than the checksum, and can achieve better than 99.9% error detection.

Digital information... has the highest form of redundant reliability of any means of signal processing.

In all of these cases only error detection is considered. In some cases it is desirable to be able to recover the lost information. This is often a requirement inside large computer systems where no loss of data can be tolerated. In this case a Hamming code is used. In this technique, instead of a single bit being added, the length of which is dependent upon how much correction is desired; for single bit recovery, 5 bits must be added to every 32 bits of information. This will also catch all possible 2 bit data losses, and most other combinations of losses. By looking at the codes the receiving device can reconstruct the lost information and thereby remove the need to retransmit.

Generally, even parity is used in asynchronous transmissions (such as that used with a 103 or 202 type MODEM) whereas odd parity is used in synchronous transmissions (such as that encountered with the special high speed MODEMs: 2400 baud and above).

In conversational type transmissions parity is quite often not used (the parity bit being replaced with a 1 bit; to simulate a stop bit).

In block text transmissions (such as a BASIC program) it is recommended that parity be used so that any errors that might occur will be caught.

When transmitting a hex dump it is also a good idea to add a checksum to the code, as it is very difficult to see errors in this type of text. (Note: normally parity is always used when checksumming is used.)

www.commodore.ca

ANNOUNCING!

COMPUTER SHOWCASE EXPOS

The end-user market for small business and personal computer systems is really taking off. And now you can take off with it. How? By exhibiting at the COMPUTER SHOWCASE EXPOS--regional shows, heavily promoted, and produced with the consistent professionalism you've come to expect from The Interface Group.

You already know us from our **INTERFACE** show, now in its tenth year, and you know us also from our unique **COMDEX** shows, which have done so much to bring vendors and ISOs (Independent Sales Organizations) together.

Now, you and your ISOs can reach the key prospects for small systems directly: businessmen, professionals, educators, scientists, corporate executives, serious personal computer users. We understand the small systems market, because our **COMDEX** shows helped create it. Now, with the SHOWCASE EXPOS, we forge the final link in a marketing chain, designed to bring you "eyeball-to-eyeball" in a selling atmosphere with tens of thousands of prospective users!

Our first round of SHOWCASE EXPOS this autumn will take us (and you) into four major metropolitan areas. We'll follow that up with additional metro bookings in 1982 and 1983. If you market small business systems, word processors, peripherals, software, data processing services, micros, educational services, or media and supplies - you will want to exhibit and/or support your ISOs' exhibit plans. **Call our Sales Department today toll-free 800-225-4620; in Massachusetts, call 617-879-4502.**

★ **NEW YORK COMPUTER SHOWCASE**
Sept. 17-19, 1981 • Pier #92 Passenger Terminal, New York City

★ **SAN FRANCISCO BAY AREA COMPUTER SHOWCASE**
Oct. 21-23, 1981 • Brooks Hall, San Francisco

★ **SOUTH FLORIDA COMPUTER SHOWCASE**
Oct. 30-Nov. 1, 1981 • Miami Expo/Center, Miami

★ **LOS ANGELES COMPUTER SHOWCASE**
Nov.13-15, 1981 • Los Angeles Convention Center, Los Angeles

Another Exposition from THE INTERFACE GROUP
160 Speen Street, Framingham, MA 01701
Call us toll-free: 800-225-4620 (in Mass. 617-879-4502).

Producers of: INTERFACE, FEDERAL DP EXPO, COMDEX, COMDEX/SPRING, THE COMPUTER SHOWCASE EXPOSITIONS

www.commodore.ca

When binary information is being sent (non-ASCII), such as a binary core dump, a CRC check should be used (parity is normally *not* used with CRC checking), as it is very important to catch all possible errors since this type of code is essentially unreadable.

The Hamming codes are normally used with stored information where recovery or retransmission of the data is not directly possible.

All this attention to error detection is important. The computer is a very controlled environment; it is built to prevent any interference from affecting its operation. Outside the computer however, we must deal with the real world where interferences abound. Since this interference cannot be prevented, a means of detecting must be used so that the errors generated can be corrected.

How The MODEM Works

Since the telephone network was designed for analog voice transmission, it is not possible to transmit digital information from a terminal or a computer in its binary form. The telephone network has a bandwidth of approximately 3000 Hz, so the modems used on the telephone network must condition signals to fit within this band.

Communications terminology can be confusing. When the term 'communication mode' is applied to modems the following nomenclature is used:

SIMPLEX: Transmission in one direction only with no way of responding. A TV set is a form of simplex communication.

HALF DUPLEX: Transmission in two directions, but only one way at a time. CB operators either transmit or receive, but cannot do both simultaneously on a single channel. At the end of transmission it is necessary to advise the other party when through transmitting and ready to receive by saying "over". Then the other operator can begin transmitting.

FULL DUPLEX: Transmission in both directions simultaneously. A personal face-to-face conversation is a form of full duplex communication, where both persons can speak and listen at the same time. (Note: ASCII standard full duplex implies that the same data rate exists in both directions simultaneously, i.e., 1200/150 bps is not full duplex, 300/300 is full duplex. When operating at two different speeds, the slower speed is usually referred to as the secondary or reverse channel, or sometimes as the supervisory channel, while the higher speed is the primary channel.)

The half duplex mode control signals are generally required to turn the modem transmitter on or off, and the receiver off or on depending on the direction of transmission, whereas this is generally not required when operating in the full duplex mode.

Terminal manufacturers often use the terms half duplex and full duplex to mean whether local copy is provided, or whether the far end loops back (echoplexes) that which was transmitted. The presence or absence of local copy has nothing to do with the communications mode of the data link.

The strength of the signal that is injected into the phone lines is important, as a weak signal will not have enough power to overcome the noise and interference inherent in the system, while too strong a signal will overdrive the capabilities of the system and cause the signal to become distorted. The signal strength (measured at the phone line) should not exceed 0 dbm (2 volts peak to peak into a 600 ohm load), and transmission levels below -12 dbm (0.5 volts peak to peak) should be avoided. A transmission level between -6 dbm (1.0 volts p-p) and -9 dbm (0.75 volts p-p) is recommended for the best transmission level with the least amount of interference to the received signal.

At the receiving end, the signal may be significantly reduced in strength, and may be received at full strength (0 dbm, 2.0 volts p-p) or at a very reduced strength (-50 dbm, 0.01 volts p-p). Signals below -50 dbm are generally not recoverable, as the signal drops below the background noise and becomes very difficult to detect. Some modems do not recover signals below -40 dbm (0.02 volts p-p) as it becomes much more difficult to recover the signal below this level and less than 10% of all calls require this much sensitivity. (Note: A signal at -40 dbm would be barely audible.)

The modem operates by changing the digital signal that is presented to it to an audio signal that can be placed on the phone line. The type of modem determines the exact method by which the signal is converted, and the frequencies that are used.

There are many different types of modems, with each type designed to perform its particular function most efficiently. Because of the large number of different modems, only those modems which are of particular interest to the hobbyist will be discussed, those being the BELL 103 compatible, and the BELL 202 compatible modems. The 103 type modems are the most common and are the type used on most timeshare systems. These modems are designed to operate at transmission rates from 0 to 300 bps, with some of them capable of operating as high as 600 bps. A substantial increase in error rate should be expected at these higher speeds. The 103 type modem operates by changing a digital 0 to a frequency of 1070 Hz if in the originate mode, or 2025 if in the answer mode. It changes a digital 1 to a frequency of 1270 Hz if in the originate mode, and 2225 Hz if in the answer mode.

Since it is not possible to transmit two signals at once at the same frequencies and derive any intelligence from the received signal, the available

www.commodore.ca

BEYOND GAMES

SYSTEMS SOFTWARE FOR YOUR 6502 PERSONAL COMPUTER

by Kenneth Skier

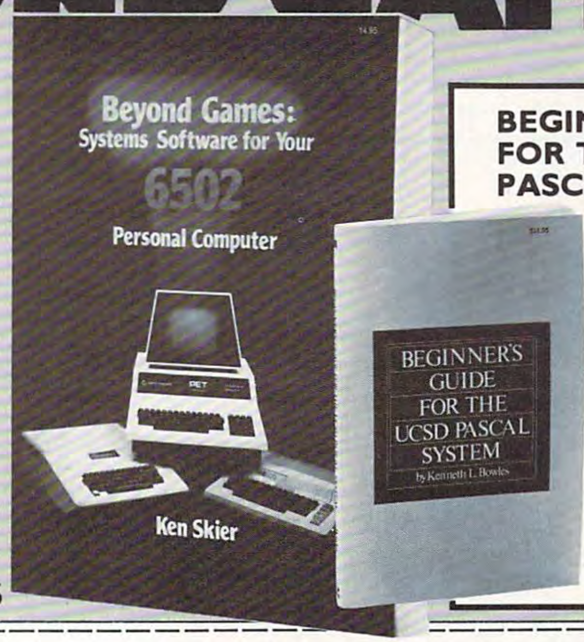
Creating Programs for the Apple, Atari, Challenger and PET Computers.

"This is a brand new book on assembly language that is fantastic! I have all the standard books on the 6502 and can't understand any of them. This book is not only beautifully written, very understandable, and takes no previous knowledge for granted . . . but it also includes many useful machine language routines such as a visible monitor, print utilities, disassembler, and a text editor. Well worth getting . . ."

M. Dunn
Editor, *ACE*
Atari Computer Enthusiasts Newsletter

ISBN 0-07-057860-5
440 pages softcover

$14.95



BEGINNER'S GUIDE FOR THE UCSD PASCAL SYSTEM

by Kenneth Bowles

Written by the originator of UCSD Pascal System, this informative book is an orientation guide to the UCSD Pascal System. For the novice, this book steps through the System, bringing the user to a sophisticated level of expertise. Once familiar with the System, the reader will find the guide an invaluable reference tool for creating advanced applications.

ISBN 0-07-006745-7
204 pages
softcover

$11.95

ORDER BY MAIL OR PHONE
CALL TOLL FREE: **800-258-5420**

Please send: ________ BEYOND GAMES **$14.95**

________ BEGINNER'S GUIDE TO PASCAL **$11.95**

Add .75 per book to cover postage and handling.

☐ Check or money order enclosed for $________.

☐ Bill VISA ☐ Bill MasterCard

Acct # ______________________ Expires ______

Name ______________________

Street ______________________

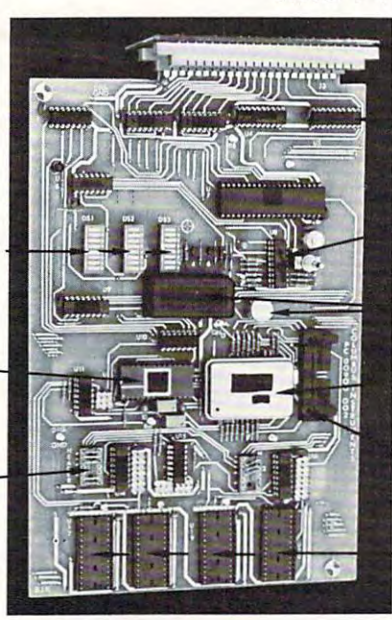
City ______________________

State ____________ Zip ____________

BYTE BOOKS
70 Main Street,
Peterborough, NH 03458

AIM-65/SYM-PET-KIM-6800

Universal Interface Board Converts AIM-65/SYM Into Professional Data Logger



(Also connects to PET or KIM with adapter cable. Adaptable to other 6502 and 6800 systems)

CONTAINS:

- ★ 12 bits, 16 channels, fast A/D converter
- ★ space for additional 16K RAM memory or 32K EPROM (or combination)
- ★ real time clock/calendar with real time interrupt capability and 10-year lithium battery backup
- ★ plugs directly into AIM-65 expansion connector with the help of a mini-mother board which supports up to three interface boards
- ★ supplied with supportive demonstration and control programs

AVAILABLE MODELS:

★	IB-902	Additional Memory Space (only)	$ 390.00
★	IB-902-A	Calendar/Clock plus memory space	$ 690.00
★	IB-902-B	A/D (12 bits, 16 channels plus memory space)	$ 960.00
★	IB-902-AB	A/D, plus memory space and calendar/clock	$1,270.00

Mini mother board to support up to three (3) interface boards $65.00

Quantity Discounts Available

COLUMBUS INSTRUMENTS INTERNATIONAL CORPORATION
Supplier of individual instruments and total measuring systems

950 N. HAGUE AVE., COLUMBUS, OHIO 43204 U.S.A.
PHONE: (614) 488-6176 TELEX: 246514

www.commodore.ca

DYNACOMP

Quality software for:*

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

CARD GAMES

BRIDGE 2.0 (Available for all computers) — **Price: $17.95 Cassette/$21.95 Diskette**
An all-inclusive version of this most popular of card games. This program both BIDS and PLAYS either contract or duplicate bridge. Depending on the contract, your computer opponents will either play the offense OR defense. If you bid too high, the computer will double your contract! BRIDGE 2.0 provides challenging entertainment for advanced players and is an excellent learning tool for the bridge novice. See the software review in 80 Software Critique.

HEARTS 1.5 (Available for all computers) — **Price: $15.95 Cassette/$19.95 Diskette**
An exciting and entertaining computer version of this popular card game. Hearts is a trick-oriented game in which the purpose is not to take any hearts or the queen of spades. Play against two computer opponents who are armed with hard-to-beat playing strategies. HEARTS 1.5 is an ideal game for introducing the uninitiated (your spouse) to computers. See the software review in 80 Software Critique.

STUD POKER (Atari only) — **Price: $11.95 Cassette/$15.95 Diskette**
This is the classic gambler's card game. The computer deals the cards one at a time and you (and the computer) bet on what you see. The computer does not cheat and *usually* bets the odds. However, it sometimes bluffs! Also included is a five card draw poker betting practice program. This package will run on a 16K ATARI. Color, graphics, sound.

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

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

THOUGHT PROVOKERS

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

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

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

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

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

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

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

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

NOMINOES JIGSAW (Atari, Apple and TRS-80 only) — **Price: $16.95 Cassette/$20.95 Diskette**
A jigsaw puzzle on your computer! Complete the puzzle by selecting your pieces from a table consisting of 60 different shapes. NOMINOES JIGSAW is a virtuoso programming effort. The graphics are superlative and the puzzle will challenge you with its three levels of difficulty. Scoring is based upon the number of guesses taken and by the difficulty of the board set-up.

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

CHOMP-OTHELLO (Atari only) — **Price: $11.95 Cassette/$15.95 Diskette**
CHOMP-OTHELLO? It's really two challenging games in one. CHOMP is similar in concept to NIM; you must bite off part of a cookie, but avoid taking the poisoned portion. OTHELLO is the popular board game set to fully utilize the Atari's graphics capability. It is also very hard to beat! This package will run on a 16K system.

DYNACOMP OFFERS THE FOLLOWING

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

AND MORE...

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

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

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

MOVING MAZE (Apple only) — **Price: $10.95 Cassette/$14.95 Diskette**
MOVING MAZE employs the games paddles to direct a puck from one side of a maze to the other. However, the maze is dynamically (and randomly) built and is continually being modified. The objective is to cross the maze without touching (or being hit by) a wall. Scoring is by an elapsed time indicator, and three levels of play are provided.

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

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

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

GIANT SLALOM (Atari only) — **Price: $14.95 Cassette/$18.95 Diskette**
This real-time action game is guaranteed addictive! Use the joystick to control your path through slalom courses consisting of both open and closed gates. Choose from different levels of difficulty, race against other players or simply take practice runs against the clock. GIANT SLALOM will run on 16K systems.

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

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

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

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

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

ADVENTURE

CRANSTON MANOR ADVENTURE (North Star and CP/M only) — **Price: $21.95 Diskette**
At last! A comprehensive Adventure game for North Star and CP/M systems. CRANSTON MANOR ADVENTURE takes you into mysterious CRANSTON MANOR where you attempt to gather fabulous treasures. Lurking in the manor are wild animals and robots who will not give up the treasures without a fight. The number of rooms is greater and the associated descriptions are much more elaborate than the current popular series of Adventure programs, making this game the top in its class. Play can be stopped at any time and the status stored on diskette.

ABOUT DYNACOMP

DYNACOMP is a leading distributor of small system software with sales spanning the world (currently in excess of 40 countries). During the past two years we have greatly enlarged the DYNACOMP product line, but have maintained and improved our high level of quality and customer support. The achievement in quality is apparent from our many repeat customers and the software reviews in such publications as COMPUTRONICS, 80 Software Critique and A.N.A.L.O.G. Our customer support is as close as your phone. It is always friendly. The staff is highly trained and always willing to discuss products or give advice.

**ATARI, PET, TRS-80, NORTHSTAR, CP/M and IBM are registered tradenames and/or trademarks.*

***TRS-80 diskettes are not supplied with DOS or BASIC.*

www.commodore.ca

BUSINESS and UTILITIES

SPELLGUARD™ (CP/M only) — **Price: $269.95 Disk**

SPELLGUARD is a revolutionary new product which increases the value of your current word processing system (WORDSTAR, MAGIC WAND, ELECTRIC PENCIL, TEXTED EDITOR II and others). Written entirely in assembly language, SPELLGUARD™ rapidly assists the user in eliminating spelling and typographical errors by comparing each word of the text against a dictionary (expandable) of over 20,000 of the most common English words. Words appearing in the text but not found in the dictionary are "flagged" for easy identification and correction. Most administrative staff familiar with word processing equipment will be able to use SPELLGUARD™ in only a few minutes.

MAIL LIST 2.2 (Apple, Atari and North Star diskette only) — **Price: $34.95**

This program is unmatched in its ability to store a maximum number of addresses on one diskette (minimum of 1100 per diskette, more than 2200 for "double density" systems!). Its many features include alphabetic and zip code sorting, label printing, merging of files and a unique keyword seeking routine which retrieves entries by a virtually limitless selection of user defined codes. Mail List 2.2 will even find and delete duplicate entries. A very valuable program!

FORM LETTER SYSTEM (FLS) (Atari, Apple and North Star diskette only) — **Price: $21.95**

Use FLS to create and edit form letters and address lists. Form letters are produced by automatically inserting each address into a predetermined portion of your letter. FLS is completely compatible with MAIL LIST 2.2, which may be used to manage your address files.

FLS and MAIL LIST 2.2 are available as a combined package for $49.95.

SORTIT (North Star only) — **Price: $29.95 Diskette**

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

PERSONAL FINANCE SYSTEM (Atari and North Star only) — **Price: $34.95 Diskette**

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

FAMILY BUDGET (Apple only) — **Price: $34.95 Diskette**

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

THE COMMUNICATOR (Atari only) — **Price: $49.95 Diskette**

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

DYNACOMP also supplies THE COMMUNICATOR with an Atari 830 modem for a combined price of $219.95. The modem is available separately for $189.95.

TEXT EDITOR II (CP/M) — **Price: $29.95 Diskette/$33.45 Disk**

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

DFILE (Atari and North Star only) — **Price: $19.95**

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

FINDIT (North Star only) — **Price: $19.95**

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

GRAFIX (TRS-80 only) — **Price: $14.95 Cassette/$18.95 Diskette**

This unique program allows you to easily create graphics directly from the keyboard. You "draw" your figure using the program's extensive cursor controls. Once the figure is made, it is automatically appended to your BASIC program as a string variable. Draw a "happy face", call it H$ and then print it from your program using PRINT H$! This is a very easy way to create and save graphics.

EDUCATION

HODGE PODGE (Apple only, 48K Applesoft or Integer BASIC) — **Price: $19.95 Cassette/$23.95 Diskette**

Let HODGE PODGE be your child's baby sitter. Pressing any key on your Apple will result in a different and intriguing "happening" related to the letter or number of the chosen key. The program's graphics, color and sound are a delight for children from ages 1½ to 9. HODGE PODGE is a non-intimidating teaching device which brings a new dimension to the use of computers in education.

TEACHER'S PET I (Available for all computers) — **Price: $11.95 Cassette/$15.95 Diskette**

This is the first of DYNACOMP's educational packages. Primarily intended for pre-school to grade 3, TEACHER'S PET provides the young student with counting practice, letter-word recognition and three levels of math skill exercises.

MORSE CODE TRAINER (TRS-80 only) — **Price: $12.95 Cassette/$16.95 Diskette**

MORSE CODE TRAINER is designed to develop and improve your speed and accuracy in deciphering Morse Code. As such, MCT is an ideal software package for FCC test practice. The code sound is obtained through the earphone jack of any standard cassette recorder. You may choose the pitch of the tones as well as the word rate. Also, various modes of operation are available including number, punctuation and alphabet tests, as well as the keying of your own message. A very effective way to learn code!

MISCELLANEOUS

CRYSTALS (Atari only) — **Price: $ 9.95 Cassette/$13.95 Diskette**

A unique algorithm randomly produces fascinating graphics displays accompanied with tones which vary as the patterns are built. No two patterns are the same, and the combined effect of the sound and graphics are mesmerizing. CRYSTALS has been used in local stores to demonstrate the sound and color features of the Atari.

NORTH STAR SOFTWARE EXCHANGE (NSSE) LIBRARY

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

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

AVAILABILITY

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

STATISTICS and ENGINEERING

DIGITAL FILTER (Available for all computers) — **Price: $29.95 Cassette/$33.95 Diskette**

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

DATA SMOOTHER (Not available for Atari) — **Price: $14.95 Cassette/$18.95 Diskette**

This special data smoothing program may be used to rapidly derive useful information from noisy business and engineering data which are equally spaced. The software features choice in degree and range of fit, as well as smoothed first and second derivative calculation. Also included is automatic plotting of the input data and smoothed results.

FOURIER ANALYZER (Available for all computers) — **Price: $16.95 Cassette/$20.95 Diskette**

Use this program to examine the frequency spectra of limited duration signals. The program features automatic scaling and plotting of the input data and results. Practical applications include the analysis of complicated patterns in such fields as electronics, communications and business.

TFA (Transfer Function Analyzer) — **Price: $19.95 Cassette/$23.95 Diskette**

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

HARMONIC ANALYZER (Available for all computers) — **Price: $24.95 Cassette/$28.95 Diskette**

HARMONIC ANALYZER was designed for the spectrum analysis of repetitive waveforms. Features include data file generation, editing and storage/retrieval as well as data and spectrum plotting. One particularly unique facility is that the input data need not be equally spaced or in order. The original data is sorted and a cubic spline interpolation is used to create the data file required by the FFT algorithm.

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

REGRESSION I (Available for all computers) — **Price: $19.95 Cassette/$23.95 Diskette**

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

REGRESSION II (PARAFIT) (Available for all computers) — **Price: $19.95 Cassette/$23.95 Diskette**

PARAFIT is designed to handle those cases in which the parameters are imbedded (possibly nonlinearly) in the fitting function. The user simply inserts the functional form, including the parameters (A(1), A(2), etc.) as one or more BASIC statement lines. Data and results may be manipulated and plotted as with REGRESSION I. Use REGRESSION I for polynomial fitting, and PARAFIT for those complicated functions.

MULTILINEAR REGRESSION (MLR) (Available for all computers) — **Price: $24.95 Cassette/$28.95 Diskette**

MLR is a professional software package for analyzing data sets containing two or more linearly independent variables. Besides performing the basic regression calculation, this program also provides easy to use data entry, storage, retrieval and editing functions. In addition, the user may interrogate the solution by supplying values for the independent variables. The number of variables and data size is limited only by the available memory.

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

ANOVA (Available for all computers) — **Price: $39.95 Cassette/$43.95 Diskette**

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

BASIC SCIENTIFIC SUBROUTINES, Volume 1 (Not available for Atari)

DYNACOMP is the exclusive distributor for the software keyed to the popular text *BASIC Scientific Subroutines, Volume I* by F. Ruckdeschel (see the BYTE/McGraw-Hill advertisement in BYTE magazine, January 1981). These subroutines have been assembled according to chapter. Included with each collection is a menu program which selects and demonstrates each subroutine.

Collection #1: Chapters 2 and 3: Data and function plotting, complex variables
Collection #2: Chapter 4: Matrix and vector operations
Collection #3: Chapters 5 and 6: Random number generators, series approximations
Price per collection: $14.95 Cassette/$18.95 Diskette

All three collections are available for $39.95 (three cassettes) and $49.95 (three diskettes).
Because the text is a vital part of the documentation, *BASIC Scientific Subroutines, Volume I* is available from DYNACOMP for $19.95 plus 75¢ postage and handling.

ROOTS (Available for all computers) — **Price $10.95 Cassette/$14.95 Diskette**

In a nutshell, ROOTS simultaneously determines all the zeroes of a polynomial having real coefficients. There is no limit on the degree of the polynomial, and because the procedure is iterative, the accuracy is generally very good. No initial guesses are required as input, and the calculated roots are substituted back into the polynomial and the residuals displayed.

ACTIVE CIRCUIT ANALYSIS (ACAP) (48K Apple only) — **Price: $25.95/$29.95 Diskette**

ACAP is the analog circuit designer's answer to LOGIC SIMULATOR. With ACAP you may analyze the response of an active or passive component circuit (e.g., a transistor amplifier, band pass filter, etc.). The circuit may be probed at equal steps in frequency, and the resulting complex (i.e., real and imaginary) voltages at each component juncture examined. By plotting the magnitude of these voltages, the frequency response of a filter or amplifier may be completely determined with respect to both amplitude and phase. In addition, ACAP prints a statistical analysis of the range of voltage responses which result from tolerance variations in the components.

ACAP is easy to learn and use. Simply describe the circuit in terms of the elements and their placement, and execute. Circuit descriptions may be saved onto cassette or diskette to be recalled at a later time for execution or editing. ACAP should be part of every circuit designer's program library.

LOGIC SIMULATOR (Apple only; 48K RAM) — **Price: $24.95 Cassette/$28.95 Diskette**

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

LOGIC DESIGNER (North Star and CP/M only) — **Price: $34.95 Diskette**

LOGIC DESIGNER is an exceptional Computer Aided Design (CAD) program. With it you may convert a large and complicated digitial truth table (the functional specification) into an optimized Boolean logic equation. This equation may then be *easily* converted into a circuit design using either NAND or AND/OR gates. Operationally, LOGIC DESIGNER is composed of a BASIC program which calls in a machine language routine to reduce execution time. Example: For a 7 variable by 127 line table, the processing time is only two minutes. LOGIC DESIGNER is clearly a fast and powerful tool for building digital circuitry.

ORDERING INFORMATION

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

Shipping and Handling Charges
Within North America: Add $1.50
Outside North America: Add 10% (Air Mail)

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

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

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

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

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

DYNACOMP, Inc.
1427 Monroe Avenue
Rochester, New York 14618
24 hour order phone: (716)586-7579 recording
Office phone (9AM-5PM EST): (716)442-8960

VISA — MasterCard

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

www.commodore.ca

signal bandwidth of the phone line must be divided into two bands (high band and low band) so that the signals present the minimum amount of interference to each other within the available bandwidth. The high band is referred to as the answer mode. This is because the station being called, the one that answers the call, is placed in this mode. The low band is referred to as the originate mode, as the station that originates the call uses this band.

	TX 1	TX 0	RX 1	RX 0
Originate:	**1270 Hz**	**1070 Hz**	**2225 Hz**	**2025 Hz**
Answer:	**2225 Hz**	**2025 Hz**	**1270 Hz**	**1070 Hz**

Because of the narrow frequency shift that is required for full duplex operation, it is very difficult to receive signals faster than 300 bps, and not practical to receive signals faster than 600 bps. The frequencies used were chosen to present the minimum amount of interference possible.

To receive a signal with the minimum amount of errors at 300 bps, the modem should be designed to operate at 400 bps. The frequency shift from a 1 to a 0 should be equal to or greater than ½ the maximum speed of transmission (200 Hz for 400 bps). The receive bandwidth should be equal to or greater than the maximum speed of transmission (400 Hz for 400 bps). A guard band should be maintained between the upper and lower bands equal to or greater than the maximum speed of transmission (guard band equals 555 Hz for the above indicated design). Using the above information, it would be possible to increase the receive bandwidth to 480 Hz and have a modem that operates up to 400 bps with a minimum error rate. To allow operation to 600 bps, the receive bandwidth is increased to allow the reception of the higher speed which causes a decrease in the guard band and then an increase in interference from the adjacent channel. This can be offset to some extent by providing more filtering at the receiver and transmitter to reduce the out of band signals as much as possible. This is, however, only a partial fix, and the signal will still be subject to a greater amount of distortion than the slower speed signals. The frequency shift could be increased to 300 Hz to match the 600 bps rate, however this causes the channel signals to be closer together, thus causing an increase in interchannel distortion. The channel spacing could be increased, but, due to phone line characteristics, a significant increase in delay distortion occurs outside the indicated bands. One way that the error rate can be reduced is to operate with local echo rather than echoplexing (half duplex operation as opposed to full duplex operation). This allows the guard band to extend down to the transmitter carrier frequency rather than the first sideband.

A problem that is encountered when using the phone lines for data communications over long distances are the echo supressors. When calling long distance, signal delays as long as 180 ms can be encountered within the continental United States and even longer delays can be encountered outside the US. These long delays can cause severe echoing which can be very disturbing to the caller. The phone company has provided a means of reducing this disturbance with a device called an echo supressor. An echo supressor inserts an amount of loss in the opposite direction of the loudest signal to reduce the echo to an acceptable level. This can affect proper modem operation. However, the phone company has recognized this problem and provided for a means to disable the echo suppressors. This is done by providing a signal of 2125 Hz ± 115 Hz for 100 ms if no signal has occurred. As can be seen, the disable signal falls within the answer modem's transmit frequency range, so that the echo supressors are automatically disabled when the answer modem begins transmitting. Even with this improvement, a significant amount of errors can be encountered. If data is to be transmitted over long distances with minimum errors it is recommended that half duplex operation be used (local copy rather than echoplexing). By having only one frequency shift occurring at any one time a minimum amount of interference will be generated.

With a private fixed line with C2 conditioning it is possible to achieve 1800 bps.

Another type of modem that is sometimes used is the BELL 202 type modem. This is a 0 to 1200 bps half duplex only type modem. Operation at 1200 bps is provided by using the full usable phone line bandwidth for transmission instead of dividing it into two bands. The frequency shift between a 1 and a 0 is expanded to 1000 Hz. With the wide frequency difference between the two states it becomes much easier to recognize when a change has occured which allows the change to be made more quickly. Although the bandwidth and frequency shift range allows for operation to 2000 bps, due to phone line envelope delay distortion and attenuation, the standard dial-up line is limited to 1200 bps. With a private fixed line with C2 conditioning it is possible to achieve 1800 bps.

Since the 202 type modem is a half duplex modem, a greater amount of control over the modem is required than over the 103 type modem. Since transmission can only occur in one direction at any one time, a means to indicate to the other end of the link that you are through transmitting and it should begin transmitting, must be provided. This is generally performed by sending a final character (ASCII EOT) indicating this after which the transmitter must be turned off and the receiver

www.commodore.ca

COMPETENCY EXAM PREPARATION SERIES

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

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

National Proficiency Series $1,299.00
N.Y.S. Regents Competency Test, Preparation Series $1,299.00

If desired separate Mathematics and Verbal packages are available for $799.00 ea. A Spanish language version of the Mathematics Instruction Package is available at no extra charge.

COLLEGE BOARD PREPARATION SERIES 81/82

for TRS-80 NORTHSTAR™ PET, APPLE OSI

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

S.A.T., P.S.A.T., N.M.S.Q.T., set includes 25 programs covering Vocabulary, Word Relationships, Reading Comprehension, Sentence Completion, and Mathematics. Price $149.95

EDUCATOR EDITION - includes all of the above programs plus detailed solutions and explanations. Price $229.95

Independent Tests of S.A.T. series performance show a mean total increase of 70 points in students' scores.

Update Pack to 81/82 specs. Available to previous owners. Price $69.95

ODYSSEY IN TIME

This spectacular adventure game adds a new dimension of excitement and complexity to **Time Traveler.** Players must now compete with the powerful and treacherous adversary in their exacting quest for victory.

To succeed they must vanquish this adversary in combat that rages across 24 time periods.

Odyssey In Time includes all the challenges of Time Traveler plus 10 additional eras, including those of Alexander the Great, Emperor Asoka of India, Attila the Hun, Genghis Khan. Each game is unique, and may be **interupted and saved** for later play.

available for APPLE & TR-80 PET, 32K - $39.95

ISAAC NEWTON

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

In a classroom setting the instructor may elect to choose "Laws of Nature" in accordance with the complete instruction manual provided.

For insight into some of the basic principles underlying **ISAAC NEWTON** see GODEL, ESCHER, BACH by Douglas R. Hofstadler, Chapter XIX and Martin Gardner's MATHEMATICAL GAMES column in **Scientific American,** October, 1977 and June, 1959. $24.95.

TIME TRAVELER

Confronts players with complex decision situations and the demand for real time action. Using the **Time Machine**, players must face a challenging series of environments that include; The Athens of Pericles, Imperial Rome, Nebuchadnezzar's Babylon, Ikhnaton's Egypt, Jerusalem at the time of the crucifixion, The Crusades, Machiavelli's Italy, The French Revolution, The American Revolution, and The English Civil War. Deal with Hitler's Third Reich, Vikings, etc. At the start of each game players may choose a level of difficulty . . . the more difficult, the greater the time pressure. To succeed you must build alliances and struggle with the ruling powers. Each game is unique.

$24.95

Send $2.00 for complete Catalogue.

$5.00 Discount Coupon included in Catalogue.

PROGRAMS AVAILABLE FOR
TRS-80, APPLE II & PET
(unless otherwise indicated)
☐ disk or ☐ cassette (please specify)

VISA

Krell Software Corp.

Send check or money order to
21 Milbrook Drive, Stony Brook, NY 11790
(516) 751-5139

All programs require 16K/TRS-80 programs require LEVEL II BASIC/APPLE programs require APPLESOFT BASIC

NY State Residents Add Sales Tax

www.commodore.ca

enabled. (This is most often referred to as turnaround.) Another problem that must be considered when turnaround is initiated: a delay of 200 ms must be allowed for the echo suppressors to stabalize. This delay can significantly eat into transmission time when large amounts of data are to be transferred, particularly when short transmission blocks are being used.

One way to reduce this delay is to maintain a reverse channel or supervisory link. The 202 modem has an optional reverse channel arrangement which consists of a 387 Hz amplitude modulated signal with a maximum transmission rate of 5 bps. It maintains the echo suppressors in the off condition and thereby reduces turnaround time to less than 100 ms, the time now being limited by the time it takes for the transmitted signal to die out and the signal from the other end to be acquired by the modem. The reverse channel also provides the ability to use supervisory signals, which allows for the early termination of a block of data. This can be useful especially when the transmission, turnaround can be requested by the receiving station by dropping the reverse channel. This way the transmission can be terminated, thereby eliminating the time lost by having to transmit the rest of the block before turnaround could be done. The reverse channel should not be used when the primary channel is in use as errors in the primary channel's data can be generated. In addition, errors in the received primary channel should be expected when the reverse channel is modulated. (Turned off or on).

So far the modems that have been discussed were of the asynchronous type. Asynchronous means that the digital information may be presented to the modem in any form or at any speed as long as the maximum bit rate (minimum duration of a stable state) is not exceeded. This is very useful, as it allows the modem to be transparent to the data being transmitted. Because at least one cycle is needed to determine a frequency change, the absolute maximum transmission speed of the available bandwidth cannot be achieved.

In applications where throughput is of the greatest importance, synchronous modems are generally used. In the synchronous modem, transparency is sacrificed for the greater speed capability. The synchronous modem synchronizes itself to the remote modem, and requires that the data sent to it be in synchronization with its transmissions. This is done with a signal that is provided either by the modem, or by the connecting device called a clock. The clock insures that all transmissions occur in sync by providing a master reference for those actions. Although asynchronous protocols are not efficient enough for maximum throughput, special protocols have been developed to obtain the maximum throughput possible, the more common of these being SDLC and HDLC.

It should be noted that SDLC and HDLC can be used with synchronous type modems also. The type of modem only refers to the hardware configuration required for the modem and not the transmission protocols.

Some of the synchronous modems in use are the BELL 201 (2400 bps), the BELL 208 (4800 bps), and the BELL 209 (9600 bps). There are many other type modems. 9600 bps is presently the maximum transmission rate being used on the standard BELL phone lines.

All digital modems, whether synchronous or asynchronous, high speed or low speed, perform the same job, they convert digital information that is presented to them to a form that can be transmitted on the phone lines, and convert the received information back to its digital form. ©

NEED MORE MEMORY?

32K BYTE DYNAMIC RAM & ROM EXPANSION BOARD

Expand Your 4K/8K PET
SYM/KIM/AIM –65 to 32K

- Easily connected to your computer via the expansion connector.
- Build huge and complex programs!
- Need 64K of RAM? Buy two boards, on board configuration circuitry will allow you to expand to 64K easily!
- New dynamic RAM technology brings you more memory in less space and at a lower cost!
- RAM chips are upgraded, compatible with the new 64K RAM chips for future expansion!
- Operates on +5 volts only, supplied from your computer power supply, no on board generators to go bad.
- Requires A LOT less power than static RAM!
- Has full invisible refresh operation, does not interfere with processor operation.
- Fully buffered DATA BUSS.
- 5 on board sockets for 2716/2732 (2K/4K) type EPROMS, addressable anywhere.
- Great for designing a two board computer system. (CPU, I/O-RAM, ROM).
- Other specifications: Disable any 4K block of RAM for I/O, place RAM above or below 8000 HEX. KIM-4 BUSS COMPATIBLE FOR CARD RACKS. Adapter cables available for non rack use.
- All these features on a 6 x 4.5" board!

ASSEMBLED & TESTED BOARDS—GUARANTEED FOR 6 MONTHS PURCHASE PRICE IS FULLY REFUNDABLE IF RETURNED UNDAMAGED WITHIN 14 DAYS

List Price — **$289.88**
Introductory Price — **$269.88**
Include $2.00 for S&H — Allow 4 weeks for delivery

Full informative documentation included with all our products.
C.O.D. Orders Accepted (702) 361-6331 Mail Order Only.

PROTRONICS
COMPUTER INNOVATIONS

1516 E. Tropicana, Suite 7A
Las Vegas, Nevada 89109

TOLL FREE
Subscription
Order Line
800-227-1617
In CA 800-772-3545
Please ask for Extension 401.

www.commodore.ca

4 new products from Matrix

stat

STATISTICS APPLICATIONS FOR TECHNICIANS

Here is a package that is so state-of-the-art that many of the statistical techniques implemented here are not even in the textbooks yet. STAT is a set of programs for performing a large portion of the most frequently used statistical inference methods. Data can be entered and stored on four different types of data files. These data files can be modified also. The statistical procedures available in the package include the following parametric inference procedures:

SUMMARY STATISTICS for each data file and date set, including the mean and standard deviation.

CONFIDENCE INTERVALS for the following: (1) the mean of a normal population (both with and without the variance known), (2) the variance of a normal distribution (both with and without the mean known), (3) the parameter (mean time to failure) of an exponential distribution, (4) the parameter (proportion) of a binomial distribution, (5) the difference of two normal means (for various combinations of assumptions about the variances of the populations) and (6) for the ratio of two normal variances.

TESTS OF HYPOTHESES about (1) a normal mean, with various cases corresponding to possible assumptions about the variance, (2) the difference in two normal means (various cases) and (3) the ratio of two normal variances.

TESTS OF THE EXPONENTIAL MEAN (mean time to failure) and **RATIO OF MEANS.**

TESTS OF THE BINOMIAL PARAMETER (proportion) and **DIFFERENCE OF PARAMETERS**

MULTIPLE REGRESSION, including estimation of coefficients, estimation of the error variance, and test of significance of the regression.

ANALYSIS OF VARIANCE for one-way and balanced two-way designs, including interaction.

The software is user-friendly, allowing easy recovery from errors and selection of alternate analyses, as desired. The user's interaction is entirely menu driven, with error recovery features. An extensive user's manual introduces the statistical inference procedures used, and gives worked examples for each situation considered, illustrating typical applications. These worked examples serve as a pattern and allow the reader to check his use of the programs. The user's manual gives complete documentation of the programs and procedures used in them. All formulae, algorithms and procedures are listed and referenced to commonly available statistical literature.

A notable feature of the package is inclusion of very efficient routines for the computation of probabilities and quantiles for the most common statistical distributions, including normal, binomial, chi-square, t and F. Thus the user is not required to furnish "tabular values" from outside sources when performing statistical analyses with this package. STAT complete with all documentation is $200.

APPLE II APPLESOFT and at least one drive
APPLE II PASCAL SYSTEM
COMMODORE 32K with 4040/8050 drive
Radio Shack Mod III and CP/M compatibility by fall.

calc

MACHINE SPEED "BASIC"

CALC was designed to provide programmers of microcomputers with a portable language that combines the programming ease of the higher languages with the speed and flexibility of assembler programming. CALC is totally portable on the Commodore and APPLE II computers. This means that CALC source code written on an APPLE II will run **as is** on a Commodore machine and vice versa.

When possible, CALC makes direct use of the BASIC ROM machine language routines in the Commodore and APPLE II. In essence, CALC provides access to the power in the BASIC ROMs without the overhead of the BASIC interpreter. This includes floating point arithmetic and all library functions. In addition, we have added features that BASIC does not have. These include true integer arithmetic and machine speed string handling with search and replacement features.

CALC can fetch and replace BASIC variables and arrays by name. The programmer indicates what is to be done using simple keyword commands (ADD, MULT, SINE, etc.) and leaves all register set-up, bitformat and the like to CALC. The object code resulting from CALC programs is very compact and consists of direct calls to the BASIC ROMs or to the CALC runtime package.

CALC comes in 4K of PROM containing a relocatable runtime package and a very complete Trace Window feature for debugging CALC programs. CALC produces romable 6502 code that does **not** require the CALC development PROM to function. Programs written in CALC will run on any stock PET or APPLE. CALC comes with a 60-page manual.

CALC PROM on Commodore is $115.: indicate 3.0 or 4.0 BASIC, 40/80 column screen and rom sockets $9000, $A000 or $B000.

CALC on APPLE II via quality slot independent board is $160.

CALC manual by itself is $10.

CALC requires Moser/Mae Macro Assembler (Tape or Disk version)

sort

MULTI-KEY MACHINE LANGUAGE

A 6502 machine language in-memory sorting algorithm of commercial quality is available as part of a new utility eprom for PET and APPLE owners. Most sorts are accomplished in less than a second and very large sorts take only a few seconds. The algorithm is a diminishing increment insertion sort, with optionally chosen increments. This algorithm has the advantage of being significantly faster (but not much longer) than simpler ones, and significantly smaller (but not much slower) than more complicated ones. Moreover, unlike some of the more complicated algorithms, there are no conditions under which the performance of this sort degenerates or fails.

SORT is intelligent to the degree that almost no user set-up operations are required. SORT handles integer, floating-point and string arrays, as well as multiple dimensioned arrays with equal ease. In addition, multi-key sorting of string arrays has been enabled. The user may specify the character within a string to begin sorting on and how many characters are to be evaluated. SORT is capable of performing up to twenty of these multi-key sub-sorts (on matches found) at the same time. This multi-level 20-KEY capacity for string arrays greatly increases the uses to which SORT can be put.

SORT comes as part of a utility EPROM that also includes a hi-speed machine language text screen dump. Complete instructions for installation and use are included.

SORT is available for large-keyboard PETS Only. One ROM will work for BASIC 3.0 & 4.0, 40 or 80 column screens. When ordering you need only to indicate which ROM socket address in PET you prefer EPROM ($9000, $A000 or $B000). PET SORT EPROM at hex $9000 location if you do not specify. PET EPROM price is $55.00 (postpaid).

SORT is available on the APPLE II via a top quality, fully socketed, EPROM board that is slot independent. The MATRIX APPLE board includes a function driver that supports up to 16 EPROM based functions in case you would like to use your own EPROM in place of ours. EPROM board with SORT, text screen dump and function driver are all slot independent and may be used in any slot except 0. Price APPLE CARD $110.00 (postpaid).

bookkeeper

TOTAL BUSINESS SYSTEM

BOOKKEEPER was designed by a team of accountants and businessmen, and then programmed especially for microcomputers. This is not hand-me-down software from mainframe computers. BOOKKEEPER is a totally integrated management and accounting system that is available now on the more popular micro systems.

This series of interlocking programs is menu-driven and self-prompting with relative file structure implemented throughout. In some versions, machine language routines have been used to provide more efficient operation. The system employs state-of-the-art techniques and has been designed to be user-friendly. No knowledge of accounting or computers is required.

We believe the system can be operated using little more than the screen prompts. But for completeness, our MATRIX User Guide (two-inch ring binder) contains almost 200 pages of details on the BOOKKEEPER system plus a helpful introduction to business accounting principles. We suggest that you send for a more complete description of BOOKKEEPER or invest in a copy of the User Guide. There is room here only for a general description.

BOOKKEEPER is available for both SERVICE and RETAIL/WHOLESALE firms. This total business system contains the following: 375 General Ledger accounts (ten departments with accompanying revenue and expense accounts), Accounts Receivable file with maintenance and report capabilities (1000 accounts), Payroll with all federal withholding computed, state and local income tax capabilities for all fifty states (100 employees); Cash Receipts and Cash Disbursements programs that keep track of inventory sales by department, Sales Tax computations, Receipts, and Invoices; Accounts Payable file with maintenance and report capabilities (100 accounts). The system also generates and prints valuable management reports such as Departmental Budgeting, Profit and Loss Statements by Department, the traditional Chart of Accounts Summation (Trial Balance), and Financial Reports.

The Retail/Wholesale version of BOOKKEEPER includes a perpetual inventory control system and permits point-of-sale invoices.

BOOKKEEPER is available now on the COMMODORE 8032/8050, 48K APPLE II+ and RADIO SHACK Model III computers. CP/M compatible version available by September.

The BOOKKEEPER system retails at $1000.00.

Bookkeeper manual by itself is $20.00.

Matrix software

315 Marion Avenue, Big Rapids, MI 49307
(616) 796-2483 or 796-0381

VISA MasterCard

Dealer Inquiries Invited.

www.commodore.ca

The Column Calculator

James L. Simonson
Gunnison, CO

Editor's Note: This program suggests many additional applications. If you add interesting expansions to it, send them in to **COMPUTE!** — *RM*

When I first got the idea to write a program for a column calculator, I imagined a very short time would be spent on the project. Sure, I had seen similar programs published in magazines and I knew there were some very sophisticated programs on the market, but I had some special problems. Principal among them was only 8K of RAM. Also, there were no programs in print that would run on my Atari 800. With this constraint, I knew I had to design a bare bones framework for my column calculator. This program is the column calculator framework that I came up with. First, I will describe the operation of the basic calculator — then we can explore the fun part.

This program provides a 12 row by 12 column matrix for data entry. Two additional rows and three additional columns are used for totals and to store other summary calculations. The program is written to provide row and column totals in column 13 and row 13. Of course, the grand total is in box 13, 13. The basic menu choices are:

1. View data columns (scan left or right in array or go directly to summary columns).
2. Enter data (choose column and number of rows).
3. Do calculations.
4. Zero the matrix.

Figure 1 illustrates the arrangement of the column calculator. Screen limitations permit only three columns on the screen at one time, thus the option to scan left and right in the array.

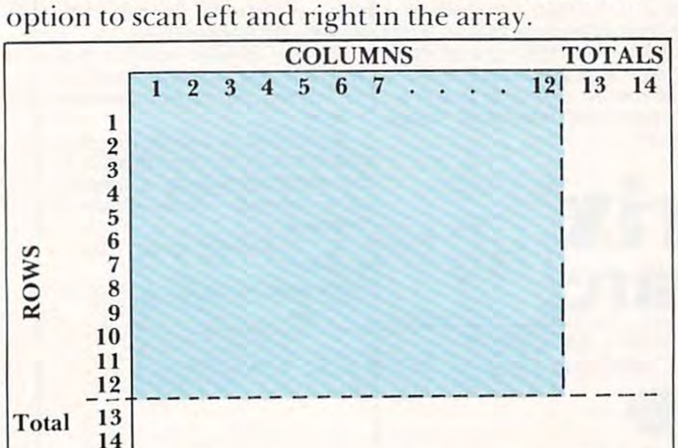


Figure 1. Total arrangement of the column calculator.

Program 1. Microsoft version.

```
20 DIM J(14,15)
30 GOTO 510
35 REM VIEW DATA COLUMNS
40 X=1
50 PRINTCHR$(147);:PRINT
60 PRINT,:FOR K=X TO X+2:PRINTK,:NEXT K:PRINT
70 RESTORE:FOR I=1 TO 14:READ A$
80 PRINTI;A$,J(I,X),J(I,X+1),J(I,X+2)
90 IF I=12 THEN GOSUB 550
100 NEXT I
110 PRINT:PRINT"DO YOU WANT TO SEE COLUMNS"
115 PRINT"LEFT  OR RIGHT? (L/R),"
117 INPUT "'S' FOR SUMMARY.";D$
120 IF D$<>"R" THEN 150
130 IF X<=10 THEN X=X+3:GOTO 50
140 GOTO 110
150 IF D$="L" AND X>1 THEN X=X-3:GOTO 50
160 IF D$="M" THEN GOSUB 510
170 IF D$="S" THEN X=13:GOTO 50
180 PRINT"INVALID DIRECTION":GOTO 110
200 REM ENTER DATA
210 PRINTCHR$(147);:PRINT
212 PRINT "WHAT COLUMN NUMBER DO YOU WISH TO"
215 PRINT"ENTER DATA IN (1-12)."::INPUT C
220 IFC=13THENPRINT"RESERVED FOR TOTALS":GOTO210
230 PRINT:PRINT "HOW MANY ROWS DO YOU WANT"
235 RESTORE
237 PRINT "TO WORK WITH (1-12)";:INPUT K
240 PRINT:PRINT"ROW","CURRENT","COLUMN ";C
250 FOR I=1 TO K:IF I=13 THEN GOSUB 550
255 READ A$
260 PRINT I;A$,J(I,C),:INPUT J:J(I,C)=J
270 NEXT I
280 PRINT "ENTER 'C' FOR ANOTHER COLUMN"
285 PRINT "'M' FOR THE MENU.";:INPUT D$
290 IF D$="C" THEN 210
295 GOTO 510
300 PRINTCHR$(147);"      ***CALCULATING***"
310 FOR I=1 TO 14
315 J(I,13)=0:J(13,I)=0
320 NEXT I:J(13,15)=0
380 FOR X=1 TO 12
385 FOR Y=1 TO 12
387 IF D$="4" THEN J(Y,X)=0
400 J(Y,13)=J(Y,13)+J(Y,X)
410 J(13,X)=J(13,X)+J(Y,X)
420 NEXT Y:NEXT X
430 FOR X=1 TO 12
435 J(13,13)=J(13,13)+J(X,13):NEXT X
510 PRINTCHR$(147);TAB(11);"COLUMN CALCULATOR"
511 PRINT "     1) VIEW COLUMNS
512 PRINT "     2) ENTER DATA
513 PRINT "     3) DO CALCULATIONS
514 PRINT "     4) ZERO THE MATRIX":PRINT
515 INPUT "WHAT IS YOUR CHOICE (1-4)";D$
520 IF D$="1" THEN 40
530 IF D$="3" OR D$="4" THEN 300
```

www.commodore.ca

WORD-CHECK

WORDCHECK is a poor spellers dream come true. Designed to interact with WORDPRO, it has 2100 root words and suffixes. In addition for the business and scientific user it has the capacity for 900 industrial or scientific terms which you load in yourself. You have a total vocabulary of approximately 7500 words at your fingertips. It simply goes through the text and flags any words that it doesn't recognize.

WORDCHECK is the ideal program to proof your spelling, whether it is one paragraph or a 100 page manual. The dictionary is versatile, allowing the user to add or delete words. You can design the program with the technical terms your profession uses, even duplicating the table and tailoring it for each person in your office. Let WORDCHECK do the work for you quickly and accurately.

CREATE-A-BASE

CREATE-A-BASE is a data base file management system that enables the user to choose the number of fields needed in a file, and add or delete fields without disturbing any of the existing data. Once a file is created you can perform any of 30 functions. Such as:

- Interact with WORDPRO 4, and 4+
- Do mathematic functions on any 2 or more fields
- Sort 650 files in only 19 seconds
- Merge any sequential file into a **CREATE-A-BASE** file, and output a sequential file from a **CREATE-A-BASE** file.
- The report generator has the feature of user defined fields and field width.
- Printouts can be generated by values such as, greater than, less than, equal to or in alpha or numeric codes.

You don't have to be a programmer to operate **CREATE-A-BASE** on your COMMODORE computer. Its menu driven and asks you questions at each step as you perform any of its many functions.

AVAILABLE

at your local COMMODORE dealer or
distributed exclusively in CANADA by
B.P.I. Micro Systems, Ltd.
80 Barbados Blvd. #14
Scarborough, Ontario M1J1K9
Special Dealer Introductory Package Available

Micro Computer Industries Ltd.

1520 E. Mulberry, Suite 170 Fort Collins, CO 80524

1-303-221-1955

www.commodore.ca

```
535 IF D$="2" THEN 200
540 GOTO 515
550 FOR D=1 TO 38:PRINT "-";:NEXT D:PRINT:RETURN
800 DATA -,-,-,-,-,-,-,-,-,-,-,-,X,X
```

Line 1 is part of the error TRAPping routine and line 20 DIMensions the variables. Lines 40 – 180 contain the "view data" routine. In this routine, I found a handy way to label my rows by using a DATA statement. Line 25 reads row labels from DATA statement 800. This way, the user only has to change the DATA statement for his or her own labels. Be certain there are always 14 entries in the DATA statement. Also, there is only room for a five-character label.

The data entry routine is in lines 210 – 295. I soon learned that, when I was revising data, I wanted to see the number which was presently stored. I accomplished this in step 260 by simply having J(I,C) printed. The next program segment, 300 – 495, is where I put the calculating routine. This is what expanded my involvement with this program. I soon realized that I could leave gaps for entering special routines, depending on the functions I wanted the calculator to perform. Lines 310 – 320 zero the column and row totals. A gap is reserved between 320 and 380. Lines 380 – 420 compute totals, but also reserve a gap between 385 and 400. Line 430 computes the grand total. Finally, lines 510 – 540 contain the menu routine, while subroutine 550 prints the dashed line. If an entry error occurs, lines 1000 – 1020 return the user to the error point for another chance.

Written for my Atari, the program, in its passive form, uses 1904 bytes of RAM. Memory use rises to 3369 when working. Only minor changes are necessary for other BASICs.

Now the fun began. I had my skeletal program to total across and down in the array, but I knew I could do much more by just entering and deleting a few lines in my reserved gaps. The results of my personal brainstorming for options are discussed below. The calculator will still total down and across, but, by entering the lines listed under any option, the calculator will also do the work described. Delete the optional lines, enter another option's lines, and you have a different tool.

Options:

1/ Compute averages across columns:

Column 14 is used to store the number of entries in a row, which is then used as a divisor of the entries in column 13. The average for each row is then stored in column 15 and shows up when you view data.

```
440 FOR Y=1 TO 12: IF J(Y,13)=0 THEN NEXT Y
445 J(Y,15)=INT (J(Y,13)/J(Y,14)): NEXT Y
```

2. Compute percent of total:

The totals in column, or row 13, can be divided by the grand total in (13,13) and the results stored in column or row 14.

Percent across columns in column 14:

```
440 FOR Y=1 TO 12: IF J(Y,13)=0 THEN 450
445 J(Y,14)=INT (J(Y,13)*100/J(13,13))
450 REMARK**RESERVED FOR PERCENT ACROSS
    ROWS STATEMENT
455 NEXT Y
```

To compute percent across rows and store the result in row 14:

```
450 IF J(13,Y)=0 THEN 455: J(14,Y)=INT (J(13,Y)
    *100/J(13,13))
```

3. Compute a weighting or cross product between data rows:

Data in one row is weighted, or multiplied, by data in another row. The result is entered within the data matrix and summed in column 13.

Note: For all the uses I could think of, the only valid totals in this option are in rows 3, 6, 9, and 12.

```
322 FOR Y=1 TO 10 STEP 3
323 FOR X=1 TO 12
324 J(Y+2,X)=J(Y,X)*J(Y+1,X)
325 NEXT X: NEXT Y
```

This calculator option can be used as a simple decision tool. You can mentally assign each set of three rows to one option being considered: For example, Ford (rows 1-3); Chevrolet (rows 4-6); etc. Then you can assign different criteria to each column — four different alternatives can be evaluated. Examples might include: Column 1 represents mileage; Column 2 represents cost; etc. In rows 1, 4, 7, and 10, enter the degree to which each alternative satisfies each criteria (1-100). In rows 2, 5, 8, and 11 enter the relative importance of each criteria (1-19). (The previous entry should be the same for each alternative). The alternative best satisfying the criteria evaluated will have the highest total in column 13 of rows 3, 6, 9, and 12. I suggest the data statement be changed to read:

```
800 DATA -RTG, -WT, -ATOT, -RTG, -WT, -BTOT,
    -RTG, -WT, -CTOT, -RTG, -WT, -DTAT, *, *
```

4. Convert hours and salaries to total costs:

You can record up to 12 people on four different jobs. Use the same steps as in "3" above with the following data statement:

```
800 DATA, -HRS, -$/H, -ATOT, -HRS, -$/H, -BTOT,
    -HRS, -$/H, -CTOT, -HRS, -$/H, DTOT, *,*
```

5. Convert quantity and unit costs to total cost:

Same steps as in "3" above with the following data statement:

```
800 DATA -QTY, -$EA, -TOT$, -QTY, -$EA, -TOT$,
    -QTY, $EA, -TOT$, -QTY, -$EA, -TOT$, *,*
```

6. You can analyze the wisdom of a contemplated investment by calculating the present net worth with a discounting interest rate of your choice. Using the column numbers to represent years into the future, enter the

www.commodore.ca

80 COLUMN GRAPHICS

The Integrated Visible Memory for the PET has now been redesigned for the new 12" screen 80 column and forthcoming 40 column PET computers from Commodore. Like earlier MTU units, the new K-1008-43 package mounts inside the PET case for total protection. To make the power and flexibility of the 320 by 200 bit mapped pixel graphics display easily accessible, we have designed the Keyword Graphic Program. This adds 45 graphics commands to Commodore BASIC. If you have been waiting for easy to use, high resolution graphics for your PET, isn't it time you called MTU?

K-1008-43M Manual only $10 (credited toward purchase)
k-1008-43 Complete ready to install package $495

Mastercharge and Visa accepted

Write or call today for our full line catalog describing all MTU 6502 products, including our high speed 8" Floppy Disk Controller for up to 4 megabytes of PET storage.

The image on the screen was created by the program below.

```
 10 VISMEM: CLEAR
 20 P=160: Q=100
 30 XP=144: XR=1.5*3.1415927
 40 YP=56: YR=1: ZP=64
 50 XF=XR/XP: YF=YP/YR: ZF=XR/ZP
 60 FOR ZI=-Q TO Q-1
 70 IF ZI<-ZP OR ZI>ZP GOTO 150
 80 ZT=ZI*XP/ZP: ZZ=ZI
 90 XL=INT(.5+SQR(XP*XP-ZT*ZT))
100 FOR XI=-XL TO XL
110 XT=SQR(XI*XI+ZT*ZT)*XF: XX=XI
120 YY=(SIN(XT)+.4*SIN(3*XT))*YF
130 GOSUB 170
140 NEXT XI
150 NEXT ZI
160 STOP
170 X1=XX+ZZ+P
180 Y1=YY-ZZ+Q
190 GMODE 1: MOVE X1,Y1: WRPIX
200 IF Y1=0 GOTO 220
210 GMODE 2: LINE X1,Y1-1,X1,0
220 RETURN
```

MTU
Micro Technology Unlimited
2806 Hillsborough Street
P.O. Box 12106
Raleigh, NC 27605, U.S.A.
(919) 833-1458

NOW 80 COLUMN PETS CAN HAVE MTU HIGH RESOLUTION GRAPHICS

www.commodore.ca

estimated annual costs and returns for up to 12 years in alternate rows (1, 3, 5 ...). The steps below will then discount each value and store the result in the row below (2, 4, 6 ...). In this case, the totals in column 13 will represent the total present net worth of each cost or return. If the returns are entered as positive values and costs as negative values, the grand total in (13, 13) will indicate the present net value of the contemplated investment. Investments with a negative total should normally be avoided.

```
325 ?"ENTER 1+(INTEREST RATE IN DECIMAL)":
    INPUT R
330 FOR X=1 TO 12: FOR Y=1 TO 11 STEP 2
340 NEXT Y: NEXT X
385 FOR Y=2 TO 12 STEP 2
800 DATA -COST, -NPW, -RTNS, -NPW, -COST,
    -NPW, -RTNS, -NPW, *,*,*,*,NA,NA
```

To change from this option to another, delete lines 325 thru 340 and enter:

```
385 FOR Y=1 TO 12
```

As a novice programmer, writing this program taught me the power of two dimensional arrays. Now you, too, have a framework for a column calculator. I have given you a few options I thought might be helpful. The fun begins when you start brainstorming your own options.

Program 2. Atari version.

```
1 TRAP 1000
5 REM *****"THE COLUMN CALCULATOR"****
6 REM -----------VERSION 1.2
7 REM -------BY JAMES L. SIMONSON-----
8 REM ***********JUNE,1981************
20 DIM J(14,15),D$(1),A$(5)
30 GOTO 510
35 REM VIEW DATA COLUMNS
40 X=1
50 GRAPHICS 0:? :? "          COLUMN CAL
CULATOR":?
60 ? "",:FOR K=X TO X+2:? K,:NEXT K:?
70 RESTORE :FOR I=1 TO 14:READ A$
80 ? I;A$,J(I,X),J(I,X+1),J(I,X+2)
90 IF I=12 THEN GOSUB 550
100 NEXT I
110 ? :? "DO YOU WANT TO SEE COLUMNS LEF
T OR":? "RIGHT? (L/R), 'M' FOR MENU, 'S'
 FOR":? "SUMMARY.";:INPUT D$
120 IF D$<>"R" THEN 150
130 IF X<=10 THEN X=X+3:GOTO 50
140 GOTO 110
150 IF D$="L" AND X>1 THEN X=X-3:GOTO 50

160 IF D$="M" THEN GOSUB 510
170 IF D$="S" THEN X=13:GOTO 50
180 ? "INVALID DIRECTION":GOTO 110
200 REM ENTER DATA
210 GRAPHICS 0:? :? "WHAT COLUMN NUMBER
DO YOU WISH TO ":? "ENTER DATA IN (1-12)
.";:INPUT C
220 IF C=13 THEN ? "RESERVED FOR TOTALS"
:GOTO 210
230 ? :? "HOW MANY ROWS DO YOU WANT TO "
:? "WORK WITH (1-12)";:INPUT K
235 RESTORE
240 ? :? "ROW","CURRENT","COLUMN ";C
250 FOR I=1 TO K:IF I=13 THEN GOSUB 550
255 READ A$
260 ? I;A$,J(I,C),:INPUT J:J(I,C)=J
270 NEXT I
280 ? "ENTER 'C' FOR ANOTHER COLUMN,'M'
FOR ":? "THE MENU.";:INPUT D$
290 IF D$="C" THEN 210
295 GOTO 510
300 GRAPHICS 0:? "        ***CALCULATING
***"
310 FOR I=1 TO 14
315 J(I,13)=0:J(13,I)=0
320 NEXT I:J(13,15)=0
380 FOR X=1 TO 12
385 FOR Y=1 TO 12
387 IF D$="4" THEN J(Y,X)=0
400 J(Y,13)=J(Y,13)+J(Y,X)
415 J(13,X)=J(13,X)+J(Y,X)
420 NEXT Y:NEXT X
430 FOR X=1 TO 12:J(13,13)=J(13,13)+J(X,
13):NEXT X
510 GRAPHICS 0:? :? "      COLUMN CALCUL
ATOR"
511 ? "       1) VIEW COLUMNS"
512 ? "       2) ENTER DATA"
513 ? "       3) DO CALCULATIONS"
514 ? "       4) ZERO THE MATRIX"
515 ? :? "WHAT IS YOUR CHOICE (1-4)";:IN
PUT D$
520 IF D$="1" THEN 40
525 IF D$="2" THEN 200
530 IF D$="3" OR D$="4" THEN 300
540 GOTO 515
550 FOR D=1 TO 38:? "-";:NEXT D:RETURN
800 DATA -,-,-,-,-,-,-,-,-,-,-,-,*,*
1000 TRAP 1000
1010 EL=PEEK(187)*256+PEEK(186)
1020 GOTO EL
```

©

TOLL FREE
Subscription
Order Line
800-227-1617
In CA 800-772-3545
Please ask for Extension 401.

www.commodore.ca

CRYSTALWARE

APPLE 48K-w/disk

THE FINEST IN FANTASY GAME SOFTWARE

ATARI 40K-w/disk

At Crystal we are doing our best to provide the finest state-of-the-art graphic adventure software in the world. Our list of credits include the first indoor-outdoor graphic adventure, the first multi-disk graphic adventure, and now for the Atari, the first graphic adventure in the world which includes screen scrolling and animation. The era of the text adventure and games which are simple combinations of static graphics and text is rapidly drawing to a close. We attempt to utilize the full potential of your computer. True, many of our games use up to 48K and we only deal in disk products, but there are a lot of users out there who have worked hard to upgrade their systems to the max and we think they deserve games that will give their computer system a run for its money.

★★★ ADVENTURE GAMES ★★★

1-THE HOUSE OF USHER- Haunted house type adventure game with scrolling in the Atari version. Wander the creepy hallways of the three story castle based on Edgar Allen Poe's short story of the same name. Written in graphics, of course, with animation and sound. We have introduced a new mystery for another $100 prize. $29.95/1 disk

2-FANTASYLAND 2041 A.D.- The largest disk based adventure game in the world (that we know of). Enter the Hall of Heroes and prepare yourself for the greatest fantasy-role-playing game you will see for years to come. To win you must survive Congoland, Arabia, King Arthur, Captain Nemo, Olympus (a sea voyage), and Dante's Inferno (Hell itself). In both the Atari and Apple versions it takes up more than 400,000 bytes of memory and uses more than 400 hires screens. The winner of the contest described in the manual with this game will receive $1000.00 and a bronze trophy. We have pushed the award date forward to February 1982 to allow more people to participate in the contest. $59.95/6 disks

3-GLAMIS CASTLE- Yes, Pat and I are on our way to Britain to stay in the dreaded Glamis Castle. If we survive our real life adventure, we'll be measuring it and will be able to provide you with a 3-D game based on this ancient haunted site where King Duncan met his end at the hands of Macbeth. Our good friend, Mark Benioff, after much research, said there's a mystery room that has never been found in this castle and a half beast, half-man creature that guards a treasure therein. Our stay will be covered by the British media and we hope to share our experience with you through the writing of this game. $49.95/2 disks

4-BENEATH THE PYRAMIDS- You are an archaeologist in 1932 and must find your way through the perilous chambers beneath the pyramids to discover a golden statue of the cat goddess Bast. This game is in hires graphics, includes sound, your little man actually moves through the corridors which you can see on the screen. The monsters are animated and very agressive. There is a new $100 prize for the first to solve the mystery; which is a toughie! $29.95/1 disk

★★★ SPACE GAMES ★★★

5-GALACTIC QUEST- An excellent combination of Star Trek and Space Trader. Battle the animated Vegan fighters as you warp from galaxy to galaxy. At the same time, you may land on and trade with hundreds of planets. Super hires graphics and lots of sound. This has been one of our most popular games. $29.95/1 disk

6-SANDS OF MARS- Take an exciting voyage to the planet Mars via the Starship Herman. This game compared to the rest, is second only to Fantasyland 2041 A.D. It includes scrolling on the Atari and hundreds of full screen graphics. You can move your little man through the terrain of Mars; if, of course, you survive the exciting journey to Mars, which occupies the whole first disk. There is a new mystery and another $100 prize just waiting for some clever adventurer out there. Good luck! $39.95/2 disks

★★★ WAR GAMES ★★★

7-WORLD WAR III- You Atari gamers will have to see this in the Atari version to believe it! If your tired of war games which take 15 minutes a move and have a manual the size of a telephone book; but still want a complex, real-time action war game-this is it! It is designed for two arm-chair generals which may manuever up to 128 seperate type of units at a time. The game displays a map of Iran & Iraq in the first scenario and later on you will find yourself moving nuclear submarines and battleships through two world wars. This is not a boring copy of a board based game but an original war game which takes a lot of skill and may take weeks to play. $29.95/1 disk

8-WATERLOO II- If you had been Napolean would you have done a few things differently? Well as you approach this final battle you are equipped with the same forces, face the same opposition, and survey the same terrain which he did. We have done a great deal of research to make this historically accurate as well as extremely complex. Even the angle of sight, fatigue of the individual soldier, and his psychological profile are included in the calculations. Oh by the way, your opposition is no slouch. You may find it more difficult to change the course of history than you think! $49.95/2 disks

★★★ ARCADIA ★★★

9-LASAR WARS- Hires-3d space war simulation. Protect the earth from alien invaders. $29.95

10-LITTLE CRYSTAL- The first of our line of education software, which will be completed by December. It includes a very fine version of Hangman, Mr. Music; which transforms the computer into a piano, Gunk-a hilarious shoot-em up game, and Storytime- an anthology of bedtime stories featuring Herman, the cat, Oscar, the Hamster, and of course, Little Crystal. $39.95

11-IMPERIAL WALKER- A fine game pack written by our Atari programmer, Michael (graphics) Potter. Includes the Walker animation which is superb, Gunfight, and Lasar Nim, a game of 'how many robots'. $29.95

12-ADVENTURE PACK- (#1-4) $112

13-SPACEOUT- (#5-6) $58

14-THE WARRIOR- (#7-8) $64

15-ARCADE- (#9-11) $60

Special Note! For those of you who live near beautiful scenic Silicon Valley, we'll be opening up a store called Country Computer at 7453 Monterey St. in Gilroy. Hopefully, we'll be carrying a full line of Apple and Atari products and invite you to drop in for our Grand Opening II on September 15.

VISA

master charge

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

(408) 778-2966 **CRYSTAL COMPUTER** **17429 Bluejay Dr., Morgan Hill, CA 95020**

www.commodore.ca

Pet, Atari, Apple: On Speaking Terms

Charles Brannon
Editorial Assistant

It would be terrific if all microcomputers spoke the same language, or at least could interpret a "universal language," but, alas, this is not the case. There are many reasons why there are so many languages and versions of the same languages: economy, memory size, expected audience, individual preference, and speed. Perhaps most responsible for this multiplicity is *progress*. "If they can do it, we can do it better." While this progress is often for the best it usually makes the process of standardizing programs difficult.

The goal of this article is to offer some suggestions on how to transport programs from one machine to another. Specifically, we'll discuss the translation from Microsoft BASIC to Atari BASIC, and vice versa. The techniques and ideas are also applicable to other program conversion problems.

The essence of translating programs is this: figure out what the foreign BASIC's statement is supposed to do, and then find a way to perform the same function in *your* BASIC. This requires that you rewrite parts of the program (it would be more accurate to say re-phrase since you don't change the logic of the statements, just their syntax). In order to effectively handle this, you must be familiar with both BASICs, and know one of the BASICs rather well. Ideally, you would be expert in both BASICs, and you really wouldn't need this article.

What I'll do is explain the differences between Atari BASIC and Microsoft BASIC, and show how incompatible statements can be re-phrased. I'll also give some specific tips on the really knotty problems.

We'll start out with the easier conversions. First, we'll work with converting algebraic statements. It is indeed easy, but there are some complications... Atari BASIC permits you to have long variable names, with every variable name being unique. Microsoft BASIC, however, only recognizes the first two characters of a variable name as significant. The problem is similar to converting "Atari Date Routines" (this issue) to Microsoft BASIC. The date routines make use of the meaningful long variable names, but as written, the program will not run properly on the PET or Apple. Microsoft BASIC will let you have long variable names like BINDATE, GREGYEAR, and LEAPYEAR, but it will interpret all references to GREGYEAR, GREGMONTH, and GREGDAY as the single variable GR, and BINDATE, BINWORK, and BININDEX as BI. The program, though, expects that these all be unique variables. The solution is to rename the conflicting variable names. GREGYEAR, GREGMONTH, and GREGDAY, become GY, GM, and GD. The other variables are similarly changed.

Incidentally, the converted program, although less readable, is completely portable, and should run on *any* BASIC. We'll go back to mathematical conversions at the end of the article.

Another area of incompatability is INPUT/OUTPUT. We won't get too specific here, since I/O isn't even standard in Microsoft BASIC. Instead, we'll work on general I/O (like PRINT) for all machines, and focus on the similarity of PET/CBM and Atari input/output. PRINT seems to be the most standard of all BASIC statements. PRINT "HELLO" will do the same thing on all BASIC-speaking machines. It is in the special formatting of a PRINT statement that problems appear. For example, the program:

```
10 FOR I=I TO 20
20 PRINT I;SQR(I)
30 NEXT I
```

will produce a list of the square roots from one to twenty. On the PET, it works fine, e.g.:

```
1  1
2  1.41421356
3  1.7320581
4  2
5  2.23606798 etc.
```

The semicolon ";" causes the value of I and the square root of I to be printed on the same line. The PET will add a space to the front of any number and a skip (cursor-right) after, so the two fields are nicely separated, but on the Atari or APPLE, they run together.

```
21.41421356
```

Instead, just change line 20 to:

```
20 PRINT I;" ";SQR(I)
```

This will insert the needed gap.

When the comma is used to separate fields, it causes much larger gaps. On most computers, the comma causes the cursor to skip to the next print position, where each print position is a set division of the screen, perhaps every 10 spaces. Keep in mind that the width between each field is different on each computer, so watch the formatting. One other item on PRINT. On the PET, and some other Microsoft BASICs, the semicolon can be left out in certain situations, but you'll get a syntax error if you try to leave it out of Atari statements.

www.commodore.ca

Osborne AWAKENS the ATARI™ and Puts it to Work.

All too often, computer users who've been used to entertainment programs have trouble coming up with ideas for **practical** computing. Your computer plays space games well enough, but when you have a down to earth problem like the future of an investment or federal withholding taxes, can it perform? It can now.

Osborne/McGraw-Hill introduces **Some Common BASIC Programs for the Atari**™. This book contains 76 short, brilliantly documented programs that you can key directly into your **Atari**™ **400 or 800** computer. You'll get a lot of math power including personal finance, taxes, and statistics. All the programs are listed below.

Future of an Investment
Future Value of Regular Deposits (Annuity)
Regular Deposits
Regular Withdrawals from an Investment
Initial Investment
Minimum Investment for Withdrawals
Nominal Interest Rates on Investments
Effective Interest Rate on Investments
Earned Interest Table
Depreciation Rate
Depreciation Amount
Salvage Value
Discount Commercial Paper
Principal on a Loan
Regular Payment on a Loan
Last Payment on a Loan
Remaining Balance on a Loan
Term of a Loan
Annual Interest Rate on a Loan
Mortgage Amortization Table
Greatest Common Denominator
Prime Factors of Integers
Area of a Polygon
Parts of a Triangle
Analysis of Two Vectors
Operations on Two Vectors
Angle Conversion: Radians to Degrees
Angle Conversion: Degrees to Radians
Coordinate Conversion
Coordinate Plot
Plot of Polar Equation
Plot of Functions
Linear Interpolation
Curvilinear Interpolation
Integration: Simpson's Rule
Integration: Trapezoidal Rule
Integration: Gaussian Quadrature
Derivative
Roots of Quadratic Equations
Real Roots of Polynomials: Newton
Roots of Polynominals: Half-Interval Search
Trig Polynomial
Simultaneous Equations
Linear Programming
Matrix Addition, Subtraction, Scalar Multiplication
Matrix Multiplication
Matrix Inversion
Permutations and Combinations
Mann-Whitney U Test
Mean, Variance, Standard Deviation
Geometric Mean and Deviation
Binomial Distribution
Poisson Distribution
Normal Distribution
Chi-square Distribution
Chi-square Test
Student's t-distribution
Student's t-distribution Test
F-distribution
Linear Correlation Coefficient
Linear Regression
Multiple Linear Regression
Nth Order Regression
Geometric Regression
Exponential Regression
System Reliability
Average Growth Rate, Future Projections
Federal Withholding Taxes
Tax Depreciation Schedule
Check Writer
Recipe Cost
Map Check
Day of the Week
Days Between Two Dates
Anglo to Metric
Alphabetize

Mc Graw Hill

Some Common BASIC Programs ed. by Lon Poole:

Atari™ edition - #53-5, $14.99 ☐
PET/CBM™ edition - #40-3, $14.99 ☐
TRS-80™ level II edition - #54-3, $14.99 ☐
Generalized BASIC edition - #06-3, $14.99 ☐

Practical BASIC Programs ed. by Lon Poole
Contains 40 additional programs with a wide range of home and business applications. (Generalized BASIC edition only) #38-1, $15.99 ☐

Make check payable to: **OSBORNE/McGraw-Hill**
630 Bancroft Way, Berkeley, CA 94710 **Dept. G7**

Phone: (415) 548-2805
TOLL FREE: 800-227-2895 Outside California

Name ____________________
Address ____________________
City/State/Zip ____________________

Plus ☐ .75/item 4th class ☐ $1.50/item UPS ☐ $2.50/item Air Mail ☐ $10.00/item Overseas
(California Residents add applicable tax.) ☐ Please send me your free catalog.

Total amount enclosed $__________ or charge my ☐ Visa ☐ Mastercard #__________ Exp. Date ___
Signature ____________________

www.commodore.ca

NEECO

WHY BUY FROM THE BEST?

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

commodore

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

ALTOS

ACS 8000-2 64K 1M	$ 4500
ACS 8000-15 64K 1M	$ 5990
ACS 8000-6 208K 14.5M	$10490
ACS 8000-7 208K 29.0M	$11690
ACS 8000-10 208K 10M	$ 8500
ACS 8000-10/MTU	$10990

NEC SPINWRITER PRINTERS

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

EPSON PRINTERS

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

DIABLO 630 PRINTER

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

APPLE

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

AMDEK MONITORS

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

INTERTEC COMPUTERS

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

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

ATARI COMPUTERS

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

NEECO carries all available ATARI Software and Peripherals.

PROFESSIONAL SOFTWARE

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

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

NEECO
679 HIGHLAND AVE.
NEEDHAM, MA 02194

(617) 449-1760
Telex: 951021

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

www.commodore.ca

NEECO INTRODUCES THE CBM VIC-20 COMPUTER!

Commodore breaks the computer price barrier —

$299.95

CBM VIC-20 PERSONAL COMPUTER

VIC-20 SPECIFICATIONS

- 8 colors - built in
- sound generation - built in
- programmable function keys
- 5K memory expandable to 32K
- standard PETBASIC in ROM
- full-size typewriter keyboard
- graphics character set
- plug-in program/memory cartridges
- low-priced peripherals
- joystick/paddles/lightpen
- self-teaching materials

* WORKS WITH ANY HOME TELEVISION

$74.95

C2N TAPE CASSETTE DRIVE

CALL NEECO TODAY FOR ADDITIONAL VIC-20 INFORMATION . . .
As the CBM VIC-20 is a "new" product, prices and specifications are subject to change w/o notice.

NEECO

NEECO WILL MATCH ANY ADVERTISED PRICE ON CBM EQUIPMENT FROM ANY OTHER COMPANY WITH PRODUCT IN STOCK.

679 HIGHLAND AVE.
NEEDHAM, MA 02194

(617) 449-1760
Telex: 951021

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

www.commodore.ca

PRINT I;ACD(I)
should be
PRINT I;A$;C$;D(I)

The INPUT statement is used to get information from a human operator. It can have the form INPUT variable name, or INPUT "prompt"; variable name. The Atari does not let you include a prompt as part of the INPUT statement. Instead, PRINT the message first, and then INPUT the data. Also, you can not INPUT directly into an array, for example, INPUT MX(Z), as you can on a Microsoft machine. Instead get the value with a temporary variable, and then assign it to the matrix variable. So instead of typing in this Microsoft statement:

INPUT "How many hours";HR(I)
use
PRINT "How many hours";:INPUT T:HR(I)=T

Also, both the PET and the Atari automatically print a question mark after the prompt, while the APPLE prompt should include the question mark if appropriate, and the PET, Atari, or other machine should delete any extraneous question marks. The last I/O statement we'll discuss is the infamous GET. This command is supposed to fetch a single keystroke from the keyboard, but the manner in which it is implemented is completely non-standard. For example, let's say we want to get a YES/NO response by letting the user type Y or N. On the APPLE, we would code:

130 GET A$

This statement will wait for the user to type a key, and then A$ will contain the character that they typed. The PET does not wait for a key to be pressed, its GET statement just attempts to fetch a key from the keyboard. If no key has been pressed, A$ will be null (no character), and we must loop until A$ actually contains a keystroke:

130 GET A$:IF A$="" THEN 130

Other machines use one of the above, or like the Atari, use something far different. On the Atari, you must first OPEN a file to the keyboard (yes, the keyboard is treated as a peripheral device), and then wait until a value is returned via the GET# command. What is returned is a number, the ASCII equivalent of the character.

100 OPEN#1,4,0,"K:"
130 GET#1,A
And instead of: IF A$="Y" THEN 1000, use
IF A=ASC("Y") THEN 1000.

The *most* incompatible aspect of computer languages are cassette and disk input/output. Printer output usually uses some variant of the PRINT command, like LPRINT, but some computers treat all output the same way. We'll now concentrate on Atari and PET input/output, since they are remarkably similar, almost to the point of compatibility.

Before any action can be performed, a file must be "opened." This delcares the type of the file, and its name, if applicable. For example, to open a file to the Commodore 2040 disk, the BASIC statement might look like this:

100 OPEN1,8,8,"0:PAYROLL,S,R"

The number one is the file number, used for further access to the file. The second number is the "device number." It tells the computer which device the file is to be opened to.

PET Device Numbers	Atari Device Names
0=keyboard	K:=keyboard
1=cassette drive 1	C:=cassette
2=cassette drive 2	S:=screen
3=screen	E:=editor
4=printer	P:=printer
8=disk drive	D:=disk drive

The third number is an optional "secondary address" which gives special information to the device. In this case, it declares which one of eight disk buffers are to be used. Inside the quotes, the 0 means drive zero, since the drives are numbered 0 and 1. The colon separates the number from the file name that follows. The file name identifies the file uniquely, and can consist of up to sixteen characters. After the file name is a comma, and then S,R. The S stands for Sequential, which distinguishes it from Program files and other types of files, and the R indicates the "direction" of the OPEN: R FOR Read, and W for Write. The same statement on the Atari would look like:

100 OPEN#1,4,0,"D1:PAYROLL"

The pound sign should be pronounced "file." The one is the file number, just like on the PET. The four specifies the direction of the OPEN. Whereas R means Read on the PET, 4 means Read on the Atari.

4=Read
6=Directory Read
8=Write
12=Read and Write

The zero corresponds to the secondary address of the PET, and is device-specific. Here a zero is used, as no number is needed by the disk drive. Inside the quotes: D1 specifies on which drive the file should be accessed. Drives can be numbered from D1 to D4, with D by itself meaning D1. (You can go up to D8 with the 815 disk drive.) The colon separates the drive number from the file name. The file name can be up to eight characters long. The first character has to be a capital letter. The remaining characters can be either a capital letter or a number. There can be an optional three-letter extension that can identify the type of file, like ADVEN.PRG, or QUICKDRAW.OBJ. Remember the difference in file name length — if necessary, abbreviate PET file names for the Atari.

Above is a list of the device numbers associated with each device for the PET. The Atari uses the first letter of the device instead (like D: or C:). To read or write to a file, the PET uses the INPUT#

www.commodore.ca

JINSAM™ Opens The Desk Top Computer Doors To Easy Application Data Management

"Your data is in good hands with the data manager from Jini Micro-Systems, Inc. JINSAM is a fast and extremely flexible data manager."

Robert Baker
Kilobaud Microcomputing

"So far, JINSAM appears to do everything my Hewlett-Packard data base does. My HP system cost $6000 and JINSAM is easier to use."

Ed Presnal
Marketing Manager
Phoenix Distributing

"We've printed over 30,000 labels with JINSAM. We no longer have an outside contract. My principal is so pleased that he's given us three more PET's and the Mother's and Father's Club (PTA) has donated a new printer with the money we've saved. We're now also using JINSAM for fundraising and accurate records for recruitment target areas."

George Morstatt
Mt. St. Michael's H.S.
New York City

JINI MICRO SYSTEMS, INC.

Box 274-C8, Riverdale, NY 10463, Telephone (212) 796-6200

Commodore Approved Software

Open 12:00 Noon–4:00 P.M.

Approved for use with WordPro™ 3,3+,4,4+
WordPro is a trademark of Professional Software, Inc.

(LABEL), Y (LABEL,X) LABEL + INDX–1

LDA STA LDX STX LDY STY

SET IFM IFP IFN IFE

6502 Assembler/Editor

- APPLE
- ATARI
- PET
- KIM
- SYM

Before you buy that off-brand Assembler/Text Editor, note that EHS is the only company that provides a line of **compatible** ASM/TED's for the **PET/APPLE/ATARI/SYM/KIM** and other microcomputers.

When you make the transition from one of these 6502-based microcomputers to another, you no longer have to relearn peculiar Syntax's, pseudo ops, and commands. Not only that, EHS ASM/TED's are the **only resident 6502 Macro Assemblers** availiable and they have been available for several years. Thus you can be sure **they work.** Our ASM/TED's may cost a little more but do the others provide these **powerful features:** Macros, Conditional Assembly, String Search and Replace, or even up to 31 characters per label? Before you spend your money on that other ASM/TED, **write for our free detailed spec sheet.**

MACRO ASM/TED

- For APPLE/ATARI/PET/SYM/KIM
- Other than our MAE, no other assembler is as powerful.
- Macros/Conditional Assembly.
- Extensive text editing features
- Long Labels
- Designed for Cassette-based systems.

$49.95

MAE ASM/TED

- For APPLE/ATARI/PET
- The most powerful ASM/TED
- Macros/Conditional and Interactive Assembly
- Extensive text editing features
- Long Labels
- Control files
- Designed for Disk-based Systems.

$169.95

VISA

EASTERN HOUSE SOFTWARE
3239 Linda Drive
Winston-Salem, N. C. 27106 USA
(Dealer Inquiries Invited)

PHONE ORDERS
(919) 924-2889
(919) 748-8446

MasterCard

.EN .BY .OS .BA .DE .CE

www.commodore.ca

and PRINT# commands. They have the form:

INPUT#1,A$ or INPUT#1,A$,E,B$
PRINT#1,"HELLO" or PRINT#1,A$;CHR$(13);E; CHR$(13);B$;CHR$(13);

The PRINT# command must place carriage returns (CHR$(13)) between each item to be printed on the same line. The INPUT# statement can either read the variables singly, or as a list. The comma immediately after the # sign is mandatory, and does not perform the usual skipping function. Atari also uses the INPUT# and PRINT# commands:

INPUT#1,A$ or INPUT#1,A$,E,B$
PRINT#1;A$ or PRINT#1;A$;CHR$(155);E;CHR$ (155);B$

The INPUT# command is identical, but notice that the Atari has a semicolon after PRINT#. A comma would cause the usual skipping, wastefully writing spaces to the disk. Also, Atari's ASCII value for the carriage return is 155, not 13. Atari has two other commands: PUT# and GET#. PUT# will write a single byte to the output device. What it sends is the ASCII value of the character to be written, e.g. PUT#1,155. GET# behaves as explained earlier (GET from the keyboard). It gets an ASCII byte, usually generated by PUT#. The PET would use PRINT#1,CHR$(A); to "PUT" a byte, and GET#1,A$ to GET a byte.

The CLOSE statement wraps it up. On the PET use CLOSE 1, on the Atari, CLOSE#1. The file is now closed, and the file number can be re-used for other files.

Another incompatibility is screen formatting. Atari lacks a TAB command. On the PET or APPLE, PRINT TAB(10);"X" will print an X at the tenth character position of the screen. It is most useful when the argument of the TAB is a variable or computed value, like PRINT TAB((40-LEN(L$))/2);L$, which will center the string L$ on a forty-column screen. The APPLE also has VTAB x which skips x lines down from the top of the screen to provide vertical positioning, and HTAB x which is like PRINT TAB(x);. Atari combines the two into the POSITION command, which will place the cursor at any (X,Y) location on the screen (e.g. POSITION 10,2). So if you know the vertical position where you're printing, POSITION 10,Y:PRINT "X" will do the same thing as the Microsoft TAB. If you don't know what line the cursor's on, just use POSITION 10,PEEK(84). Memory location 84 keeps track of the vertical position of the cursor. Alternatively, you could modify the horizontal position of the cursor without changing the vertical with POKE 85,10.

Most 6502 BASICs don't provide PRINT USING, so I won't go into that, but you can use Jim Butterfield's "Simulated PRINT USING" (**COMPUTE!** #9) or "Formatted Output for Atari BASIC" (**COMPUTE!** #10). Incidentally, if you want to simulate PRINT #1;TAB(30);N$ on your Atari printer, just send out thirty spaces, and then the string, e.g. FOR I = 1 TO 30:PUT#1,32:NEXT I:PRINT#1;N$.

Almost all programs clear the screen at times. The PET uses the PRINT command to print a special character that causes the screen to clear. It looks like a reverse-field heart in program listings. The Atari also can print a special clear screen character, or use the command GRAPHICS 0 to do the same thing. The APPLE uses a machine language ROM routine to do the job: CALL -936.

We'll now go on to the most difficult conversion — strings. I'll use the PET as the Microsoft computer reference (because it's such a small word!), but the comments apply to Microsoft BASIC in general. (Apple, OSI, and SBC BASIC, too).

Almost all computers permit you to use and manipulate strings, but the method and efficiency of this string-handling differs widely (wildly?). A string is a sequence of characters, like pearls on a necklace. Get it! *Strung* together. Both the PET and the Atari permit you to use strings easily. When you want to use a string on the PET, the string is always available — it's just another variable type. You can have any number of strings. The limitation is that the length of the string cannot exceed 255 characters. This freedom with strings results in their being used carelessly and abundantly in many programs. This can waste memory and cause the dreaded "garbage collection" delay. The Atari, on the other hand, requires that you declare each string and its length at the start of the program. It sets aside a block of memory for that string, so the memory that the string uses is allocated even before a string is filled. The command used is DIM, since it is similar to DIMensioning an array. DIM A$(20) will permit the string A$ to be used, but only 20 characters can be accessed. For your conversion, make note of each string used in the PET program. Then write a series of DIM statements at the start of the program. What length should you use? Since the PET permits up to 255 characters, 255 would be a conservvative number, but it is not conservative of memory. Eighty characters would seem to be sufficient for most strings, since that's the most that can be entered via the INPUT statement. If you can discover the maximum string length, use that. If K$ is only used to get a YES or NO answer, DIM K$(1) will only permit K$ to be one character long, so that if the user types in YES, K$ = "Y", conveniently enough. String manipulation poses another problem. It might be used to pull the slashes out of a date like 8/25/81, or to reverse the order of a person's name from JOHN DOE to DOE, JOHN. What I'll do now to show some specific examples of converting statements from PET BASIC to Atari BASIC. The reverse can be inferred.

www.commodore.ca

MICRO MINI COMPUTER WORLD

74 Robinwood Ave.
Columbus, Ohio 43213
(614) 235-5813 or 235-6058

PRESENTS

BUSINESS ENHANCEMENTS
COMPUSERVICE BUSINESS SOFTWARE
for
COMMODORE AND APPLE COMPUTER SYSTEMS

B.E.C. #3000 Series

General Ledger	(Apple II & CBM 2001/8032/2040)	$150.00
Accounts Receivable	(Apple II & CBM 2001/8032/2040)	$150.00
Payroll	(Apple II & CBM 2001/8032/2040)	$150.00
Inventory	(Apple II & CBM 2001/8032/2040)	$150.00
Mail List	(Apple II & CBM 2001/8032/2040)	$150.00
Job Cost	(Apple II & CBM 2001/8032/2040)	$150.00
Accounts Receivables with Order Entry	(CBM 2001/8032/2040)	$220.00
ROM Control Chip required with CBM System		$ 70.00

B.E.C. #4000 Series NEW for 8032/8050

General Ledger	(CBM 8032/8050)	$175.00
Accounts Receivables with Order Entry	(CBM 8032/8050)	$220.00
Accounts Payable	(CBM 8032/8050)	$175.00
Payroll	(CBM 8032/8050)	$175.00
Inventory	(CBM 8032/8050)	$150.00
Mail List	(CBM 8032/8050)	$175.00
Job Costing	(CBM 8032/8050)	$150.00
ROM Control Chip required with CBM System		$ 70.00

Special Offer, buy any two (2) of the above packages and receive a 10% discount off suggested retail. Offer expires 1 July, 1981.

Send for documentation package for $20, apply this amount towards your first purchase or return documentation in re-saleable condition for complete refund.

CALL or WRITE for MMCWI's FREE CATALOG of computer products.

B.E.C. Software is Distributed in the East by MICRO MINI COMPUTER WORLD INC.

Dealer Inquiries Invited

Introducing
THE DOUGLAS POINT OF SALE SYSTEM

The DOUGLAS POINT OF SALE SYSTEM is a data processing system operating on the Commodore Business Machines 8032 or 2001 (with BASIC 4.0 ROM's installed), and the 8050 megabyte disk computing equipment. The standard business keyboard is required. The system integrates retail sales facilities and activities with the merchantile inventory to provide a complete "point-of-sale" data processing system for the retail trade establishment.

The DOUGLAS POINT OF SALE SYSTEM supports:

- Major requirements of the retail sale transaction:
  - Normal out-of-inventory sales.
  - Non-inventory sales such as installation fees.
  - Customer returns.
  - Down payments in cash or in used equipment.
  - Cash and credit transactions.
  - Provision for tax exempt customers.
  - Control of loaned inventory items.
  - Sales tax computations for sales and returns.
  - Printed receipts with company name and receipt number.
  - Capability to add personalized notes on each receipt.
- Integration of sales activity with inventory:
  - Automatic adjustment to inventory at time of sale.
  - Customer returns posted back to inventory at time of sale.
  - Stock replenishment determined at time of sale.
  - Complete sales history captured during each sale.
- Automated processing of:
  - Mail lists of customer and distributor addresses.
  - Major item customer inventory.
  - Trade-in acceptance data for used inventory report.
  - The inventory receiving process and associated reports.
  - Service and maintenance contracts.
  - Daily sales activity report.
  - File purge and system backup.
  - Password security for unattended computer.
  - Pricing and addressing labels.

DISTRIBUTED BY:

micro mini COMPUTER WORLD Inc.
TOWN & COUNTRY SHOPPING CENTER

SUGGESTED RETAIL $750.00

CALL OR SEND FOR MMCWI CATALOG

DEALER INQUIRIES INVITED

74 ROBINWOOD AVE. COLUMBUS,OHIO 43213
(614) 235-5813 (614) 235-6058

MICRO MINI COMPUTER WORLD

74 Robinwood Ave. Columbus, Ohio 43213
(614) 235-5813 or 235-6058

PRESENTS

THE INTEGRATED COMPUTER TECHNOLOGIES

Prioress-44 Internal Motherboard:
The ICT P-44 is a 44 pin internal motherboard that facilitates expansion of your PET/CBM within the PET enclosure. The Prioress-44 is fully shielded on its underside by a massive ground plane. The connectors utilize any standard 44 pin edge card (many styles are available from Radio Shack). The following signals comprise the P-44 bus:

- +9v, −9v, +16v,GND,IRQ,RES,NMI,RDY,B02.
- BA0-BA15,BD0-BD7,BR/W,BW/R,SEL8,SEL9,SELA,SELB.
- DIAGNOSTIC SENSE,SYNC and 3 User definable.

The Prioress-44 is currently available for the new 2000 and 4000 series, and is under development for the 8000 series.

All ICT cards utilize the Prioress-44 bus.

Price:	
Prioress-44 with one connector	$79.00
Prioress-44 with six connectors	95.00
Each additional connector (specify when ordering)	4.00

The ICT Programmable Character Generator:
The ICT Programmable Character Generator is a 2K RAM replacement for the PET/CBM Character Generator ROM. The device allows the user to reprogram any or all of the 256 standard PET screen characters. The PCG also functions as 2K bytes of RAM in the $9000-$BFFF address range.

Uses of the ICT PCG:
a) Foreign character sets.
b) Math, Engineering and special notations.
c) Music notation.
d) Flow control and modeling.
e) Schematic and logic symbols.
f) Character oriented game symbols.
g) Architectural Drawings.
h)320Hx200V BIT GRAPHICS.
i) ...many, many more.

The PCG has an empty socket for the original PET/CBM ROM. With the provided external switch, RAM or ROM may be selected.

ICT provides over 128K of software and data, allowing the user to immediately utilize the graphics system with extreme ease. Software is provided on 2040 format diskette and includes:
a) 7 complete 2K character sets (Russian, Katakana +).
b) Predefined graphics (including the Real-time rotating cube).
c) Development Tools including;
Charentry - used to program characters in an 8x8 matrix.
Draw - a program that allows drawing in a 320x200 area.
Plot - two versions, x,y plotting in the 320x200 matrix. A fast assembler version and a readable BASIC version.
Screen Dump - an assembler program to dump the EXACT screen contents to a Commodore 2022 printer.

Price:	
PCG with 2040 diskette and manual	$240.00
Manual alone	7.50

The ICT HexROM:
A six socket programmable ROM board. Any three of the sockets may be programmed to become ROMs at $9000, $9800, $A000, $A800, $B000 and/or $B800. A simple BASIC POKE equates any socket to any of the above addresses.

Price:	
HexROM and manual	$110.00
DumROM (6 sockets at fixed addresses)	69.00

The ICT EPROMer:
The EPROMer will READ/PROGRAM/VERIFY the following EPROMs:
2758, 2716, 2732 (24 pin EPROMs) and
2764, 27128 (28 pin EPROMs).
To a maximum of 36 pin I/O (5V).

The software (written in assembler) will support the above EPROM types and also allow the user to define any new EPROM configurations (5V Vcc, 25V Vpp).

Price: EPROMer, software and manual $180.00

ICT Products distributed by Micro Mini Computer world Inc.

***** Special introductory offer *****
10% off all suggested retail prices
(offer expires July 1, 1981) **DEALER INQUIRIES INVITED**

Integrated
Computer
Technologies

www.commodore.ca

Microsoft	Atari
M$ = MID$(L$,I,1)	M$ = L$(I,I)
R$ = MID$(NAME$,Z)	R$ = NAME$(Z)
F$ = LEFT$(NAME$,Z)	N$ = NAME$(1,Z)
LN$ = RIGHT$(DATE$,2)	LN$ = DATE$(LEN(DATE$)-2)
B$ = B$ + N$	B$ = (LEN(B$) + 1) = N$
B$ = A$ + C$	B$ = A$:B$(LEN(B$) + 1 = C$

If you own an Atari, review the string documentation in the *Atari BASIC Reference Manual.* It will help a lot. Space does not permit a thorough discussion of string handling here.

Next, we'll briefly go into string-array simulation. You may want to refer to my article, "String Arrays in Atari BASIC" (**COMPUTE!** #11). Microsoft BASIC permits you to use string arrays. An array is much like a list; it lets you refer to a sequence of numbers by using a single variable name, using an index to specify the position of the number in the array. A string array is similar, but is a list of strings. For example, in a game of chance, you might want to have two lists — the names of the players of the game, and their current cash amount. You might use the statement:

```
PRINT "PLAYER #";I;NAME$(I);" HAS $";
  CASH(I);"LEFT."
```

to print each player's name and his booty. If you remember the Atari BASIC string syntax, you'll realize the above statement can't work, because NAME$(I) will return all the characters after and including the I'th position of the single string NAME$. This syntax prevents normal string array notation. Instead, you need to partition a single large string into many different substrings. Each substring contains the contents of each "name" in the list. To access each substring, just specify the starting and ending positions of the substring within the main string. For example, if you reserved ten-character names for all the players, NAME$(1,10) would return the name of player one. In order to use it with variables, you need a general purpose formula:

```
NM$ = NAME$((I-1)*10 + 1,I*10)
```

This will return the I'th name in the list. To assign a name to the array, just reverse the statement:

```
NAME$((I-1)*10 + 1,I*10) = NM$
```

In your use, change the ten to the maximum permitted length of each substring. Determine this as previously discussed, and change the DIM statement from something like:

```
35 DIM ARK$(20)
   to
35 DIM ARK$(20*50)
```

where fifty-character substrings are allowed. This won't solve all your problems, however. Neither RUN nor CLR will clear out the contents of the string, so you will have to do this before storing a new value, or use the techniques discussed in "String Arrays in Atari BASIC" (using a numeric array to keep track of the length of each substring, and only printing the specified number of characters).

If a program uses a lot of string arrays, your job will be arduous, but perhaps worth the trouble. (If you come across a three-dimensional string array — I once did — just give up!)

We'll finish up by going back on some mathematical incongruencies. Atari has dual-mode trigonometry — it can either interpret all arguments and return all results in either DEGrees or RADians. Microsoft BASIC treats all functions in radians, but can be changed to give degree results. If the Atari program is in the RAD mode, no changes are required, but if it has a preceding DEG statement, the following statement has to be multiplied times every argument: function (A*PI/180), where A is the argument, and PI has been defined as 3.1415927 or its equivalent. So Y = INT(150*SIN (ANG)) becomes: Y = INT(150*SIN(ANG*PI/ 180)). The PET offers an integer variable type, specified with a percent sign, e.g., A% = 18/3. To avoid conflict with the floating point variable A, it should be renamed in Atari BASIC as AINT = INT(18/3). It's too bad that the variable looks like bad English. The Atari has a function that Microsoft does not directly support:CLOG(x), or the base ten logarithm. Instead of L = CLOG(N*5), use L = LOG(N*5)/LOG(10). It works just as well.

While we're on the subject of functions, let's go into a slightly more sticky problem — how to implement defined functions on the Atari. Microsoft BASIC lets you create your own defined functions with the same syntax as built-in functions. For example, if this statement was executed at the start of a program:

```
DEF FNR(V) = INT(V*RND(1)) + 1
```

then N = FNR(X) would assign a random number from one to the value X to the variable N. When a program uses a defined function, it tends to use it a lot. What you want to do is to write the function as a subroutine. You can even label the subroutine with a meaningful variable name. So instead of defining a function, define the starting line number of the subroutine:

```
LET RANDOM = 5000,
```

and instead of using N = FNR(X), just use:

```
V = X:GOSUB RANDOM:N = V
```

The subroutine RANDOM would look like:

```
5000 V = INT(V*RND(1)) + 1:RETURN
```

There are other incompatibilities, such as graphics, but that's another article. I'll leave you with a bit of advice: when you convert a program, try to change as little of it as possible. Be especially careful with line numbering, or GOTOs and GOSUBs will confuse you into an early death. Be brave, computerists, and be hopeful — Atari is releasing a Microsoft BASIC this fall. ©

www.commodore.ca

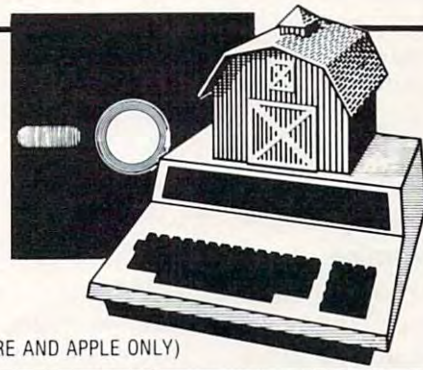
Agricultural Software from Cyberia™

NEW! NOW AVAILABLE FOR APPLE AND TRS-80

Farmer's Workbook™

Now available for Commodore, Apple and TRS-80. The most powerful management tool ever for the agricultural producer. Farmer's Workbook combines the power of Visicalc® * with the knowledge of a major midwestern university. The Farmer's Workbook is a collection of templates that are designed to be run on the Visicalc® program. The templates include labels, formulas, sample data, test cases and full documentation. Template titles include: Cattle Feeder, Pig Production, Sheep Production, Grain Management, Loan Payments, Market Average, Land Purchase and many others.

Cyber-Farmer™ The complete accounting package for today's farm operations



- Records, sorts, combines and prints the results of the farm operation.
- Account headings and numbers are pre-assigned for nearly every type of farm income or expense, but any account may be deleted, altered or added.
- Keeps personal, family and house-hold accounts as well.
- Cyber-Farmer management tools include cash-flow analysis, depreciation schedules, inventory and budget reports.
- No computer experience is necessary to operate this system.

(COMMODORE AND APPLE ONLY)

*Visicalc is a trademark of Personal Software, Inc.
MasterCard, VISA and C.O.D. orders accepted. Dealer inquiries invited.

CYBERIA inc. 515-292-7634
2330 LINCOLN WAY, AMES, IOWA 50010

LETTER QUALITY WORD PROCESSOR PRINTER/TYPEWRITER

FOR APPLE, ATARI, COMMODORE, NEC

OLYMPIA ES100

- 92 character electronic keyboard
- 8 character buffer memory
- Dual pitch, 10 and 12
- 17.5 C.P.S.
- All settings from keyboard
- Auto. correction
- Daisy type print mechanism
- Cartridge ribbons
- 14 1/8 inches writing line
- 1400 dealers nationwide

REN TEC ES

- Installation in 15 minutes using existing ES100 cables
- CMOS logic for minimal drain on ES100 power supply
- Hi or low true status bits
- Accepts RS232 serial with 7 crystal controlled Baud rates
- Accepts Centronics parallel interface
- Selectable auto. line feed

$1495.00* TYPEWRITER & INTERFACE **$295.00* INTERFACE**

RENAISSANCE TECHNOLOGY CORPORATION

3347 VINCENT ROAD
PLEASANT HILL, CALIFORNIA 94523
(415) 930-7707

RENAISSANCE TECHNOLOGY'S PRODUCTS FOR ATARI

Atari 850 to Diablo 1640/1650, 3 meter, serial cable	60.00
Atari 850 to Diablo 630, 3 meter, serial cable	60.00
Atari 850 to Epson 800, 3 meter, serial cable	60.00
Atari 850 to Anadex, 3 meter, serial cable	60.00
Atari 850 to Anadex, parallel cable	90.00
Atari Joy Stick Ports to Anadex, parallel cable	90.00
Atari 850 to Diablo 1640/1650, parallel cable	90.00
Atari Joy Stick Ports to Diablo 1640/1650, parallel cable	90.00
Atari 850 to Watanabe Miplot, parallel cable	90.00
Atari Joy Stick Ports to Watanabe Miplot, parallel cable	90.00
Atari 400 & 800 10-Key Accounting Pad	**124.95**
Atari 850 Parallel to Olympia ES100 Interface	**295.00**
Parallel Cable (required for above)	**60.00**

DEALER INQUIRIES WELCOME

www.commodore.ca

The Mysterious Age Guesser

Dr. Richard C. Vile, Jr.
Ann Arbor, MI

When I was a teacher of mathematics, I taught a course entitled Functional Math to prospective elementary school teachers. In general, the course was dull, boring, uninteresting, soporific — you think of a rap to put on a course, that course deserved it! However, from time to time I would demonstrate some mathematical trickery that at least woke up the front row. This article presents one such piece of trickery, couched in the form of BASIC programs.

The Premise

Both Programs 1 and 2 will guess your age! That is, provided you are between the ages of 1 and 63 and provided that you answer the questions given by the programs truthfully. Try one out. The first is less than a page long and can be keyed into your computer in five minutes. It is written in fairly standard BASIC (actually APPLESOFT) and should run with minor mods on most micros. The second version is in APPLE Integer BASIC and is a little more "souped up" than the first. If you are satisfied with the trick, then read no further. However, if your curiosity is aroused, the remainder of the article will explain it to you.

The Plot

The program asks a series of questions about collections of numbers. Now someone who hasn't seen how short the program is will be tempted to claim that somehow the computer is using a process of elimination to guess the answer. This impression will be especially vivid for people who are 2,4,8,16, or 32 years of age!

However, you keyed the program in. You know how short it is and that it is impossible that it is performing some mysterious elimination process or using a fancy data base stored on disk. Well then, just how does it work?

The Culprit

No, the butler didn't do it! The whole scheme rests on the *binary* system of enumeration. What you are doing, in effect, with your yes and no answers to the computer's queries is telling it your age — *in binary!*. Of course you had to be telling it your age — computers can't read peoples' minds.

When a number is expressed in binary, each digit of 1 in its numeral represents a specific power of two. The powers of two, for those of you who are *completely* non-mathematical, are:

1, 2, 4, 8, 16, 32, 64, ...

Program 1.

```
5  DEF  FN MOD2(A) =
   INT ((A / 2 -  INT (A / 2)) * 2 + .05)
10  HOME : VTAB 5: PRINT "I WILL GUESS YOUR AGE"
12  FOR I = 1 TO 2000: NEXT I
15 AGE = 0
20 POW = 1
40  HOME : VTAB 5:J = 1
50 X =  INT (J / POW)
60 X =  FN MOD2(X)
61  IF X <  > 1 THEN 70
62  IF J < 10 THEN  PRINT "     ";
63  IF J >  = 10 THEN  PRINT "    ";
65  PRINT J;
70 J = J + 1
75  IF J < 64 THEN 50
78  PRINT : PRINT : PRINT
80  INPUT "IS YOUR AGE HERE(Y/N)?";A$
85  IF (A$ = "Y") OR (A$ = "YES") THEN AGE = AGE + POW
90 POW = POW * 2
95  IF POW < 64 THEN 40
99  PRINT "YOUR AGE IS ";AGE
100  END
```

Sound familiar? These were the ages suggested above. Any one of them will have but a *single* digit of 1 in its binary representation. That means that a person whose age is one of these numbers will only say "yes" once while playing the age guessing game.

Let's examine one example in particular detail. I happen to be 38. The number 38 may be expressed as the sum of powers of two as follows:

$$38 = 32 + 4 + 2$$

To make that a little more suggestive, let's put in the powers of two that are *not* used as well as those that are:

$$38 = 0 \cdot 1 + 1 \cdot 2 + 1 \cdot 4 + 0 \cdot 8 + 0 \cdot 16 + 1 \cdot 32$$

Now, reading left to right, this may be interpreted as answers to a series of questions as follows:

Does the number 38 require a 1 in its binary expansion?
***No* (the coefficient of 1 = 0.**
Does the number 38 require a 2 in its binary expansion?
***Yes* (the coefficient of 2 = 1).**
Does the number 38 require a 4 in its binary expansion?
***Yes* (the coefficient of 4 = 1).**
Does the number 38 require an 8 in its binary expansion?
***No* (the coefficient of 8 = 0).**
Does the number 38 require a 16 in its binary expansion?
***No* (coefficient of 16 = 0).**
Does the number 38 require a 32 in its binary expansion?
***Yes* (the coefficient of 32 = 1).**

Reading the answers from top to bottom will give the exact pattern of answers to the program. Try it (pretend you're 38!).

The Corpus Delecti

Now that I've no doubt started you yawning a little, let me finish you off by explaining how the program produces the sets of numbers it displays in asking its silly little questions.

It's really very simple you see (yawn!). The

www.commodore.ca

Perfectly Balanced

educational software

from

MICRO-ED

for

PET®

and

VIC®

Send for our free catalog*

*please specify PET or VIC

MICRO-ED, Inc. • P.O. Box 24156

Minneapolis, MN 55424

or telephone us at (612) 926-2292

VIC and PET are the registered trademarks for Commodore Business Machines.

www.commodore.ca

Program 2.

```
   1 REM ==========================
   2 REM =                        =
   3 REM =   AGE GUESSING GAME    =
   4 REM =           BY           =
   5 REM =DR. RICHARD C. VILE, JR.=
   6 REM =                        =
   7 REM ==========================
  10 DIM TWOPOW(6)
  11 DIM A$(40)
  14 HOME=-936
  15 INTRODUCTION=1100:POWERS=1000
  16 DISPLAYCARD=2000:DIGITS=2100
  99 GOSUB INTRODUCTION
 100 GOSUB POWERS
 102 AGE=0
 105 FOR I=0 TO 5
 110 T=TWOPOW(I)
 115 SHOWN=0
 120 GOSUB DISPLAYCARD
 125 IF SHOWN THEN AGE=AGE+T
 130 NEXT I
 150 PRINT "YOUR AGE IS ";AGE
 155 PRINT "SOMEONE ELSE CARE TO TRY"
 160 INPUT A$
 165 IF (A$="Y") OR (A$="YES") THEN 102
 199 END
1000 REM ========================
1001 REM = SET UP POWERS OF TWO =
1002 REM ========================
1005 FOR I=0 TO 5
1010 TWOPOW(I)=0
1015 NEXT I
1020 POW=1
1025 FOR I=0 TO 5
1030 S= RND (7)
1035 IF TWOPOW(S)#0 THEN 1030
1040 TWOPOW(S)=POW
1042 POW=POW*2
1045 NEXT I
1049 RETURN
1100 REM ========================
1101 REM =      INTRODUCTION    =
1102 REM ========================
1105 CALL HOME: VTAB 5
1110 PRINT "  IF YOU ARE BETWEEN THE AGES OF 1"
1115 PRINT "AND 63, THEN I'LL GUESS YOUR AGE!!"
1120 PRINT "I WILL SHOW YOU SEVERAL SCREENS OF "
1125 PRINT "NUMBERS.  SIMPLY TELL ME ON WHICH "
1130 PRINT "ONES YOUR AGE APPEARS, AND I'LL TELL"
1135 PRINT "YOU YOUR AGE."
1140 PRINT : PRINT "READY?"
1145 INPUT A$
1149 RETURN
2000 REM ======================
2001 REM =  DISPLAY AGE LIST  =
2002 REM ======================
2005 CALL HOME
2010 VTAB 5
2015 FOR J=1 TO 63
2020 GOSUB DIGITS
2025 IF ((J/T) MOD 2)#1 THEN 2050
2030 L=5-NUMDIG
2035 FOR M=1 TO L: PRINT " ";: NEXT M
2040 PRINT J;
2050 NEXT J
2055 PRINT : PRINT "DOES YOUR AGE APPEAR ON THE LIST(Y/N)"
2060 INPUT A$: IF (A$="Y") OR (A$="YES") THEN SHOWN=1
2099 RETURN
2100 REM ======================
2101 REM = COMPUTE DIGITS IN J =
2102 REM ======================
2105 NUMDIG=1
2110 IF J>9 THEN NUMDIG=NUMDIG+1
2115 IF J>99 THEN NUMDIG=NUMDIG+1
2149 RETURN
```

program cycles through the powers of two from 1 to 32. For each power, it goes through all the numbers from 1 to 63 and asks:

Does this number require this power in its binary expansion?

It asks this via the following rule, translated into suitable BASIC statements:

n requires p ⇔ ((n divided by p) mod 2) = 1

If the power is required, then it is printed in the list preceding the question. If you answer yes to the question, the power is added in to your (ever-growing) age. When it's all over, you have told all!

Now that you are fully asleep, maybe it's time to wake up and try this out on your friends. Oh by the way, take out a piece of paper and a pencil, there's going to be a short quiz...

Just Kidding!

Program 3. Atari version.

```
100 DIM A$(1):GRAPHICS 0
105 FOR I=1 TO 5:? CHR$(127);CHR$(158);:
NEXT I:?
107 FOR I=1 TO 5:? "   ";CHR$(159);:NEXT
 I
110 PRINT CHR$(125):POSITION 2,5
120 PRINT "I WILL GUESS YOUR AGE"
130 FOR I=1 TO 1000:NEXT I
140 AGE=0
150 POW=1
160 PRINT CHR$(125):POSITION 2,5:J=1
170 X=INT(J/POW)
180 X=INT((X/2-INT(X/2))*2+0.05)
190 IF X<>1 THEN 230
220 PRINT J;CHR$(127);
230 J=J+1
240 IF J<64 THEN 170
250 PRINT :PRINT :PRINT
260 PRINT "IS YOUR AGE HERE (Y/N)";:INPU
T A$
270 IF A$="Y" THEN AGE=AGE+POW
280 POW=POW*2
290 IF POW<64 THEN 160
300 PRINT "YOUR AGE IS ";AGE;"."
310 END
```

©

TOLL FREE
Subscription
Order Line
800-227-1617
In CA 800-772-3545
Please ask for Extension 401.

www.commodore.ca

INTRODUCING . . .
TEACHER'S AID

DR. DALEY'S SOFTWARE is excited about our latest software release—TEACHER'S AID.

TEACHER'S AID is the grade management system you've been waiting for. Its many features mean that you can be free from the drudgery of hours of record keeping and grade reporting. Now you can devote more time to the pleasures of teaching.

TEACHER'S AID is easy to use, menu driven and features—

1. **Flexible class assignment structures.** This means that you can set up and keep records of any combination of homework, quiz, test, lab, etc. scores.
2. **Grade averaging done in a variety of ways.** Grade averages can be prepared using weighted scores, possible scores, tables, percent, or a combination of these methods.
3. **Student progress reports.**
4. **An individualized list of missing assignments.**
5. **Easy editing and additions to any of the files.**
6. **Reports on either the screen or printer.**

All of this power is yours for only $59.95. TEACHER'S AID comes on disk complete with comprehensive, easy to read documentation, packaged in an attractive binder.

When ordering please tell us your computer configuration. TEACHER'S AID is available on these systems:

Apple II or Apple II Plus
(32K with single disk)
Pet or CBM 2000, 3000, or 4000 series
(16K with 2040 or 4040 disk)

TEACHER'S AID will be ready soon on the Atari 800 and TRS-80 Model I or Model III.

Call or write for details of our other software offerings.

NOTE OUR NEW ADDRESS

DR. DALEY'S SOFTWARE
Water Street
Darby, MT 59829
Phone: (406) 821-3924
(Hours: 10 a.m. to 6 p.m. Mountain Time)

MasterCard VISA

CREATIVE SOFTWARE

is your source for

VIC®
PROGRAMS

VIC Games **$24.95**
- VIC Trap
- Seawolf
- Bounce Out

Household Finance **$34.95**
- Part I — Entering & Updating
- Part II — Summing & Displaying
- Part III — Budgeting & Graphing
- Part IV — Deductibles Analysis

Home Inventory **$14.95**
- Part I — Entering Inventory
- Part II — Evaluating Inventory

Logic Games **$14.95**
- Code Maker
- Code Breaker

Recreational/Educational I **$14.95**
- Hangman
- Hangmath

Recreational/Educational II **$14.95**
- Math Hurdler
- Monster Maze

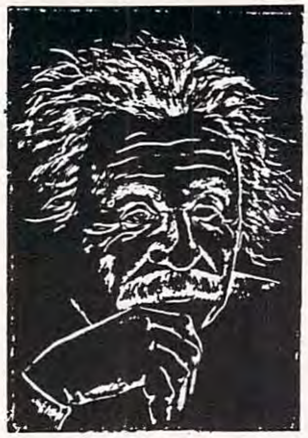
Ordering Information: *VISA/Mastercard, check or money order accepted. If charge, please include expiration date of card. Add $1.50 for shipping and handling. Calif. residents add sales tax.*

CREATIVE SOFTWARE

201 San Antonio Circle, #270
Mountain View, CA 94040
(415) 948-9595

www.commodore.ca

An Intelligent Alternative



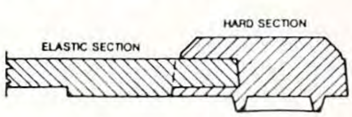
TYPRINTER 221

In the research you are doing before purchasing your computer printer, you are probably confused by the various claims, speeds, choices, shapes and prices. Well, we'd like to clear the air a bit and tell you about the most unusual computer-printer around — the TYPRINTER 221.

You see, it's unusual because it is **totally compatible** with **every** computer and word processing program . . . from the largest to the smallest. It's versatile to the point of incredibility . . . We'll discuss the broad advantages and explain the details.

THE DAISY WHEEL

The special daisy wheel supplied is of a unique design consisting of a 100 character carrying radii. Each radii is formed of two distinct types of plastic — an "elastic plastic" for the stalk of the radii, and a comparatively "hard plastic" used to form the character area. This, combined with a very narrow character profile and a special positioner on each of the 100 radii, guarantees a uniform character density. There is near perfect geometric positioning of the character with no character higher or lower than the others. And because of its unique dual material design, micro-vibrations have virtually been eliminated, leaving your final copy clean, clear and smudge free. The copy produced is comparable to that produced by metal daisy wheels and at a fraction of the cost.



THE KEYBOARD

The keyboard has been referred to as a triumph of human engineering - from the way the keys seem to have been custom designed to fit your fingers, to the way the special feature switches have been grouped. A flip of a switch (or under computer control of course) and the printer becomes a foreign language machine. Push a button, and like magic the printer automatically locates and lines up columns of figures, perfectly balanced between the margins. This incredibly fast, extraordinarily quiet electronic keyboard puts more programming power at you fingertips then printers costing five to ten times as much.

THE DISPLAY

The TYPRINTER 221 presents a new dimension in operator/machine communications. In the manual (typewriter) mode, the printer controls and verifies all entries before printing. The display exhibits the last 15 characters of the text, word-by-word, until the end of the line. The operator may control what will be printed before the actual printing takes place. This new found flexibility enables you to make modifications along the entire line and in both directions. This 20 character plasma display has the ability to scroll backwards as well as forwards; will give the operator a visual indication as to which print mode is currently being selected as well as the number of characters remaining before the right margin is reached. The display will also indicate to the operator:

The number of characters available in the memory	What characters will be inserted into an existing text.
When the printer is in an error condition	When the memory for the previous line has been selected.
When a pre programmed form lay out has been selected	A warning message that the end of the page is being approached.
When the printer is operating from the internal memory.	That a hyphenation decision must be made

PRINT MODE

The TYPRINTER 221 will allow you to automatically highlight individual characters, words or complete sentences. Whatever is entered from the keyboard or from the computer, even an existing text file, can be printed in one or more of the five different modes:

- traditional printing;
- underlined characters;
- true bold characters where the horizontal component of the character is increased without disturbing the vertical component;
- characters which are both bold and underlined, and;
- a feature unique among computer printers-printing in reverse — white on black, sort of reverse video on paper.

MULTILINGUAL CAPABILITY

A unique and useful feature of the TYPRINTER 221 is its capability of being able to print in several languages without changing the daisy wheel. In addition to English, every standard daisy wheel has the ability and the necessary characters to print in French, Spanish, Italian and German.

THE FEATURES

Automatic justification of the right margin

The electronics of the TYPRINTER 221 have made right hand justification a simple, automatic operation.

Phrase and format storage

Phrases, dates, addresses, data, etc. that may be stored in your computer's memory may be sent over to the printer and stored in one of the "memory bins" of the printer. This information may then be used by the operator in the manual mode. This can save you hours when trying to get a form "just right."

Automatic centering

The TYPRINTER 221 will not only center any title between the pre-set margins, but will also center over one or more columns, or over any specific point and will even align copy with the right margin independent of the left margin.

Automatic vertical lines

A command from the computer enables an automatic feature which prints vertical lines at any point on the paper.

Automatic tab sequence recall

With the TYPRINTER 221 you may store and recall the most frequently needed margin and tab sequences for applications such as daily correspondence, statistical reports, etc. This guarantees consistent high quality appearance of each document.

Paragraph indent

A computer command instantly sets a temporary margin in order to print one or more indented paragraphs with respect to the right margin.

Automatic decimal point location

No matter how many figures to either the left or right of the decimal point, the TYPRINTER 221 will automatically line up the figures with the decimal point in any position you choose. Statistical printing has never been easier.

Column layout

This feature allows you to obtain automatic and perfect distribution of spaces between columns in respect to the margins. A perfect page balance is assured without the need to carry out calculations or additional operations.

There is a wide variety of options that you can add to TYPRINTER 221.

By now you are probably convinced that **we are sold** on our machine, and we hope you can understand why. In fact, why don't you use these facts to measure against any and/or all the other computer printers on the market.

When you do, you will realize the TYPRINTER 221 is an intelligent electronic typewriter, a text formatter — and a brilliant computer printer — available at a suggested list price of only $2850.

TYPRINTER 221 is available at your local computer shop — or we'll tell you where you can see and try one if you call us at

HOWARD INDUSTRIES

2051 E. CERRITOS AVE., 8C
ANAHEIM, CA 92806
714/778-3443

www.commodore.ca

THE NATIONAL COMPUTER SHOWS

HAVE WE GOT A PROGRAM FOR YOU IN '81 & '82

Attend the biggest public computer shows in the country. Each show has 100,000 square feet of display space featuring over 50 Million Dollars worth of software and hardware for business, industry, government, education, home and personal use.

You'll see computers costing $150 to $250,000 including mini and micro computers, software, graphics, data and word processing equipment, telecommunications, office machines, electronic typewriters, peripheral equipment, supplies and computer services.

All the major names are there including; IBM, Wang, DEC, Xerox, Burroughs, Data General, Qantel, Nixdorf, NEC, Radio Shack, Heathkit, Apple, RCA, Vector Graphic, and Commodore Pet. Plus, computerized video games, robots, computer art, electronic gadgetry, and computer music to entertain, enthrall and educate kids, spouses and people who don't know a program from a memory disk.

Don't miss the Coming Of The New Computers– Show Up For The Show that mixes business with pleasure. Admission is $5 for adults and $2 for children under 12 when accompanied by an adult.

Ticket Information

Send $5 per person with the name of the show you will attend to National Computer Shows, 824 Boylston Street, Chestnut Hill, Mass. 02167. Tel. 617 739 2000. Tickets can also be purchased at the show.

THE MID-WEST COMPUTER SHOW

CHICAGO
McCormick Place
SCHOESSLING HALL
23rd & THE LAKE
THURS-SUN
SEPT 10-13, 1981
11AM TO 7PM WEEKDAYS
11AM TO 6PM WEEKENDS

THE MID-ATLANTIC COMPUTER SHOW

WASHINGTON, DC
DC Armory/Starplex
2001 E. CAPITAL ST. SE
(E CAP. ST. EXIT OFF I 295 -KENILWORTH FRWY)
ACROSS FROM RFK STADIUM
THURS-SUN
SEPT 24-27, 1981
11AM TO 7PM WEEKDAYS
11AM TO 6PM WEEKENDS

THE NORTHEAST COMPUTER SHOW

BOSTON
Hynes Auditorium
PRUDENTIAL CENTER
THURS-SUN
OCT 15-18, 1981
11AM TO 7PM WEEKDAYS
11AM TO 6PM WEEKENDS

THE SOUTHEAST COMPUTER SHOW

ATLANTA
Atlanta Civic Center
395 PIEDMONT AVE NE AT
RALPH McGILL BLVD
THURS-SUN
OCT 29-NOV 1, 1981
11AM TO 7PM WEEKDAYS
11AM TO 6PM WEEKENDS

THE SOUTHERN CALIFORNIA COMPUTER SHOW

LOS ANGELES
LA Convention Center
1201 SOUTH FIGUEROA
THURS-SUN
MAY 6-9, 1982
11AM TO 7PM WEEKDAYS
11AM TO 6PM WEEKENDS

www.commodore.ca

Book Review:

Video/ Computers: How To Select, Mix, And Operate

Richard Mansfield
Assistant Editor

Personal Computers and Home Video Systems, by Charles Sippl and Fred Dahl, proposed to examine the trends leading to the ultimate IVT (integrated video terminal). It is not excessively technical, yet it does examine its subject in enough detail to demonstrate, for example, why high frequency is a necessity for TV transmission. More importantly, the authors make such demonstrations clear for the less hardware-oriented readers.

The title is misleading: you are not going to really know how to select or operate video or computer devices after reading the book. Nor will you be able to jump up and interface your TV to your computer. The book is more general, more predictive. It attempts to follow two converging technologies — personal computing and video technology (including satellite, videocassette recording, data transmission, and other issues). The authors make a number of interesting and useful observations about the coming meld of computer and television devices.

For one thing, computers are digital and television is, essentially, analog. They explore this conflict and declare digital the winner — even given current transmission and memory-size constraints. Their reasoning is persuasive and much can be learned about several such issues by following their logic. Take the graphics problem: how much digital information is contained on an average 21" color TV screen? Let's say that you want to use your computer to draw a realistic, high resolution picture of the Grand Canyon or something. The TV screen has 1,200,000 bits (color dots) of information. Roughly, this would mean that you would need to program and store that many pieces of information. To simplify your drawing, you might take advantage of the fact that the bits are grouped by threes (color groups) so if you select green, then red and blue could be automatically turned off. This would bring you down to only 400,000 programming decisions. Of course you might cheat (go analog) and use a light pen or something.

Animating your picture would bring in some extraordinary additional problems: you would need a new picture 30 times per second. To give you an idea of the memory storage squeeze, you would need a 60 minute cassette to store 12 seconds of animation. It might be better to just buy a postcard. Even the most diehard futurephile will conclude that pure digitalization has its limits.

The authors do have their weaknesses. They seem to know video in somewhat greater depth than they know computing. For instance, they mention (pg. 110) that the Commodore 8032 has a "built-in color monitor." They define the Atari 400 as "the general-purpose system" and the 800 as "a specialized system." What's more, their descriptions of the rest of the home computer market are either vague, wrong, or very close to promotional literature. They also focus more on Bally, Mattel, etc. than they do on CBM, Atari, or Apple. In sum, their chapter on computers is by far the weakest in the book.

Nontheless, if you have ever wondered why such a thing as slow-scan TV exists, or what the future computer is likely to look like, or what effect CPU speed has on graphics — this book will explain these things and many others. It will not make you a hardware expert, but you will probably know much more than you did before. ©

LEARN

Having trouble learning to use your computer?

Reference manuals don't teach. Most BASIC texts don't cover specific personal computers.

TIS solves these problems with step-by-step books tailored for your machine.

For PET/CBM

Understanding Your PET/CBM $16.95
Vol 1: Basic Programming
PET Graphics .... $ 6.95

For OSI CIP/C4P

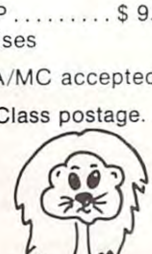
Understanding Your C1P/C4P .... $ 9.95
A Workbook of BASIC Exercises

Money Back Guarantee. VISA/MC accepted.

All prices include UPS or 1st Class postage.

TIS

Total Information Services, Inc.
Box 921, Dept. C
Los Alamos, NM 87544



www.commodore.ca

A WIDE SPECTRUM OF TITLES

Computers for Everybody
Jerry Willis and Merl Miller
This fun-to-read book covers all the things you should know about computers. If you're anxious to buy one, use one or just want to find out about them, read this book first.
ISBN 0-918398-49-5 $4.95

Microsoft FORTRAN
Paul M. Chirlian
Here is the the book for you microcomputer users who want to implement FORTRAN on your machines. Even if you've never used FORTRAN before, you will be writing and running FORTRAN programs almost at once.
ISBN 0-918398-46-0 $14.95

How to Get Started with CP/M
Carl Townsend
One of the world's most popular operating systems is explained in simple terms. Includes a handy guide on shopping for an operating system, a glossary, a list of hardware manufacturers supporting CP/M and a list of major CP/M software.
ISBN 0-918398-32-0 $9.95

Take AIM: Volume One
James H. Clark
This lab and learning manual for the AIM-65 and other 6502 microcomputers includes computer precautions, programming basics, and 30 fully documented utility and game programs which teach math, data handling, simulation and more.
ISBN 0-916460-29-0 $16.95

Year of the Robot
Wayne Chen
This thought-provoking book illustrates how a robot encroaches upon the turfs of religion, morality and philosophy, teaching us how to behave.
ISBN 0-918398-50-9 $7.95

Small Computers for the Small Businessman
Nicholas Rosa and Sharon Rosa
If you're ever considered a computer for your business but didn't know where to turn, this is the book that will arm you with all the information you'll need to make an intelligent, cost-effective decision.
ISBN 0-918398-31-2 $12.95

From dilithium Press

Write or call for free catalog! 800-547-1842

Most bookstores and computer stores carry our books. Call us on our toll free number and we'll tell you the one nearest you.
dilithium Press, P.O. Box 606, Beaverton, OR 97075

www.commodore.ca

Writing For COMPUTE!

Telecommunications, APPLE games, RTTY interfacing, math CAI for elementary students, machine language, Basically Useful BASIC, everything you wanted to know about RND ... **COMPUTE!** *is for children and for professors of physics.* **COMPUTE!** *is an encyclopedia of information on the 6502 family of computers — APPLE, COMMODORE, ATARI, OSI — and we welcome articles which instruct the novice, involve the enthusiast, or inform the expert.*

We feel that many excellent programs and articles are never mailed in. If you have not written for a magazine before — now is the time to start. By using the guidelines in **COMPUTE!'s Style Sheet** *you will be following the format used by professional writers. Don't underestimate the possible value of your favorite programs and ideas. Send them to* **COMPUTE!** *for review. If we cannot immediately accept your work, perhaps we can offer some suggestions which will lead to later acceptance for publication.*

COMPUTE!, *and our new magazine for the Commodore VIC,* **Home and Educational Computing**, *contain articles on all varieties of topics for all levels of computer sophistication. So, if you've got something you like — send it in and share it with the rest of us.*

COMPUTE! Style Sheet

Most of the following suggestions are common to all magazines and serve to improve the speed and accuracy of publication. **COMPUTE!** is primarily interested in new and timely articles on APPLE, COMMODORE, ATARI, OSI, etc. — the entire family of 6502 computers. For this reason, we are much more concerned with the content of an article than with its style.

These guidelines, however, permit your good ideas and programs to be more easily edited and published:

1. The upper-left corner of the first page should contain your name, address, telephone number, and the date of submission.

2. The following information should appear in the upper right corner of the first page. If your article is specifically directed to one make of computer, please state the brand name and, if applicable, the BASIC or ROM or DOS version(s) involved. In addition, please indicate the memory requirements of programs. **COMPUTE!** uses the Butterfield Convention when naming Commodore ROM versions: Original, Upgrade, and 4.0 ROMs are the correct names.

3. The title of the article, underlined, should start about 2/3 of the way down the first page.

4. Following pages should be typed normally, except that in the upper-right corner there should be an abbreviation of the title, your last name, and the page number.

For example: Memory Map/Smith/2.

5. Short, five to 20 line programs can easily be included within the text. Longer programs should be separate listings. Program listings help us to evaluate articles more easily and should be included with all articles. It is also essential that we have a copy of the program, recorded twice, on a tape or disk. The tape or disk should be labeled with the author's name, the title of the article, and, if applicable, the BASIC/ROM/DOS version(s). Tapes are fairly sturdy, but disks need to be enclosed within plastic or cardboard mailers (available at photography, stationery, or computer supply stores).

If in spite of all your best efforts, you are unable to furnish a program listing, please don't hesitate to submit a manuscript because of that.

It is far easier for others to type in your program if you use CHR$ (X) values and TAB (X) or SPC (X) instead of cursor manipulations to format your output. For five carriage returns, FOR I = 1 TO 5:PRINT :NEXT is far more "portable" to other computers with other BASICs and also easier to type in. And, instead of a dozen right-cursor symbols, why not simply use PRINT SPC (12)? A quick check through your program — making these substitutions — would be greatly appreciated by your editors and by your readers.

6. Where possible, please provide a sample of the program RUN output and, for machine language, a BASIC loader program.

7. If your article is accepted and you have since made improvements to the program, please submit an entirely new program listing, a new tape or disk, and a new copy of the article reflecting the update. We cannot easily make revisions to programs and articles. It is necessary that the author send the revised version as if it were a new submission entirely, but be sure to indicate that your submission is a revised version.

8. All lines within the text of the article should be spaced so that there is about 1/2 inch between them. A one inch margin should be left at the right, left, top, and bottom of each page. No hyphens should be used at the ends of lines to break words. And please do not justify. Leave the lines ragged.

9. Standard typing paper should be used (no onionskin or other thin paper) and typing should be on one side of the paper only (upper/lower case).

10. Sheets should be attached together with a paper clip. Staples should not be used.

11. A good general rule is to spell out the numbers zero through ten in your article and write higher numbers out (1024). The exceptions to this are: Figure 5, Table 3, TAB (4), etc. Within ordinary text, however, the zero through ten should appear as words, not numbers. Also, symbols and abbreviations should not be used within text: use "and" (not &), "reference" (not ref.), "through" (not thru).

12. For greater clarity, it is best to use all capitals when referring to computer languages (BASIC, ALGOL), keyboard references (RETURN, TAB, ESC, SHIFT), BASIC words (LIST, RND, GOTO), and computer names (APPLE, CBM, ATARI, OSI).

13. If possible, it is best to locate machine language programs in a memory area common to all machines. In

www.commodore.ca

this way, JMP instructions and internal JSR's will not need modification. To illustrate, starting a machine language routine at 826 decimal is fine for all PETs except those using BASIC 4.0 which uses this memory area. Starting the routine at 864 will permit all PETs to run the program. Perhaps the best memory area, for the greatest number of computers, would be in the 8000 decimal area (above BASIC, yet under the 8K memory limit).

14. COMPUTE! pays between $25 and $250 for published articles. In general, the rate reflects the length of the article.

15. COMPUTE! welcomes the idea that authors of long programs will offer to make tape or disk copies to save other readers the typing time. However, we request that the copying fee be $3.00 Such articles should conclude with the remark that interested readers must send a stamped, self-addressed mailer and a blank tape or disk. Be aware, though, that you might be inundated with requests.

16. Articles can be of any length — from a single-line routine to a multi-issue series. The average article is about four, double-spaced, typed pages.

17. If you want to include photographs, they should be 5x7, black and white glossies.

18. We purchase all rights to your manuscript and software. **COMPUTE!** will cooperate with authors on specific programs on a case-by-case basis to allow the author to share commercial rights in the software should any opportunities other than publication in printed form arise. ©

MICRO EXPRESS
301-792-2341

ATARI	
800 Microcomputer w/32K	727.00
810 Disk Drive	450.00
815 Dual Disk Drive (Double Density)	1,100.00
830 Acoustic Modem	160.00
850 Interface Module	140.00
ALTOS	
8000-15D 208KB 2 1mb Disks	5200.00
8000-6D 308KB 14 Meg Hard	7200.00
8000-10D 208KB 10 Meg Hard	7200.00
AXLON	
256K-Mem Expansion Box 2/64K	650.00
32K-Axlon 32K Memory Module	140.00
EPSON	
MX-100 132 Column Printer	725.00
MX-80	465.00
MX-80 FT	565.00
MX-80 FT w/Graphics	635.00
HAYES	
Smart Modem	235.00
MM II Apple Modem	300.00
MM 100	320.00
NEC	
Spinwriter 5530	2500.00
PC-8001 A Micro w/32K	995.00
PC-8012 A I/O Unit w/32K	650.00
TELEVIDEO	
TVI 910 Terminal	595.00
TVI 920 C Terminal	875.00
TVI 950 Terminal w/Function Keys	975.00
TS 80 950 CRT, Z80A, 64K	1,400.00
NORTHSTAR	
2D-64K 64KMEM, 2 DD Disks	3,195.00
2Q-64K 64KMEM, 2QD Disks	3,500.00

WE CARRY THESE LINES:
Adds
Altos
Amdek
Anadex
Atari
Axlon
Dymarc
Dysan
Epson
Hayes
LJK
Microtek
Mountain
NEC
Northstar
Peachtree Software
Personal Software
Sanyo
Silverman
Televideo
Trendcom

MICRO EXPRESS
12602 Silverbirch
Laurel, MD 20708
(301) 792-2341

VISA **MasterCard**

We ship same day by UPS Freight Collect.
5% charge for MasterCard & VISA. MD Residents add 5% sales tax.

Introducing a totally new world in professional software for the Apple II and Commodore Pet.

Until recently, the tremendous explosion in microcomputer technology left an unfilled void in real quality software. MicroCraft Systems is totally dedicated to filling that void for the Apple II and the Commodore Pet. With powerful, elegant and versatile software.

Beyond the software house. To a complete 6502 development system.

More than a software distributor or discount house, MicroCraft Systems now provides you with access to a whole new spectrum in software systems. MicroCraft software includes: MacroLink, a complete macroassembler for the 6502, SuperEdit, a full-screen editor with a superb range of powerful capabilities plus, DiskScreen, a screen-oriented disk utility. Complete System $200/$40

MacroLink $125/$20

Uses text files for source code • Links source code or object code programs • Fully nestable file include capability • Recursive macros • Nestable conditional assembly • Over 50 error messages • File I/O macros • Sample programs • Disk assembler: source code need NOT fit in memory.

SuperEdit $75/$20

Full Screen Editor • Single keystroke commands • Uses standard text files • Move cursor by character, line or page • Block move and copy • Search and replace • Macro capability • Plus extra software: SuperEdit '56 x 27' provides 56 x 27 character screen for Apple II • And more!

DiskScreen $40/$10

Displays a complete sector in hex and ASCII • Completely visual and interactive • Sector move and copy • To change byte value, move cursor and type.

NOTE: All programs require single disk drive. MacroLink also requires 48K. First price: software and manual. Second price: manual only (applied toward purchase of software).

Call or write for our complete MicroCraft Systems catalog at no extra charge. Other programs include: S.T.A.R. Reading Program • "C" Compiler • Graphics Development Systems • Programs in ROM • SuperEdit for 80 x 24 video cards

MicroCraft Systems, Inc.
In software systems, a name to remember.

188 Kerby Road
Grosse Pointe Farms, MI 48236
(313) 881-3089

Source I.D. CL0472 **MicroNet I.D. 70270, 222**

VISA or MasterCharge accepted. Inquire at your local computer store or order direct.

www.commodore.ca

THE apple® GAZETTE

A Tape "EXEC" For Applesoft:

Loading Machine Language Programs

Sherm Ostrowsky
Goleta, CA

Apple owners with Disk systems have available a very powerful DOS command, "EXEC", which will effectively turn control of the computer over to a text file on the disk. The lines in this file are treated as if they had been typed in at the keyboard in Immediate mode, and are executed. Unfortunately, we owners of "obsolete" cassette-tape based systems don't have the benefit of this capability. But, in this article I will show you how to obtain some of the power of an "EXEC" file on tape. I'll demonstrate the method, which is actually quite general, by showing how to load Machine-Language (ML) programs just as easily as you now load Applesoft programs, and how to combine Applesoft and ML loads on one cassette in an effective manner. It has been said that most apple owners have disks, but I suspect that those who still use tape include a high proportion of relative beginners, so this article will be slanted toward them.

Some of the programs in my library are in Applesoft and others are in ML, but all of them are still on cassettes. As you are probably aware, these two different types of programs must be loaded into the computer by entirely different commands. An Applesoft program is loaded very simply, by typing LOAD. You don't have to know how long the program is or where in memory it is supposed to be stored; Applesoft takes care of all those details for you. But a ML program is a pain in the neck to load. First you have to enter the Monitor by typing CALL -151. Then you have to know the exact addresses of the beginning and end of the program, so you can type:

(Begin Address).(End Address)R

to start the loading process. And woe unto you if you are off by even one byte in remembering where the programm is supposed to go: you'll get that dreaded "beep" and "ERR" message.

And, after it's loaded, the difference between Applesoft and ML programs continues to exist, to the discomfort of the latter. To run the Applesoft program, you type RUN — what could be simpler? To run the ML program you have to know its Entry Address, which may or may not be the same as its Begin Address; then you type (still in the Monitor)

(Entry Address)G

to get it started. You have to keep referring to written notes in order to load and run a ML program successfully.

Well, I got tired of all this. I wanted to load all my programs, whether in Applesoft or ML, in exactly the same way — by typing LOAD. And I wanted to run them all the same way — by typing RUN. The computer has a better memory than I have, so let *it* keep track of where the darn ML program begins and ends, and where to enter it. After a while, I found a way to do this, and I'll describe it to you below. In the process, I discovered that the method would also solve some other problems connected with how to combine Applesoft Programs with ML subroutines in a convenient fashion. These, too, I shall pass on to you.

Although the method I am about to describe is very easy to use, it is actually based upon some rather intimate details concerning the inner workings of Applesoft. So, as a byproduct, I hope this article will add to your knowledge in this area, so vital to making fullest use of the capabilities of the Apple.

Let us begin by solving the problem of how to simplify the loading and running of a single ML program. We'll assume that you start out with the program already in the computer's memory, having been loaded (for the very last time, let us hope) by the tedious old method. We must also assume that you can, if you wish, SAVE the ML program back onto a cassette tape by typing:

(Begin Address).(End Address)W

in the Monitor. This last assumption may be more of a stumbling block than you may think, since some commercial programs are "protected" so that they cannot easily be copied, i.e., SAVEd onto another cassette. Sorry, folks, but if that is the case with your program, then I can't help you.

Now, leave the Monitor and enter the Applesoft level by typing Control-C (Return), and type in an Applesoft loader program like the one I'm going to show you below. The example is for a

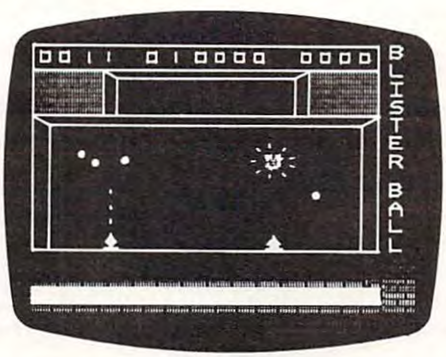
www.commodore.ca

The BEST games are from Creative Computing Software

1978: Adventure
1979: Air Traffic Controller
1980: Super Invader
1981: Blister Ball and Mad Bomber

Blister Ball

Blister Ball is the first completely original arcade-type game for a computer. Not a copy, not an adaptation, not a spinoff. **Blister Ball** is new—it's a new idea—better than Invaders, better than Circus, better than Asteroids, better than Galaxian. If you've played other games for hours, you'll play **Blister Ball** for days.



How does it work? Well, some mean but fun-loving aliens have produced some bouncing bombs. First they drop one and you've got to position yourself under it and zap it with your laser. If you miss, that's OK. It will bounce around, although each bounce is lower, and you have several chances to zap it. Got the hang of it? OK, here come two bouncing bombs. You zap them. Then you're faced with three, then four and five.

As they bounce longer and longer the walls begin to close in so you're faced with either zapping the bombs or being hit. Each hit knocks you a little further toward the gutter. But you can survive two hits which is usually enough to zap all the bombs.

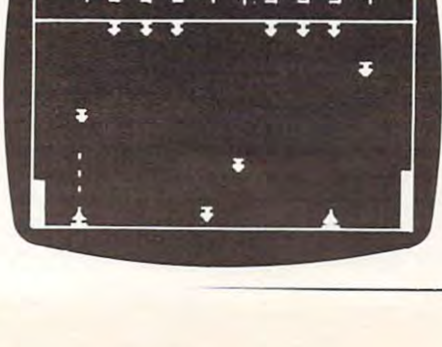
Feeling confident? Don't. Because after 5 bombs the murderous little devils drop 5 bonus bombs, worth ten times as much. These don't bounce, so you get only one shot. You need nerves of steel and the reflexes of a tail gunner.

After you complete one round,the game starts again with bombs that bounce faster and lower (and are worth more) than the previous ones.

Blister Ball is a fantastic solo game. But there are two-player options as well in which players can play as a team or as opponents. Each player can move the entire width of the screen and zap any of the bombs. Here, you're not only trying to survive, but trying to outscore your opponent. The game has two skill levels.

Mad Bomber

In **Mad Bomber** you are faced with aliens in a huge ship hovering overhead. They have bomb racks which they constantly fill with bombs. Your object is to move from side to side on the ground and zap the bombs in the bomb racks or as they fall.



As the game progresses, the aliens fill up their bomb racks more quickly and the bombs fall faster. You lose after ten bombs have hit the area which you are defending.

Mad Bomber can be played by one player solo or by two players as a team or as opponents. Two skill levels.

Order Today

Blister Ball and **Mad Bomber** are available together for $29.95 on disk (DOS 3.2) only and require a 48K Apple with paddle controls. (We recommend using the Super Paddles from Peripherals Plus).

To order send $29.95 plus $2.00 shipping and handling to the address below. Credit card customers should include card number and expiration date of Visa, MasterCard or American Express card. Credit card orders may also be called in to our toll-free number in the continental U.S.

If you also wish to order a set of Super Paddles from our Peripherals Plus subsidiary, the cost is just $39.95. The paddles are backed by a 90-day limited warranty from the manufacturer as well as Peripherals Plus' moneyback guarantee of satisfaction.

Blister Ball and **Mad Bomber** are colorful, challenging, fast and noisy. They are the games of the year from Sensational Software.

creative computing

Attn: Hope
39 E. Hanover Avenue
Morris Plains, NJ 07950
Toll-free **800-631-8112**
In NJ 201-540-0445

www.commodore.ca

specific program that I use a lot: my Assembler. And the example has had a few unnecessary bells and whistles added to it to enhance its convenience to me; you may want to leave these off for your application. Instead of describing the program in the usual way, with a lot of REM statements, I intend to do a far more thorough job of explaining it in the following text. So here's my Loader program, and the explainations come after it.

```
 10 REM APPLESOFT LOADER FOR
 20 REM THE S-C ASSEMBLER
 30 :
 40 HOME : VTAB 12: HTAB 8: PRINT
    "LOADING THE S-C ASSEMBLER"
 50 PRINT :X= POS(0)
 60 Y$ ="1000.24FFR   D823G"
 70 FOR I=1 TO LEN(Y$): POKE 511+I, ASC
    (MID$ (Y$,I,1) )) +128: NEXT
 80 POKE 72,0: CALL - 144
 90 T=POS (0): IF T>X+1 THEN 200
100 POKE 214,85
110 PRINT CHR$ (7);"LOAD SUCCESSFUL —
    STOP TAPE": PRINT
120 FOR PAUSE =0 TO 2000: NEXT
130 CALL 4096: END
199 REM LOADING-ERROR EXIT
200 PRINT CHR$ (7); CHR$ (7); CHR$ (7);
    "*** LOADING ERROR ***": PRINT
210 END
```

Here is the explanation.

Lines 10–30 just tell what the program is for.

Line 40 is one of my "bells-and-whistles." It isn't necessary for proper operation of the program, but I find it comforting. It displays a message on the screen telling me what is going on, and keeps the message there for me to look at while the tape is being read. Some of these ML programs take a L-O-N-G time to load, and you sometimes begin to wonder if the computer is still doing anything.

Line 50 is also not strictly necessary, but it is very useful. It is part of an error detection scheme to keep me from trying to run the program if it didn't load in correctly. The Apple keeps a running tally of a checksum during the load process, and will give an "ERR" message if it fails to agree with the value that accompanies the program on the tape (thereby indicating that something has gone wrong in the loading), but, other than this message, the Apple doesn't set any error flags that can be read by a program. So here, before we even begin to load the tape, we record, in variable X, the horizontal position of the cursor. This will be used in line 90 (below) to determine if a loading error has taken place.

Lines 60–80 are the heart of the loader. They constitute a clever scheme by which an Applesoft program can, in effect, fool the computer into believing that you have typed in the line:

(Begin Address).(End Address)R

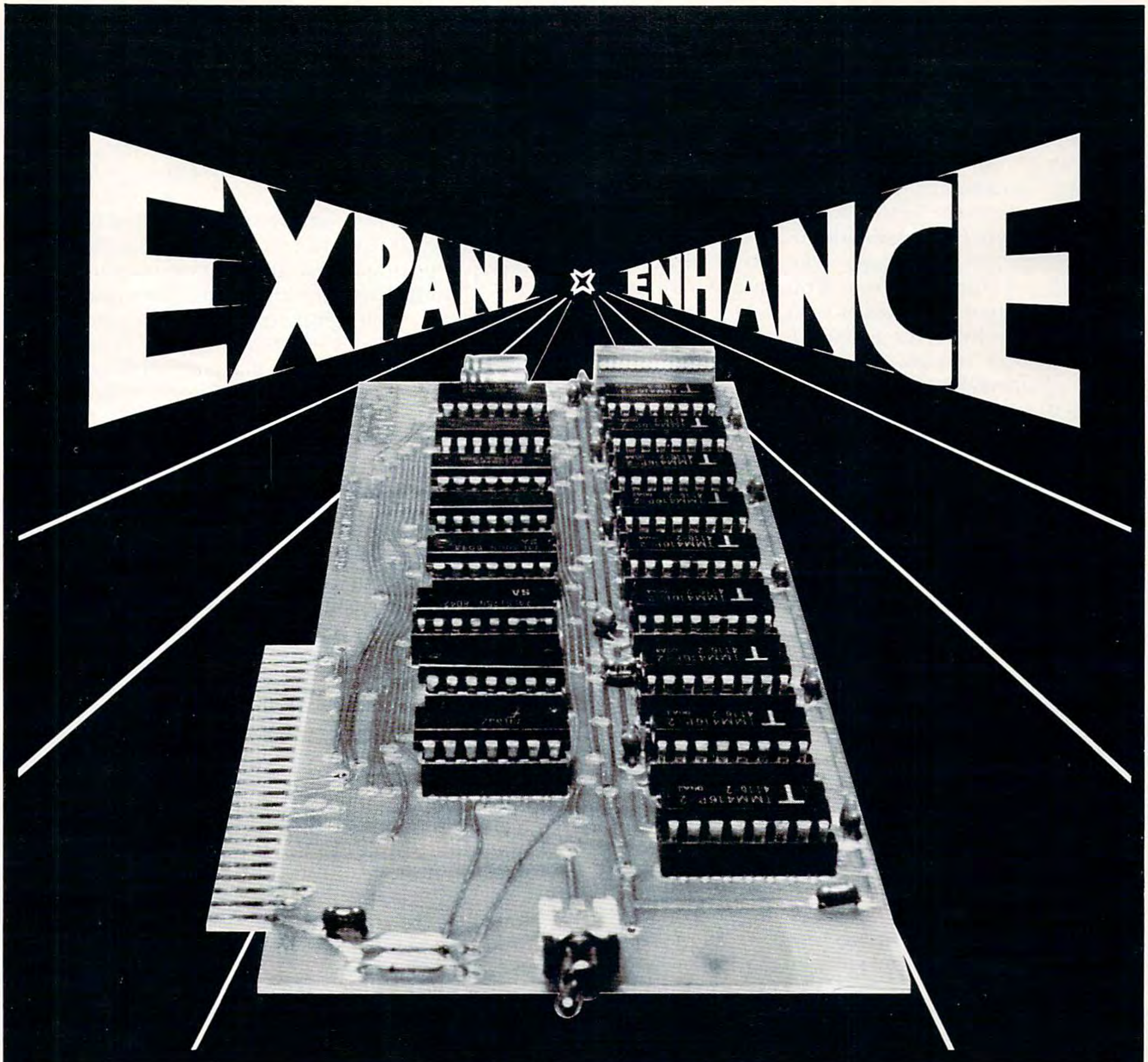
by way of the Monitor! It was invented by S.H. Lam. In line 60, the string variable Y$ contains a sequence of literal Monitor commands, just as you would have typed them in by way of the keyboard. The first part is the instruction to Load the ML program starting at address $1000 and ending at address $24FF (the "dollar sign" signifies a hexadecimal number, in 6502 notation). There follows an obligatory space, to separate this Monitor command from the next one. The second and last command on this line is "D823G", which tells the Monitor to execute an Applesoft subroutine located at $D823. This particular subroutine happens to be the so-called "running return" to Applesoft, after which the computer will begin to execute whichever proper Applesoft command it encounters next.

You'll notice, however, that so far this Monitor command line is still resident in a string variable; how do we get the Monitor to see it and execute it? Well, line 70 pokes this string, one byte at a time, into memory starting at location 512 (in decimal). But 512 is equivalent to $0200, the start of the Apple's keyboard input buffer where it goes to find every new line after you have typed it in. So the effect of line 70 is to place the pseudo-input line defined in line 60 into the input buffer. Those who are particularly observant may be wondering about the reason for adding 128 to the value produced by the ASC command, before POKEing it into the buffer. This is due to a little known incompatibility between Applesoft and the Monitor in the way they interpret ASCII character codes. Strangely enough, although Applesoft uses "true ASCII," in which the highest bit (bit 7) of each byte is off (i.e., = 0), the Monitor uses a different version of ASCII in which bit 7 of each byte has to be on (i.e., = 1). The addition of 128 (decimal) turns this bit from off to on.

Now line 80 gets the Monitor to look into the keyboard buffer and execute whatever commands it finds there. The POKE of 0 into location 72 is just a precaution, to make sure that no strange values have gotten into the location which will be stored in the Processor Status Register when the Monitor call is executed. Those of you who know something about the operation of the 6502 Microprocessor will understand what this means; for the rest of you it is of no great significance — it just needs to be done to prevent possible trouble. Finally, the command CALL-144 jumps to the Monitor subroutine referred to above: the one that scans the input buffer and executes whatever commands it sees there.

As I mentioned above, lines 60-80 are the heart of the technique being discussed in this article. But I want to emphasize that the procedure outlined in the past few paragraphs is *extremely* powerful and quite general. By using it, you can make the Apple execute any commands which can be input by way of the Monitor, such as moving ranges of memory around, storing machine language pro-

www.commodore.ca



16K RAM EXPANSION BOARD FOR THE APPLE II* $195.00

The Andromeda 16K RAM Expansion Board allows your Apple to use RAM memory in place of the BASIC Language ROMs giving you up to 64K of programmable memory. Separate Applesoft* or Integer BASIC ROM cards are no longer needed. The 16K RAM Expansion Board works with the Microsoft Z-80 card, Visicalc, DOS 3-3, Pascal, Fortran, Pilot, and other software. A switch on the card selects either the RAM language or the mainboard ROMs when you reset your Apple.

The Andromeda 16K RAM Expansion Board has a proven record for reliability with thousands of satisfied customers.

Now with One Year Warranty.

*Apple II and Applesoft are trademarks.

ANDROMEDA
INCORPORATED
P.O. Box 19144
Greensboro, NC. 27410
919 852-1482

Distributed By:

COMPUTER DATA SERVICES
P.O. Box 696
Amherst, NH. 03031
603 673-7375

www.commodore.ca

grams wherever you wish, or any of the other things discussed on pages 39–60 of the *Apple II Reference Manual*. But you can do all this from *within* a running Applesoft program, without ever stopping to enter the Monitor or typing in any commands at the keyboard. To those with a fertile imagination, the possibilities inherent in such a capability are enormous — enough to fill several articles as long as this one. You can have some fun thinking up some ideas of your own.

Meanwhile, let's get back to the subject at hand. After executing line 80, the computer should have loaded the ML program from tape into the specified location in memory. Line 90 checks the horizontal position of the cursor after the load has been completed. If the loading failed, the computer will have printed out the message "ERR", and so the cursor will be three spaces farther to the right of where it was before the loading process began. In this case, line 90 causes a jump to line 200, the error exit. Here the "bell" is beeped thrice (those CHR$(7)s) to wake me up, an appropriate message is printed on the screen, and the program ends, to let me rewind the cassette and try again.

But this doesn't happen very often — the Apple cassette system has been very reliable for me. So, usually, upon completing the tape load, the program goes to line 100. This is another very important line whose significance, however, cannot be easily explained at this point in the discussion. Let us put off the explanation of line 100 until we have finished looking at the remainder of the program. There's not much left to say. Line 110 lets me know, with a "beep" and a message, that the loading process has been successfully completed and reminds me to turn off the tape recorder. Line 120 causes a delay of about three or four seconds to give me time to see and act on that message, because line 130 causes the program to begin executing.

This may need a bit of comment. Although it is necessary, in Applesoft, to RUN to start a program after loading it, I think that most of the time the user would be just as happy to have the program begin running as soon as the load was completed, if only Applesoft had such a "LOAD-AND-GO" command. Certainly in the present example, since I know that the entry address to initialize my Assembler is at $1000 (decimal 4096), I prefer to have the loader program do this for me by doing a "CALL 4096". You can "Load-and-Go" your own ML programs in the same way by putting an equivalent CALL to the entry address in your version of this loader.

However, if you insist on retaining the two-step process, and want to be able to start your program by typing RUN in the regular Applesoft manner, the program can easily be modified to do this instead. Just replace line 130 with the following:

```
130 DEL 10,130: END
140 CALL 4096: END
```

The new line 130 causes the whole front part of the loader to self-destruct (in memory only of course, not on your cassette), leaving only line 140 as the first active command. Now typing RUN executes just this one remaining line, making your ML program start running at its entry address.

This has been a rather exhaustive description of a short Applesoft program, but since it contains several techniques which may be new and unfamiliar to many readers and since these techniques seem to me to be of great usefulness, I thought it worthwhile to explain thoroughly.

One Of Applesoft's Least-Known Features

But we're not quite done explaining yet. There is one more technique which is required to make the loader perform properly. And this is perhaps the most mysterious and least-known of all the features of Applesoft, so even some of you semi-pros might be able to learn something new from the next few paragraphs.

As things now stand, the loader program and your ML program have not yet been joined together on tape so that the former can help you to load in the latter. You will recall that, before I started describing the loader program, I left you with your ML program already in memory. Now you should also have typed in a customized version of the loader program, with the beginning and ending addresses in string Y$ replaced by the values appropriate to your ML program, and your entry address (in decimal) replacing my "4096" in line 130 (or 140 if you chose to go that route). (By the way, I hope that your ML program didn't occupy any of the memory spaces now containing the loader (from $0800 to $09A2 in my case), since I forgot to warn you about this unfortunate way to clobber the whole thing.) Assuming that all is still well, you now want to put both the loader and the ML program onto tape, with the loader first, of course. But, before you hasten to type SAVE to put the Applesoft loader program on tape, wait just a little longer while I explain the last secret.

The secret is this: *before* you SAVE the loader, type in the following Applesoft command in Immediate Mode (to be executed from the keyboard):

```
POKE 82, 128
```

This seemingly innocuous command is the key to making the loader behave like an EXEC file — doing its job without human intervention. It represents an almost totally undocumented feature of Applesoft and works like this: any Applesoft program which is SAVEd to tape after this POKE has first been executed will AUTO-RUN as soon as it has been LOADed! That is, if you rewind the SAVEd tape and type LOAD, Applesoft will not only load in the program, but will also immediately

www.commodore.ca

HUNTINGTON COMPUTING

ONE OF THE WORLD'S LARGEST INVENTORIES

Now Selling Atari, PET, TRS-80 Software

Atari

ADVENTURE INTERNATIONAL

Adventureland (cass.)	$19.95 now	$16.74
Pirate's Adventure (cass.)	$19.95 now	$16.74
Mission Impossible (cass.)	$19.95 now	$16.74
Voodoo Castle (cass.)	$19.95 now	$16.74
The Count (cass.)	$19.95 now	$16.74
Strange Odyssey (cass.)	$19.95 now	$16.74
Mystery Fun House (cass.)	$19.95 now	$16.74
Pyramid of Doom (cass.)	$19.95 now	$16.74
Ghost Town (cass.)	$19.95 now	$16.74
Savage Island I (cass.)	$19.95 now	$16.74
Savage Island II (cass.)	$19.95 now	$16.74
Angle Worms/Crolon Div. (cass.)	$9.95 now	$8.54
Deflection (cass.)	9.95 now	$8.54
Mountain Shoot (cass.)	$9.95 now	$8.54
Sunday Golf (cass.)	9.95 now	$8.54
Galactic Empire (cass.)	$19.95 now	$16.74
Star Trek 3.5 (cass.)	$19.95 now	$16.74
Lunar Lander (cass.)	$19.95 now	$16.74

ARTSCI

Poker Solitaire (cass.)	$14.95 now	$12.74
Gomoku (cass.)	$19.95 now	$16.94
Reversi (cass.)	$19.95 now	$16.94
Cypher Bowl (cass.)	$29.95 now	$25.44

AUTOMATED SIMULATIONS

Rescue at Rigel (cass.)	$29.95 now	$25.44
Star Warrior (cass.)	$39.95 now	$33.94
Invasion Orion (cass.)	$24.95 now	$21.24
Datestones of Ryn (cass.)	$19.95 now	$16.94

AVALON HILL GAME COMPANY

Empire of the Overmind (disk)	$35.00 now	$29.74
Conflict 2500 (cass.)	$15.00 now	$12.74
Empire of the Overmind (cass.)	$35.00 now	$29.64
Tanktics (cass.)	$24.00 now	$20.44

DATASOFT

Atari Mailing List (disk)	$24.95 now	$21.24
Atari Character Generator (disk)	$19.95 now	$16.94
Text Wizard (disk)	$99.95 now	$84.44
Atari Character Gen. (cass.)	$15.95 now	$13.54
Le Stick	$39.95 now	$33.94

ON-LINE SYSTEMS

HI-RES Adv #2 - Wiz & Princess (disk)	$32.95 now	$27.94

PERSONAL SOFTWARE

Visicalc (disk)	$199.95 now	$164.44
Checker King (cass.)	$19.95 now	$16.94
MicroChess (cass.)	$19.95 now	$16.94

UNITED SOFTWARE OF AMERICA

Survival/Adventure (disk)	$24.95 now	$21.24
3-D Supergraphics (disk)	$39.95 now	$33.94
3-D Supergraphics (cass.)	$39.95 now	$33.94

VERSA COMPUTING

Mind-Bogglers I (disk)	$19.95 now	$16.94
Mind-Bogglers I (cass.)	$15.95 now	$13.54
VersaWriter Graphics Tablet	$300.00 now	$254.44
Hidden Words	$17.50 now	$15.44
Spatial Relations	$17.50 now	$15.44
Word-Scramble	$15.00 now	$13.54
Preschool Fun	$15.00 now	$13.54
Fastgammon (cass.)	$19.95 now	$16.94
Assembler (cass.)	$24.95 now	$21.14
6502 Disassembler (cass.)	$11.95 now	$10.14
6502 Disassembler (disk)	$14.95 now	$12.64
Tank Trap (cass.)	$11.95 now	$10.14
Tank Trap (disk)	$14.95 now	$12.64
Tari Trek (cass.)	$11.95 now	$10.14
QS Forth (disk)	$79.95 now	$67.94
Starbase Hyperion (disk)	$22.95 now	$19.44
Name That Song (cass.)	$14.95 now	$12.64

Pet

AVALON HILL GAME COMPANY

B-1 Nuclear Bomber (cass.)	$15.00 now	$12.77
Midway Campaign (cass.)	$15.00 now	$12.77
No. Atlantic Convoy Raider (cass.)	$15. now	$12.77
Nukewar (cass.)	$15.00 now	$12.77
Conflict 2500 (cass.)	$15.00 now	$12.77
Planet Miners (cass.)	$15.00 now	$12.77
Computer Acquire (cass.)	$20.00 now	$16.97
Lords of Karma (cass.)	$20.00 now	$16.97

AUTOMATED SIMULATIONS

Introductory 3-Pack (disk) (Rescue, Morloc's, and Datestones)	$49.95 now	$39.97
Rescue at Rigel (cass.)	$29.95 now	$25.47
Temple of Apshai (cass.)	$39.95 now	$33.97
Hellfire Warrior (cass.)	$39.95 now	$33.97
Starfleet Orion (cass.)	$24.95 now	$21.27
Invasion Orion (cass.)	$24.95 now	$21.27
Morloc's Tower (cass.)	$19.95 now	$16.97
Datestones of Ryn (cass.)	$19.95 now	$16.97

PERSONAL SOFTWARE

VisiCalc (disk)	$199.95 now	$167.77
Checker King (cass.)	$19.95 now	$16.97
Gammon Gambler (cass.)	$19.95 now	$16.97
MicroChess (cass.)	$19.95 now	$16.97
Bridge Partner (cass.)	$19.95 now	$16.97
Time Trek (cass.)	$19.95 now	$16.97

UNITED SOFTWARE OF AMERICA

KRAM (disk)	$99.95 now	$84.97
Super KRAM (disk)	$175.00 now	$148.77
Request (disk)	$225.00 now	$191.27
Thinker (disk)	$495.00 now	$420.77
Space Intruders (cass.)	$19.95 now	$16.97
All MICRO-ED		10% Off List
All Microcomputer Workshops		15% Off List

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

TRS-80

BIG FIVE SOFTWARE

Super Nova (cass.)	$15.95 now	$13.58
Galaxy Invasion (cass.)	$15.95 now	$13.58
Attack Force (cass.)	$15.95 now	$13.58
Cosmic Fighter (cass.)	$15.95 now	$13.58
Meteor Mission II (cass.)	$15.95 now	$13.58

BRODERBUND SOFTWARE

Galactic Trilogy (disk)	$39.95 now	$33.98
Galactic Empire (cass.)	$14.95 now	$12.68
Galactic Trader (cass.)	$14.95 now	$12.68
Galactic Revolution (cass.)	$14.95 now	$12.68
Tawala's Last Redoubt (cass.)	$19.95 now	$16.98

DATASOFT

Iago (disk)	$24.95 now	$21.18
Football Classics (disk)	$24.95 now	$21.18
Arcade-80 (disk)	$24.95 now	$21.18
Iago (cass.)	$19.95 now	$16.98
Football Classics (cass.)	$19.95 now	$16.98
Arcade-80 (cass.)	$19.95 now	$16.98
Sigmon (COLOR) (cass.)	$29.95 now	$25.38
SECS (COLOR) (cass.)	$29.95 now	$25.38

ACORN SOFTWARE

Invaders From Space (disk)	$20.95 now	$17.78
Duel-N-Droids (disk)	$20.95 now	$17.78
Pinball (disk)	$20.95 now	$17.78
Pigskin (disk)	$20.95 now	$17.78
Quad (disk)	$20.95 now	$17.78
Basketball (disk)	$20.95 now	$17.78
Gammon Challenger (disk)	$20.95 now	$17.78
Everest Explorer (disk)	$20.95 now	$17.78
SuperScript (disk)	$29.95 now	$25.38
System Savers (cass.)	$14.95 now	$12.68
Invaders From Space (cass.)	$14.95 now	$12.68
Duel-N-Droids (cass.)	$14.95 now	$12.68
Pinball (cass.)	$14.95 now	$12.68
Pigskin (cass.)	$14.95 now	$12.68
Quad (cass.)	$14.95 now	$12.68
Basketball (cass.)	$14.95 now	$12.68
Gammon Challenger (cass.)	$14.95 now	$12.68
Everest Explorer (cass.)	$14.95 now	$12.68
All Adventure International		15% Off List
All Automated Simulations		15% Off List
All Avalon Hill		15% Off List
All Hayden		15% Off List
All Microsoft		15% Off List

Apple

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

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

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

HUNTINGTON COMPUTING
Post Office Box 1235
Corcoran, California 93212

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

SUPER DISCOUNTS
apple® SOFTWARE

VISA
master charge

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

www.commodore.ca

begin running it *without waiting for you to type RUN*. So Applesoft does, after all, have an AUTO-RUN command; you just have to know how to get at it.

Now the background for making the Tape-Exec loader is complete. Do the POKE, then SAVE the loader program, then enter the Monitor and SAVE your ML program by typing:

(Begin Address).(End Address)W

in the usual way. (In the case of my Assembler, for example, I used 1000.24FFW to SAVE it.) Now try it out. Rewind the tape, and type LOAD in the good old Applesoft way. The loader will be loaded and will immediately begin to run by itself, causing your ML program to be loaded too, in accordance with the instructions placed in its Y$ string by you. From now on, it will be as easy to load this ML program as any Applesoft program.

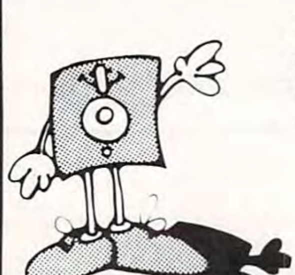
There is just one potential problem with all this, but I have taken care of it by the as-yet-unexplained POKE in line 100 of the loader. You see, the magic words "POKE 82,128" which you invoked before SAVEing the loader constitute a much more powerful spell than I have yet indicated. They do more than just cause an Applesoft program (in this case, the loader) to Auto-Run. They also completely lock up Applesoft so you can't use it very much. It will allow you to RUN the program in memory, but any other valid or invalid Applesoft command will be ignored. You won't be able to LIST, SAVE, alter, or do anything else to the program as long as the effects of that POKE remain active. This is a very powerful magic you have invoked here, but it would take us too far afield from the main topic to explain all its ramifications now.

Fortunately, however, it is not hard to undo the effects of that magic from within the loader program itself (although quite difficult, and sometimes impossible, to undo it from outside a running program!) Line 100 is the required antidote. It leaves everything just as you are accustomed to having it in an Applesoft environment. ©

(Continued in next issue.)

Dysan CORPORATION

Solve your disc problems, buy 100% surface tested Dysan diskettes. All orders shipped from stock, within 24 hours. Call toll FREE (800) 235-4137 for prices and information. Visa and Master Card accepted. All orders sent postage paid.



PACIFIC EXCHANGES
100 Foothill Blvd
San Luis Obispo, CA
93401 (In Cal. call
(805) 543-1037)

SEPTEMBER IS "TAKE CARE OF BUSINESS" MONTH

VISICALC

Take any problem you would normally work out on paper in rows and columns, use VISICALC and see why it is probably the single most valuable program yet developed for personal and professional use. VISICALC, the electronic worksheet that performs arithmetic calculations INSTANTLY.

Regular Price: $199.95
September Special: $174.95
Cat No. 2718 Apple II/II + , 32K disk
Cat No. 2808 Atari 800, 32K, disk

The Home Money Minder

Been looking for an easy to use home management tool? This program includes: checkbook balance, budget analysis, income and expenditures with graph comparisons and more.
Regular Price: $34.95 September Special: $29.95

Cat No. 2938 Apple II + , Applsft, 48K disk

How To Order

Prices include UPS ground shipping & handling (USA only). Pay by check, M.O., Visa, M/C or C.O.D. ($1.40 ADDL) Mention this AD to qualify for special prices. Offer expires 10/1/81.

HW ELECTRONICS

19511 Business Center Dr., Dept. G9
Northridge, CA 91324

(800) 423-5387 (Outside Calif.)

(213) 886-9200 (In Calif.)

Send for our FREE catalogue Today!

www.commodore.ca

Text Composition On The Apple II Plus

R. R. Hiatt, John Rustenburg and
Stefan Demmig
St. Catharines, ONT

Text composition on the Apple II Plus presents two problems, interfacing to some sort of printer and distinguishing between upper and lower case on both CRT and printer output. The first of these is readily solved by the Apple II Reference manual: We are pleased to report that the circuit given as Figure 1 on p. 118 requires no modification for interfacing a Decwriter II to the Apple II Plus. The software (TTYDRIVER, p. 119), does require a small change: deleting the text window width setting to 72 (replacing the code in $378-$37B with NOP's). This avoids the system crash that results when control is returned to the CRT with a text window greater than 40. Furthermore, text window setting is more flexible when incorporated in the BASIC calling routine.

Upper vs. lower case with the standard Apple keyboard is trickier. The shift key is live only for dual function keys such as @/p and 7 N. The CTRL key is also dual purpose; e.g., if CTRL M were to be interpreted as cap M, there would be no unambiguous signal for carriage return. Fortunately, the ESC key can be made to suit the purpose after a bit of fooling around to see how it affects code received at the keyboard inport ($C000).

Programs 1 and 2 are short ESC demo routines. ESCDEMO1 shows the transient nature of the ESC effect. (Key ESC; then before all 16 27's are printed, key a letter.) ESCDEMO2 is a little more amusing. The ESC, (CHR$(27)), is captured in an apparently infinite loop. Subsequent keying of a letter, however, breaks the loop and results in a print of *both* "UPPERCASE" and "LOWERCASE," as if both of the two mutually exclusive IF clauses were being followed. (Of course, they are, but not as it immediately appears. We leave it to the reader to figure out the logical paradox.)

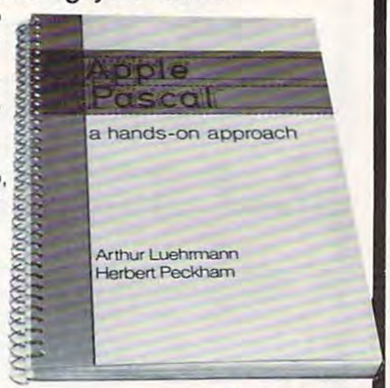
Program 3 gives a simple text composition routine, employing both DECWRITER (our name for the modified TTY driver) and ESC for upper case letters. The main program, starting at line 400, augments the routine of Program 2 by capturing the ESC'ed ASCII code and then resetting the keyboard strobe. (Resetting the strobe first de-ESC's the value.) To facilitate corrections, the text is echoed to the CRT, with left arrow (–) activated for erasure, and is sent to the printer only after a <CR> (end of line).

Training oneself to use ESC for upper case, rather than shift turns out not to be as difficult as it might seem, as long as upper case letters are distinguished on the CRT in some way. We have taken the route of setting upper case to FLASH via the code in the subroutine at 200.

The single character FLASH requires a POKE at the appropriate screen memory address. While the base vertical address can be worked out from the vertical cursor position (PEEK(37)) and a base 8 algorythm, it turns out that the base address for TEXT/LORES graphics is easily obtained by PEEK(40) + 256*PEEK(41). Adding PEEK(36) (horizontal) to this gives the cursor position.

Obviously Program 3 is not a text editor or even a proper front end for one. It does, however, solve what we have felt to be the major problems — those involving the system. The rest is simply a matter of creative BASIC.

APPLE PASCAL™
A HANDS-ON APPROACH
Gives you everything you need to use the most powerful language your computer can have.



User-tested, step-by-step, A-to-Z approach features hands-on experiences in creating, running, and debugging programs. Volume is spiral-bound to lie flat by the keyboard while it takes you from basics to advanced programming and graphics applications. No math background is required. Only $14.95. Order today — put the power of Pascal to work for you.

Also available
BASIC: A Hands-On Method
Second Edition
Herbert D. Peckham,
Spiral-bound, 306 pp., $12.95

Mail the coupon today

Mail to:
D. LaFrenier
McGraw-Hill Book Company
1221 Avenue of the Americas
New York, N.Y. 10020

SAVE MONEY: Remit with order and we pay all shipping and handling costs. Full return privileges still apply.

Please send me the books checked for 15 days' free examination. At the end of that time I will pay for the books I keep, plus local tax, postage and handling, and return any unwanted books postpaid.

APPLE PASCAL (49171-2), $14.95
BASIC: A HANDS-ON METHOD (49160-7), $12.95

Name ______
Address ______
City/State/Zip ______

62-U444-6681-3

www.commodore.ca

Program 1. ESCDEMO1

```
10 GET Q$
20 FOR I = 49152 TO 49167
30 PRINT PEEK (I), I: NEXT: GOTO 10
```

Program 2. ESCDEMO2

```
10 P = 49152
20 GET Q$
30 IF PEEK (P) = 27 THEN 30
35 CH = PEEK (P)
40 IF CH > 127 THEN PRINT "UPPER CASE":
   GOTO 20
50 IF CH < 128 THEN PRINT "LOWER CASE":
   GOTO 20
```

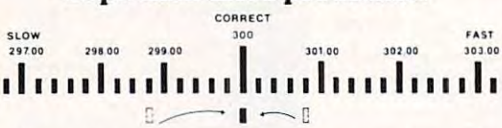
TOLL FREE
Subscription
Order Line
800-227-1617
In CA 800-772-3545
Please ask for Extension 401.

★★★★★★★★★★★★★★★★★

Bugs in your Apple?

DDT

Disco-Tech's Disc Drive Timer program
zaps disc drive problems!



Analyze disc drive motor speed on a routine basis with an adjustable real-time speedometer. Accurate to one-tenth of one r.p.m. out of 300 r.p.m.

Fine-tune disc drive motor speed yourself. All you need is DDT, two screwdrivers, and five minutes' time.

$29.95 Postpaid
California residents add 6% tax
Diskette & complete manual.

Engineering Business
Architecture Utilities
Surveying

Also available for TRS-80 Model I.

To order or for more information, write or call

disco tech T.M.
microcomputer products
a division of
Morton Technologies, Inc.
600 B Street
P.O. Box 11129 • Santa Rosa, CA 95406
707/523-1600
Dealer inquiries invited

★★★★★★★★★★★★★★★★★

```
10 REM SECRETARY
20 DIM LN% (200)
30 D$ = CHR$ (4): B$ = CHR$ (7): P =
   -16384: Q = -16368
40 PRINT "PROGRAM FOR WRITING HARD COPY
   TO DEC II": PRINT: INPUT "WANT
   INSTRUCTIONS ? "; Q$: IF Q$ > = "Y"
   THEN GOSUB 600
50 INPUT "LINE LENGTH FOR DEC II ? "; LW
60 IF PEEK (880) < > 169 THEN PRINT D$;
   "BLOAD DECWRITER"
70 GOTO 400
100 REM SBR BACKSPACE
110 IF LL = 0 THEN RETURN
120 LL = LL -1
130 PRINT CHR$(8);
140 RETURN
200 REM SBR FLASH CAPS
210 HZ = PEEK (36)
220 PN = 256 * PEEK (41) + PEEK (40) +HZ
230 POKE PN, CH - 128
240 POKE 36, HZ + 1
250 RETURN
300 REM SBR WRITE TO DEC II
310 CALL 880
320 POKE 33, LW
330 FOR I = 1 TO LL
340 PRINT CHR$( LN%(I));
350 NEXT
360 PRINT
370 POKE 33,40: PR# 0
380 PRINT
390 RETURN
400 REM CHARACTER INPUT
410 CALL - 936
420 LL = 0
430 UC = 0
440 GET Q$: CH = PEEK (P)
470 IF CH = 8 THEN GOSUB 100: GOTO 430
480 IF PEEK (P) = 27 THEN UC = 1: GOTO
    480
490 CH = PEEK (P)
500 POKE Q, 0
510 IF CH > 64 AND CH < 91 THEN CH = CH +
    32
520 LL = LL + 1: IF LL = LW - 8 THEN
    PRINT B$;
530 IF LL > LW THEN LL = LL - 1: GOSUB
    300: LL = 1
540 LN%(LL) = CH
550 IF UC THEN GOSUB 200: GOTO 430
560 PRINT Q$;: GOTO 440
600 REM INSTRUCTIONS
610 HOME : PRINT "TYPE NORMALLY, BUT USE
    ESC KEY FOR
620 PRINT "UPPER CASE LETTERS>  (UPPER
    CASE ON": PRINT "SCREEN IS SET TO
    FLASH)": PRINT
630 PRINT "THE SHIFT KEY IS STILL USED
    FOR UPPER": PRINT "SYMBOLS ON DUAL
    FUNCTION KEYS": PRINT
640 PRINT "TO END PROGRAM KEY CTRL A"
650 FOR I = 1 TO 3000: NEXT: PRINT:
    RETURN
```

©

www.commodore.ca

Algebra String
A Self-Altering Program For The Apple-II

Winston Cope
St. Petersburg, FL

BASIC is essentially an arithmetic language. Its symbol manipulating capability is used mainly to provide conveniences for the user, to provide instructions for the user, or to give headings. An algebraic expression is part of the program text, and is considered a calculation.

There is no easy way to operate on a mathematical expression itself, for example, to take a derivative. A program must be written which inputs a mathematical expression as a string and yields another string as output. Applesoft provides string manipulation commands which make this possible. The expression is still a string, however, and there is no easy way to derive numbers from it, to graph it, for example.

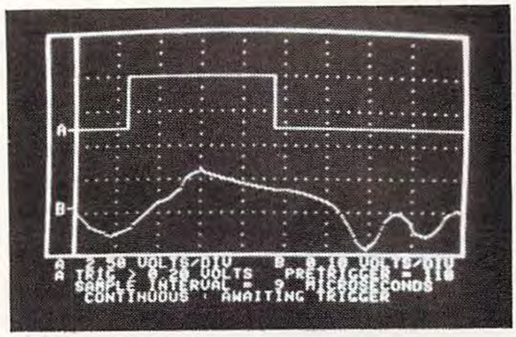
"ALGEBRA STRING" is a demonstration of how a mathematical string expression may be transformed into an arithmetic variable expression which can be used by the program. The concept behind this program is to take an algebraic string expression, Y$, to expand it to a standard length, and to poke it back into the program text itself at the proper position. Care must be taken to translate operation symbols, such as +, into their token form.

This program considers a function Y of X, and subroutine 2000 performs a simple listing of an array Y(X), for X going from 1 to N. Subroutine 62100 inputs the expression as Y$, and expands to a length of 50 characters, by concatenating "+"s and a final "0". Subroutine 62200 takes this expanded expression and POKES it into memory so that it appears at the proper place in the program text, here, at step 1020, beginning at memory location Z. Arithmetic operators are represented in strings as ASCII, but have token values when used for calculation, so this subroutine performs these substitutions. The arithmetic expression in the program whose place is taken by Y$, in step 1020, must have the same length as Y$, here 50.

Subroutine 62000 determines Z. This is simply done by finding the memory location which contains a "+," such that the next 5 locations also contain "+." The odds are very small that this would happen anywhere else than Z. LO and HI could be 0 and 64000, but for a particular program one can markedly narrow the range of the search.

When the operator is finished entering expressions to evaluate, the program will initial-

APPLESCOPE
DIGITAL STORAGE OSCILLOSCOPE
Interface for the Apple II Computer



The APPLESCOPE system combines two high speed analog to digital converters and a digital control board with the high resolution graphics capabilities of the Apple II computer to create a digital storage oscilloscope. Signal trace parameters are entered through the keyboard to operational software provided in PROM on the DI control board.

- DC to 3.5 Mhz sample rate with 1024 byte buffer memory
- Pretrigger Viewing up to 1020 Samples
- Programmable Scale Select
- Continuous and Single Sweep Modes
- Single or Dual Channel Trace
- Greater than or less than trigger threshold detection

Price for the two board Applescope system is **$595**

For futher information contact: **RC ELECTRONICS INC.**
7265 Tuolumne Street
Goleta, CA 93117
(805) 968-6614

Dealer Inquiries Invited

CLASS OF '82 . . . BOOT UP!

ASSISTANT PRINCIPAL - Whether you're too small for a big computer or if you just want to bring your administrative duties under your control, the "Assistant Principal" is just what you need. It prints school rosters, allows you to input teacher grades, prepare report cards, and maintain student master records. You should have no more trouble with student scheduling, grade averaging or recording of student grades. For Apple II and Apple II Plus, two disk drives, 48K Memory and an 80 column printer. Item M-3839-11, Price $500.00.

ROLL CALL - Let this program keep track of that all important attendance data. Whether you need information on an individual's tardy, absence history or need to prepare attendance reports for the board or your state funding agency, this complete system will make the task easy work. The system will create its own data base or work off the Assistant Principal's files. For Apple II or Apple II Plus, two disk drives, 48K Memory, and an 80 column printer. Item M-52. Price $250.00.

SYSTEMS ANALYST PLUS - Whether you're a novice computer student, teacher or principal, you will find this program most helpful. Even the advanced programmer should find these seventeen teaching sessions on programming useful for brushing up on the language and commands. For Apple II or Apple II Plus with Applesoft, one disk drive, and 32K Memory. Item M-40. Price $69.95.

Call our toll free number for a free catalog on professional, business, accounting and educational programs.

(800) 854-0561
In Ca. (800) 432-7257
Ext. 802 or write:

MONUMENT COMPUTER SERVICE
Village Data Center
P.O. Box 603
Joshua Tree, California 92252

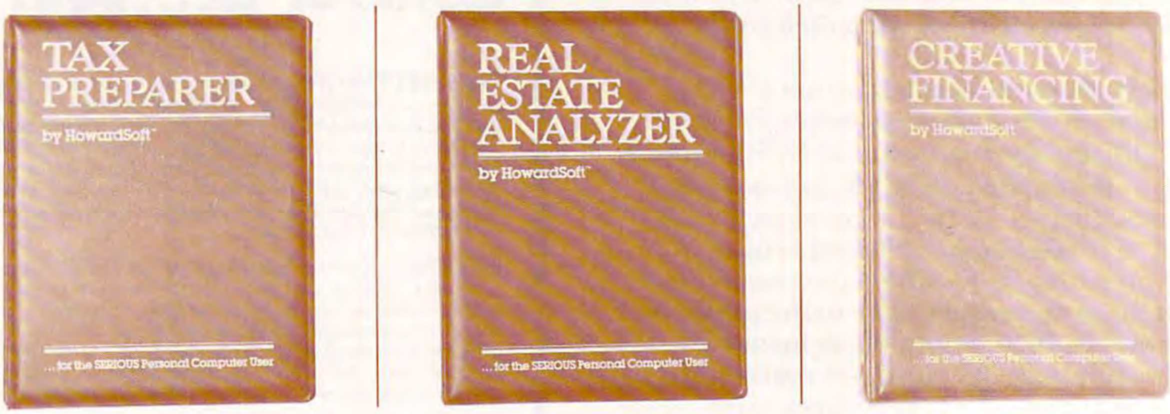
www.commodore.ca

ize itself by filling up the expression at 1020 with "+"'s so it can be run again.

```
5 ALGEBRA STRING
10 PRINT "ENTER # OF POINTS": INPUT X
100 LO=2000:HI=2500
105 DIM Y(N)
110 GOSUB 62000
115 GOSUB 1000:GOSUB 2000
120 PRINT "ALL DONE? (Y/N)":GET C$
125 IF C$="N" THEN 115
130 Y$="++++++": GOSUB 62200
135 END
1000 REM LOAD ARRAY Y(X)
1005 PRINT "ENTER ALGEBRAIC WITH
DEPENDENT VARIABLE X":INPUT Y$
1010 GOSUB 62100:GOSUB 62200
1015 FOR X=1 TO N
1020 Y(X) = + + + + + + + + + + + + + + +
+ + + + + + + + + + + + + + + + + + + + +
+ + + + + + + + + + + + + + + + + + + + +
+ + + + + + + + + + + + + + + + + + + + +
+ 0
1025 NEXT X
1030 RETURN
2000 REM DISPLAY Y(X)
2005 J=0
2010 FOR X=1 TO N
2015 PRINT X,Y(X):J=J+1
2020 IF J=20 THEN PRINT:PRINT "PRESS ANY
KEY":GET C$:J=0
2025 NEXT X
2030 RETURN
62000 REM FIND Z
62010 FOR J=0 TO 5
62015 IF PEEK(I+J) <> 200 THEN 62030
62020 NEXT J
62025 Z=I: RETURN
62030 NEXT I
62035 PRINT "CANNOT FIND Z": END
62100 REM STANDARD LENGTH FOR Y$
62105 IF LEN(Y$)<49 THEN Y$=Y$ + "+":GOTO
62100
62115 RETURN
62200 REM SUBSTITUTE Y$ INTO PROGRAM TEXT
62205 FOR I=1 TO LEN(Y$)
62210 Q= ASC(MID$(Y$,I,1))
62215 IF Q=43 THEN Q=200
62220 IF Q=45 THEN Q=201
62230 IF Q=47 THEN Q=203
62240 IF Q=94 THEN Q=204
62245 NEXT I
62250 RETURN
```

©

A NEW standard in software QUALITY . . .



. . . for the SERIOUS personal computer user

These packages will turn your Apple Computer into a powerful financial tool previously accessible only to the largest corporations. These packages are so unique in the completeness and sophistication of both the software and the documentation that they have been acclaimed by business professionals everywhere. Yet the software is so friendly that it can be operated by the financial or computer novice. See for yourself at your local dealer.

HSS Howard Software Services
. . . for the SERIOUS Personal Computer User
6713 Vista del Mar | La Jolla, CA 92037 | (714) 454-5079

www.commodore.ca

THE ATARI® GAZETTE

Positioning Player-Missile And Regular Graphics In Memory

Fred Pinho
Valley Cottage, NY

Have you ever used PM graphics only to notice funny-looking colored lines or dots on the screen with your carefully crafted images. When you moved your player or missile, these lines and dots seemed to acquire a life of their own. While it was fascinating to watch this "extra" display, it also quickly became frustrating to your programming attempts. The problem is that all the instructional articles I've seen tell you that you must step-back in RAM a minimum of 1K (4 pages) for double-line resolution and 2K (8 pages) for single-line resolution. They either ignore, or barely mention in passing, the important fact that you must also allow for the screen display memory in this calculation. The Atari uses two blocks of memory to control the TV screen display. Residing at the very top of RAM is the Display Data. This block of memory contains a bit map for the TV screen in graphic modes 3-8 and a character map for text modes 0-2. Residing just below the Display Data is the Display List. This block of memory is essentially a short program that tells the Atari how to set up the TV screen for the desired mode. The total memory required for the Display List and Display Data varies with the graphics mode used. This is illustrated in Table 1. As you can see, the highest resolution mode, GR.8, requires the most RAM.

Thus, the explanation for the "extra bonus" lines or dots in your PM display is that the program did not step-back far enough into RAM and consequently located the PM data in the Display Data memory area. The Atari then obediently displayed this data both from the normal display and through the PM system. Since the Display Data is displayed as a number of bytes per line (Table 2), you will see a line of varying colored dots. By contrast, the PM display is organized to display the bytes in a "stack" arrangement and so you see the desired figure (hopefully as you designed it).

To aid you in using PM graphics, Table 1 gives the number of pages that must be stepped back in memory (from the top of available RAM) to avoid interference between the two systems. For those not familiar with the concept of paging, the memory addressing system of the 6502 microprocessor within the Atari is based on the concept of a memory *page*. Each page is equivalent to 256 bytes of memory. Thus there are four pages of memory in each K (1024 bytes) of memory.

Note that, in calculating the step-back value for Table 1, a restriction must be observed: positioning for the PM RAM must be on a 1K Boundry for double-line resoltion and on a 2K boundry for single-line resolution. If you position the PM memory incorrectly the PM data will not be displayed. Since Atari will be equipped with a varying amount of memory, it must be able to keep track of the amount available so that it knows where to locate the display data and display list. This is done at memory location 106 (RAMTOP). If you PEEK this location, you'll find the number of pages, *not the number of bytes*, in your machine. You can get the number of bytes by multiplying by 256. POKEing into this location can be very useful for the programmer. One example is the location of large machine language programs that must be placed in a secure location that is not touched by the BASIC system. One way to accomplish this is to POKE a lower number of pages into RAMTOP, fooling the computer into believing that it has less memory than is the case. Then you can load your machine code in this safe hiding place yet still access it when needed. Another use is as a safe location for a redefined Atari character set. Again, there is one restriction. The relocated Display Data cannot cross a 4K boundry (Graphics modes up to 7). If, you don't observe this restriction, you'll find that you will be unable to plot and draw on part of the screen. Ramtop for Graphics 8 must be lowered in multiple 4K blocks. If you try it otherwise, you'll see wierd and unwanted displays on your screen.

I hope these tables aid you in using the PM and Graphics systems. The systems are powerful and unique to the Atari and their use will result in increasingly sophisticated displays.

www.commodore.ca

GRAPHICS MODE	TOTAL MEMORY BYTES ALLOCATED TO: DISPLAY DATA — Bottom Text Window	Unused Bytes — Always	Unused Bytes — Conditional	Text or Graphics Screen	DISPLAY LIST — Unused Bytes	DISPLAY LIST — Used Bytes	Total Bytes	Memory Step-Back To Be Added To PM Step-Back, ages
0	none	none	none	960	none	32	992	4
1	160	none	80	400	none	34	674	3
2	160	none	40	200	none	24	424	2
3	160	none	40	200	none	34	434	2
4	160	none	80	400	none	54	694	3
5	160	none	160	800	none	54	1174	5
6	160	none	320	1600	none	94	2174	9
7	160	none	640	3200	96	94	4190	17
8	160	16	1280	6400	80	176	8112	32

Notes: 1. RAMTOP is at extreme left of table. RAM decreases towards the right.
2. If 16 is added to the graphics mode number, then the conditional unused bytes are added to the screen memory block. The bytes formally used for the text window then become unused. Also the display list expands slightly.
3. The memory step-back in pages is calculated to the nearest, higher, whole page.

Table 1.

Graphics Mode	Number Of Bytes Of RAM Per Screen Mode Line
0	40
1	20
2	20
3	10
4	10
5	20
6	20
7	40
8	40

Example Of PM Positioning In Memory

Assume you wish to run PM in Graphics mode 7. You want to use all four players so all of the Player-Missile memory must be free and clear of the Screen-Display memory.

	Required Step-Back in Memory, pages
Graphics, 7, Screen-Display (Table 1)	17
PM Graphics, single-line resolution (requires 2K)	8
Total =	25 pages

However 25 pages is not on a 2K boundry:

6K = 24 pages
8K = 32 pages

Therefore you must step-back 32 pages for proper positioning of the PM system.

Table 2.

TOLL FREE Subscription Order Line
800-227-1617
In CA 800-772-3545
Please ask for Extension 401.

ATARI® Software for Personal Computers

A collection of 10 challenging programs created to provide a unique entertainment value — and 2 personal/business programs with broad functional value. In disk and/or cassette as indicated.

GAMES	• HELICOPTER BATTLE	Req. — 16K RAM/Cassette	$ 9.95
		— 16K RAM/Disk	14.95
	• HORSE RACING	Req. — 16K RAM/Cassette	9.95
		— 16K RAM/Disk	14.95
	• KENO	Req. — 8K RAM/Cassette	9.95
		— 16K RAM/Disk	14.95
	• LIGHTNING BOLTS and REACTION	Req. — 16K RAM/Cassette	9.95
		— 24K RAM/Disk	14.95
	• THE MAD MARBLE	Req. — 8K RAM/Cassette	9.95
		— 16K RAM/Disk	14.95
	• MUSIGAME (2 Games)	Req. — 16K RAM/Cassette	9.95
		— 24K RAM/Disk	14.95
	• SUPERMASTER	Req. — 8K RAM/Cassette	9.95
		— 16K RAM/Disk	14.95
	• TAG	Req. — 16K RAM/Cassette	9.95
		— 16K RAM/Disk	14.95
	• TRACTOR BEAM	Req. — 8K RAM/Cassette	9.95
		— 16K RAM/Disk	14.95
	• WAR AT SEA	Req. — 16K RAM/Cassette	14.95
		— 24K RAM/Disk	19.95
BUSINESS	• CCA Data Management System	Req. — 48K RAM/Disk	99.95
	• LETTER WRITER	Req. — 24K RAM/Disk	19.95

Mastercard & VISA Accepted

DIVISION OF CUSTOM ELECTRONICS, INC.
CE SOFTWARE

238 Exchange St., Chicopee, Massachusetts 01013
(413) 592-4761

• Dealer And Distributor Inquiries Invited
• Closed Mondays — Open Daily 'Til 5:30 — Fridays 'Til 8

www.commodore.ca

Invasion Orion: Can You Defeat The Klaatu and Your Computer?

Look your computer straight in the eye, load in the new **Invasion Orion** and suddenly you are the Fleet Admiral, the Commander-In-Chief. And only you can stop the alien forces: the robotic Klaatu who have just invaded your Stellar Union space.

You command as many as nine starships; each ship spends energy on moving, on shielding itself, on firing its three weapon systems—destructor beams, missiles, torpedoes. There are 30 different types of starships—all armed with such a fantastic array as to intimidate the Klaatu.

But it isn't as easy as it seems. Your ships have only a limited amount of energy and you must decide how to allocate that energy to destroy the enemy.

Will your ship's armor be enough to stop the enemy's torpedo? Or should you divert energy from your beam to your shield? Move in for the kill on your weakened opponent and risk a beam attack? Or outrun the enemy? With each turn you take, your energy is replenished. Can you defeat the Klaatu? Or is your computer smarter than you are?

Invasion Orion is an EPYX game. Like all EPYX games, you will never get bored playing. Not in your or your computer's lifetime. Every game is different and fresh. Choose from three levels of skill: beginner, intermediate, expert. Ten fully tested scenarios, from one-on-one starship combat to full scale battle. Two programs: the first uses your pre-created scenarios to play the game; the second lets you create your own scenarios and design your own ships. A game that is infinitely expandable.

Yet so very easy to learn. With any of the ten scenarios, the computer takes care of all the details; no complex rules to remember. The screen shows prompts for your battle orders. Just concentrate on your strategy for victory.

Complete with superb graphics (if you have either an Apple or an Atari, you can enjoy color and sound!) and with battle manual, game program, scenario creation program, data files for your computer.

Invasion Orion. Another bug-free, easy-loading lifetime computer game from EPYX. With the unique EPYX lifetime warranty: If anything happens to your cassette or disk at any time and for any reason, send it back with just $5.00 for shipping and handling and we will send you a brand new one.

(Of course, there is also 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. No questions asked.)

Visit your dealer now and pick up **Invasion Orion** in its good-looking, protective box with the best instruction book you've ever read. Now available on disk for the Apple II (48K RAM with Applesoft) and Radio Shack's TRS-80 (32K RAM). Or on cassette for the Atari (32K), TRS-80 (Level II, 16K), Apple (16K and Applesoft) and Pet (16K). **Only $24.95, disk or cassette.**

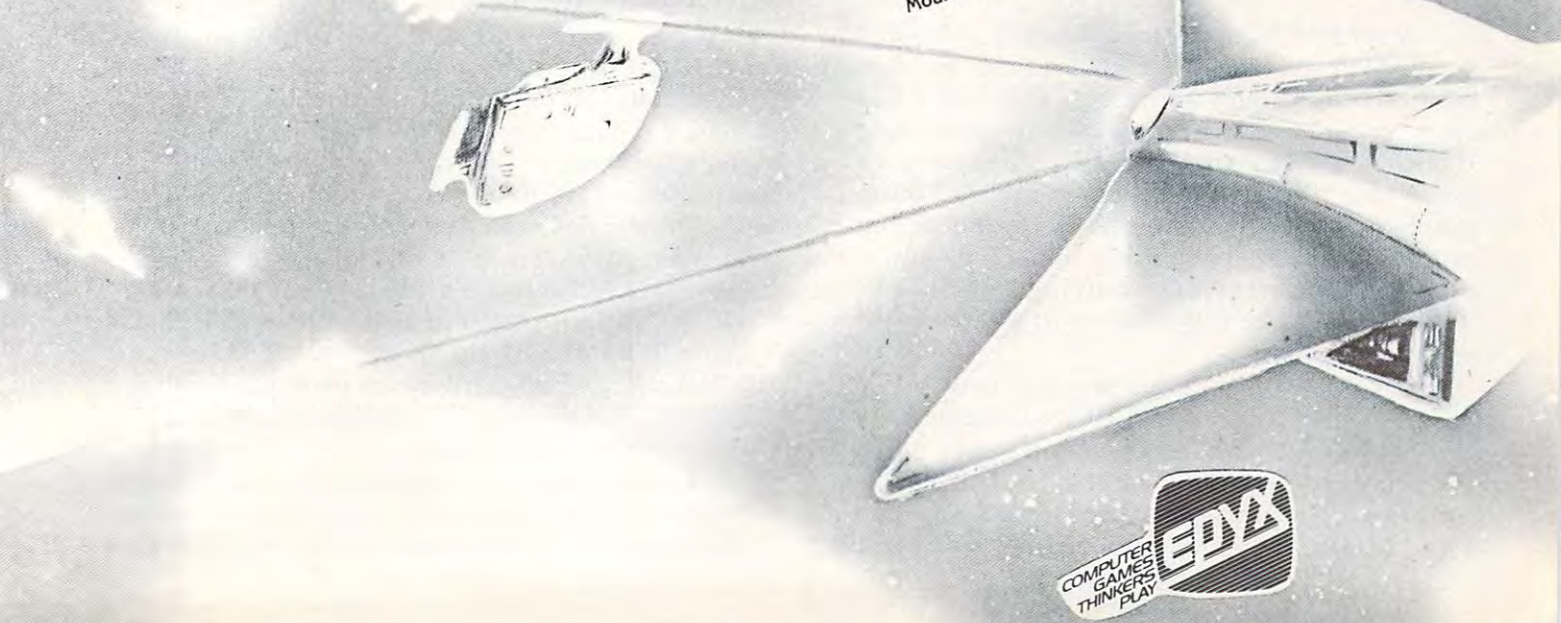
If your dealer is out of stock and you can't wait, order directly from Automated Simulations. $24.95 plus $2.00 for shipping and handling (and sales tax if you are in California).

We guarantee you will be delighted with **Invasion Orion.** (If it is not exactly what you want, return it to us for full refund. Again, no questions asked.)

Enclose your check. Or if you order by Visa or Master-Card, use our toll-free phones: In the United States: operator 861 **(800) 824-7888;** In California: operator 861 **(800) 852-7777;** In Hawaii and Alaska: operator 861 **(800) 824-7919.**

Order today. You and your computer deserve the fun.

AUTOMATED SIMULATIONS, INC.
Dept. 52, P.O. Box 4247
1988 Leghorn Street
Mountain View, CA 94040



www.commodore.ca

INSIGHT: Atari

Bill Wilkinson
Cupertino, CA

Editor's Note: We're quite pleased to announce a new column this month for Atari owners. INSIGHT: Atari, *written by Bill Wilkinson and other staff members of Optimized Systems Software, will bring you monthly programming insight and support.*

We feel you'll be quite pleased. — RCL

Hi. I'm Bill Wilkinson, and this is the premiere of what will be a regular feature in **COMPUTE!** magazine: a column dedicated to the *software* side of the Atari microcomputers. We may occasionally include little tricks to make better use of the hardware, but the intent is that this column will uncover the facts and foibles of Atari software.

This column will normally be written by some of the authors of Atari BASIC, Atari's Assembler-Editor, Atari's Disk File Manager, and BASIC A+ and OS/A+. We are not all experts in Atari hardware, but we know a lot about the software.

Addressable DATA or Who Needs String Arrays?

Perhaps the most frequent complaint made about Atari BASIC pertains to its lack of string arrays. In 10K bytes of ROM one can pack only so much program; long variable names and instant syntax checking take room; HP and DG have very successful BASICs that don't use string arrays; Atari-style strings are fast and flexible. All this doesn't mean much to you if you can't figure a way to convert that neat Applesoft program to Atari. There are many legitimate uses of string arrays, but the most common use is a kind of in-memory random access data file. Example: in an adventure game the various room descriptions are kept in elements of a string array. This is not the fullest exploitation of string arrays, since the data is static and the arrays merely provide a convenient method of addressing it.

Atari BASIC users, take heart! You have available to you an even more powerful and flexible method of randomly addressing static data. Did you ever notice that Atari BASIC supports the syntax "RESTORE line-number"? Did you ever notice that "line-number" can be either a constant number or (surprise) *any* arbitrary numeric expression? These two capabilities combine to allow some extremely powerful programming constructs in Atari BASIC. The following short program will serve to illustrate.

```
1000 REM a demonstration of addressable DATA
1010 REM allocate some variables
1020 DIM ROOM$(100),GO$(1),DIRECTION(4),
     DIRECTION$(4)
1030 LET DIRECTION$="NESW"
1100 REM the following variables are used as line
     numbers, etc.
1110 LOOKROOM=3000 : LOOP=2000 :
     DESCRIPTIONS=9000
1120 DESCRIPTIONSIZE=10
1900 REM variables are set up — initialize player
     status
1910 ROOM=2 : GOSUB LOOKROOM
2000 REM the main program loop
2010 PRINT "WHICH WAY"; : INPUT GO$
2020 DIRECTION=0
2030 FOR I=1 TO 4: IF GO$=DIRECTION$(I,I)
     THEN DIRECTION=I
2040 NEXT I
2050 IF NOT DIRECTION THEN GOTO LOOP
2060 GO=DIRECTION( DIRECTION )
2070 IF NOT GO THEN PRINT "CAN'T GO THAT
     WAY" : GOTO LOOP
2080 IF GO>1000 THEN GOSUB GO : GOTO LOOP
2090 ROOM =GO : GOSUB LOOKROOM
2100 GOTO LOOP
3000 REM subroutine to get and print details of a
     new room
3010 RESTORE DESCRIPTIONS + ROOM *
     DESCRIPTIONSIZE
3020 FOR I=1 TO 4 : READ TEMP : DIRECTION(I)
     =TEMP : NEXT I
3030 READ ROOM$ : PRINT "YOU ARE IN";
     ROOM$
3040 RETURN
8000 REM special routines for special actions
8010 PRINT "YOU MADE IT OUT! CONGRATU-
     LATIONS!" : END
9000 REM the room descriptions and connections
9010 DATA 3,5,0,0,A LARGE CAVERN
9020 DATA 0,4,5,3,A SMALL CAVERN
9030 DATA 0,2,1,0,A CURVING PASSAGEWAY
9040 DATA 0,8010,5,2,AN ANTECHAMBER
9050 DATA 2,4,0,1,A MAZE OF TUNNELS
```

Let us go through this program carefully and search out the tricks. Lines 1000-1030 are fairly straightforward; the variable names were purposefully chosen to demonstrate that Atari BASIC considers *all* characters in a name to be significant. Lines 1100-1120 initialize the variables which will be used for "address arithmetic" later in the program; "LOOP," for example, simply gives a name to the line number where all the action starts.

Line 1910 begins the start of the tricks: it GOSUBs to LOOKROOM. Notice how much more readable this is than simply coding GOSUB 3000, which tells you nothing of the purpose of the statement. Looking at routine LOOKROOM (lines 3000-3040), we note the usage of "RESTORE expression." As an example, assume that LOOKROOM is called with ROOM=2. Then line 3010 becomes equivalent to "RESTORE 9020." The subsequent READs then fill the array DESCRIPTION() with the numeric data of line 9020 and the string ROOM$ with the string, "A SMALL CAVERN." Finally, the user is prompted with a message ("YOU ARE IN A SMALL CAVERN,") and the subroutine exists.

Continuing our main program at lines 2000-2040, we simply ask the user for a direction (from the choices 'N', 'E', 'S', and 'W'). An invalid answer

www.commodore.ca

Announcing

ATARI software™

from the authors of An Invitation to Programming

exciting games and educational programs for kids, teenagers and adults featuring sound and color graphics.

available on guaranteed-to-load cassettes at fine computer dealers in your area or, write us directly for descriptive materials

PDI

Program Design, Inc.
Department CA
11 Idar Court
Greenwich, CT 06830

203-661-8799

SOFTWARE for the ATARI 800* and ATARI 400* from QUALITY SOFTWARE

STARBASE HYPERION™
By Don Ursem

Become absorbed in this intriguing, original space simulation of war in the far future. Use strategy to defend a front line Star Fortress against invasion forces of an alien empire. You create, deploy, and command a fleet of various classes of space ships, while managing limited resources including power generators, shields and probes. Real time responses are sometimes required to take advantage of special tactical opportunities. Use of color, sound, and special graphics add to the enjoyment of this program. At least 24K of RAM is required.

On Cassette — $19.95 On Diskette — $22.95

NAME THAT SONG
By Jerry White

Here is great entertainment for everyone! Two players listen while the Atari starts playing a tune. As soon as a player thinks he knows the name of the song, he presses his assigned key or joystick button. There are two ways to play. The first way requires you to type in the name of the song. Optionally, you can play multiple choice, where the computer asks you to select the title from four possibilities. The standard version requires 24K of RAM (32K on diskette) and has over 150 songs on it. You also get a 16K version that has more than 85 songs. The instructions explain how you can add songs to the program, if you wish. Written in BASIC.

On Cassette — $14.95 On Diskette — $17.95

QS FORTH
By James Albanese

Want to go beyond BASIC? The remarkably efficient FORTH programming language may be just for you. We have taken the popular fig-FORTH model from the FORTH Interest Group and expanded it for use with the Atari Personal Computer. Best of all we have written substantial documentation, packaged in a three ring binder, that includes a tutorial introduction to FORTH and numerous examples. QS FORTH is a disk based system that requires at least 24K of RAM and at least one disk drive. Five modules that may be loaded separately from disk are the fig-FORTH kernel, extensions to standard fig-FORTH, an on-screen editor, an I/O module that accesses Atari's operating system, and a FORTH assembler.

Diskette and Manual — $79.95 Manual Only — $39.95

FOR OUR COMPLETE LINE OF ATARI SOFTWARE
PLEASE WRITE FOR OUR CATALOG

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

ASK FOR QUALITY SOFTWARE products at your favorite computer store. If necessary you may order directly from us. MasterCard and Visa cardholders may place orders by calling us at (213) 344-6599. Or mail your check or bankcard number to the address above. California residents add 6% sales tax. *Shipping Charges:* Within North America orders must include $1.50 for shipping and handling. Outside North America the charge for airmail shipping and handling is $5.00. Pay in U.S. currency.

*Indicates trademarks of Atari.

www.commodore.ca

causes DIRECTION to be zero and the question to be asked again. Let us assume we are still in room two and also assume that the WHICH WAY? query was answered by "E." GO then becomes 4 (from DIRECTION(2)); and, since it is nonzero (line 2070) and less than 1000 (line 2080), the current ROOM becomes number 4 and we GOSUB LOOK-ROOM again.

The only things left to note about this program are what happens if GO is zero (e.g., if we had tried to go "N" from room 2) or greater than 1000 (if we try to go "E" from room 4)? The case of GO = 0 is easy: the program treats that as an illegal move, prints "CAN'T GO THAT WAY," and makes the player try again. For GO greater than 1000, another action unique to Atari BASIC hap-we GOSUB to the apparent room number contained in GO. In the particular example shown, the only GOSUB is to line 8010 which ends the "adventure," but this mechanism can be used to allow sophisticated checks on movement (e.g., you can only go from room 31 to room 33 if you have the Golden Fleece). The concept of addressable GOSUBs was heavily exploited, and we will try to cover those techniques in a future column.

Each of these columns will cover one or two programming topics and answer a few questions (presuming that you, the reader, will supply us with some questions). In this initial column, we would like to try to comment on some of the points raised in the "ASK THE READERS" column from **COMPUTE!** #14.

16K Memory

I. Regarding D. Gallagher's query about PRINT FRE(0) in his 48K machine.

When you plug the first (left, in an Atari 800) cartridge into an Atari, you "lose" the top 8K of the possible 48K of RAM. Thus your 48K does you no more good than 40K would. It can get worse: if Atari ever comes out with a dual cartridge product, you will lose the top 16K of your 48K. The reason: Atari's memory map simply doesn't leave any other place to put the cartridges, so Atari cleverly arranged the circuitry so that plugging in the cartridge disables any RAM at the same addresses. Does this mean that it is a waste to put 48K bytes of RAM into your Atari? Not at all! There are several products already available that use no cartridges at all (Visicalc, BASIC A+, Forth, etc.). In fact, look for Atari systems with 160K bytes of RAM, or more, in the near future. And by the way, it is not surprising to hear of the "foreign" memory board in the Atari: systems suppliers have been doing that in the minicomputer (DEC, HP, etc.) and S-100 (8080 and Z-80) markets for years! After all, if the dealer can give you more for less, why complain? Oh yes, for the curious, herewith the Atari memory map:

		FFFF
OS ROM (10K bytes)		**D800**
I/O hardware		**D000**
Reserved for future use		**C000**
48K of RAM	**Left cartridge**	**A000**
	Right cartridge	**8000**
		0000

II. Comments about the letter discussing RFI from an APPLE II.

Atari owners, stand up and be proud! Did you know that your machine is the only full-fledged computer that was able to pass the FCC's former (and very strict) RFI regulations? But thanks to TI, and some extensive lobbying with the FCC, the RFI rules are much relaxed and even the Apple II (with the help of some new shielding) can now pass the tests. But even so, the Atari has to be one of the quietest (in terms of RFI) machines ever produced. So while you owners are enjoying noise-free television, remember that the abysmally slow disk I/O speeds you also "enjoy" are part of Atari's solution to the RFI problem. That serial bus didn't just happen by accident: it was the result of some superb — but, alas, no longer necessary — engineering.

III. An answer to Tracy Principio about GR.X from assembly language.

Anyone contemplating writing in assembly language for the Atari is virtually required to purchase the *Hardware Manuals* (as did Tracy); but, even if you don't have a disk, the Atari DOS manuals and OS listings are *also* de rigueur. Any kind of I/O must go through CIO, the heart of the Atari OS, and graphics on the Atari are most easily done via I/O. Did you know that PLOT, DRAWTO, POSITION, FILL, and more are *not* in Atari Basic? They are actually routines in the I/O section of the OS ROM, and BASIC simply provides an interface to them. So, if you understand the I/O subsystem, you can do graphics in assembly language almost as easily as you can do them in BASIC. The whole subject of I/O and graphics from assembly language would make a beautiful *series* of columns (tell us if you'd like to see some), so we must "answer" Tracy's question by noting that GR.X is equivalent to:

OPEN #6,12,X-16,"S:"   if X is greater than 16
(full screen graphics)
or OPEN #6,12+16,X,"S:"   if X is less than 16
(mixed characters and graphics).

Note that the "12" is simply 8 + 4, read *and* write access, just as with a disk.

That's all for this month. We hope that by increasing your awareness of its capabilities we can convert you, too, into more informed and capable Atari users. ©

www.commodore.ca

Now for adults.

Textwizard™ transforms Atari into a powerfully serious word processor.

Textwizard™ is no kids game. It's a sophisticated and complete word processing program for the **Atari 800 computer.

Here is instant control over the creation, editing and formating of any writing. Insert words. Replace phrases. Delete sentences. Move paragraphs. Automatically repositions surrounding text. *Textwizard™ does it all with over 50 simple commands.

On your command, Textwizard™ will search out and correct mistakes throughout your document. And it protects you from common operating errors by warning you before you make them.

Textwizard™ will print out as many original "copies" as you need. In boldface, elongated or condensed lettering, upper or lowercase, with proportional spacing. And you can store your work on a diskette.

So, whether you write legal briefs, computer programs, or The Great American novel, Textwizard™ can make your work easier. And though it's no toy, it's fun to use.

At $99.95, you don't have to write a best-seller to afford Textwizard.™ Textwizard™ is another of the many creative products from Datasoft.

Datasoft Inc.™
COMPUTER SOFTWARE

Software for people who aren't easy to please.

19519 Business Center Drive / Northridge, California 91324 / (213) 701-5161

Check your local software dealer or
Send check or money order with $2.00 postage/handling. California residents, add 6% sales tax.

*Textwizard will perform on a 32K system with one or more disc drives. It is compatible with the Atari® 825, Centronics® 737 and Epson® MX-80 printers.
** Atari is a registered trademark of Atari Computers Inc.

www.commodore.ca

The ATARI 825: An Assembler Interface

John Elliott
New York, NY

The ATARI 825 printer must be one of the most versatile in its price range. It offers three distinct character sets:

- Monospaced characters at 10 characters per inch (10 CPI)
- Monospaced condensed characters at 16.7 CPI
- Porportionally spaced characters at 14 CPI

For each of these character sets, you can select either the normal printing mode or the elongated character mode, which prints characters at twice the normal width. There are many other useful features, too. These include true underlining, superscripting and subscripting, reverse as well as forward line feed, character backspacing, and so on. This is a truly remarkable printer!

What is more, all of these features are completely under software control. That is, you, the programmer, control exactly how your data is to be printed, without any need for operator intervention. This is made possible by the fact that the printer has a built-in central processing unit (CPU), which can recognize and interpret printer control codes.

When the printer is powered on, the CPU automatically selects the 10 CPI monospaced character set. Any data characters sent to the printer will be printed using this default character set, which will stay in effect until the printer receives control codes which specify an alternate character set. Subsequent data will be printed in the new character set, until yet other control codes are received, or until the printer is power cycled (turned off, then back on).

The user manual which is supplied with the printer is also deserving of praise. It is comprehensive and very clearly written, with all kinds of tables, charts and diagrams, to help you understand the text, and to get the most out of the printer. It even includes a BASIC program for right-justifying text lines, another very very useful capability of this printer. All in all, I have nothing but praise for both the printer and the manual.

Of course, if everything in the garden were rosy, then there would be no need for this article. There is just one slight hitch. Assembly language programmers have been somewhat ignored.

Neither the 825 manual, nor the ASSEMBLER/EDITOR manual describes how to make use of the special features of the printer from the ASSEMBLER/EDITOR, or from an assembly language routine. So, if you want to use a character set other than the default 10 CPI set, to print an assembly listing, for example, then you've got a problem. And, of course, just about the first thing I wanted to use the printer for was to print an assembly listing, using the neat, paper-saving condensed character set. I searched through the manuals for a clue as to how I might do this, but found none.

Necessity being the mother of invention, I set to work designing an assembly routine which would select a character set of my choice. The short program listed here is the result. It is a very simple routine, and, as coded, will select the 16.7 CPI condensed character set, which I use for printing my assembly listings.

I will now describe the program logic in more detail, and give instructions for executing it. Finally, I will describe how you can modify the program to select any print mode of your choice. I assume that the reader is familiar with the ASSEMBLER/EDITOR cartridge, and has access to an ATARI 825 printer user manual.

The program opens the printer (device code P:) using input/output control block #6 (IOCB #6). This establishes a link between our program and the central input/output (CIO) subsystem of the ATARI operating system (OS). The ATARI OS is the program in the 10K ROM module in your ATARI console. This program contains many routines written specifically for communicating with the input/output devices, such as printers, disks, cassettes, etc. These routines are referred to collectively as CIO routines, and they provide the application program with a means of accessing the peripheral devices in a standard, device-independent manner. There is a single entry point to the CIO routines, and the IOCB is the vehicle of communication between CIO and the application program.

Having established the link with CIO, our program then transmits a string of control codes to the printer, through CIO, to select the desired character set. In the listing shown here, the codes are those required for selecting the condensed character set. The program then closes the file, thereby breaking the link with CIO and freeing the IOCB for other I/O. It then issues the BRK instruction which will return control to the DEBUGGER.

And that's all there is to it.

To use the routine, first assemble it into RAM. Then, using the DEBUG program in the ASSEMBLER/EDITOR cartridge, execute the

www.commodore.ca

Drawing Tablet

VersaWriter operates on a simple principle, but produces graphics which match or exceed those of other digitizers. Its rugged yet precision construction makes it easy to use and trouble free. Operation is mastered in minutes. It plugs directly into your ATARI personal computer.

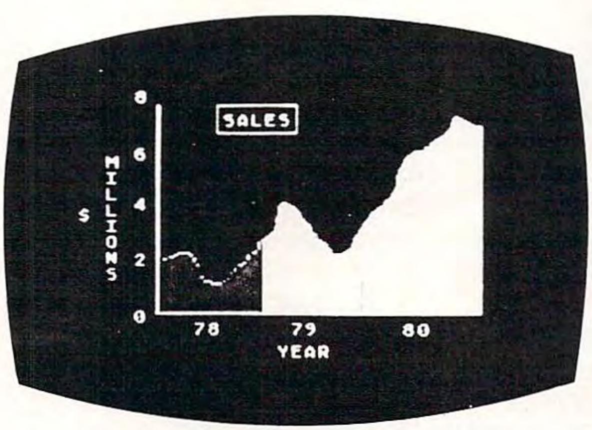
VersaWriter

FOR ATARI PERSONAL COMPUTERS

Suggested Price $299.00

Graphics Software

Easily the most capable, complete, and usable graphics software for ATARI personal computers available. Designed for hobbyists, but engineers, artists, doctors, and educators are also finding exciting ways to expand their computer's value with VersaWriter.



UNIQUE OFFER

Send us YOUR disk and $1. We will promptly return the disk with a slide package of 10 color pictures drawn with VersaWriter.

☐ Enclosed is $1 and my disk. Send me the slide package.

☐ ATARI ☐ APPLE II

☐ Send more information including VersaWriter dealers in my area.

DEALER INQUIRIES INVITED.

NAME

ADDRESS

CITY STATE ZIP

Versa Computing, Inc. • 887 Conestoga Circle • Newbury Park, CA 91320 • (805) 498-1956

www.commodore.ca

program by typing G600 (assuming you assembled it into page 6). The program will transmit the control codes, as described above. Don't forget to have the printer properly connected and powered on.

After transmitting the control codes, the program gives control back to the DEBUG program. You will know that this has happened when the DEBUG prompt appears on the screen. You can then go back to the EDITOR (by typing X), and LIST, PRINT, or ASM output to the printer, using the character set selected by the assembler routine. The control codes will stay in effect until the printer is power cycled, or until you transmit some other control codes to the printer.

Changing the program to select a print mode of your choice is quite straightforward. Consult Table 2 in the 825 user manual. This table lists the printer control codes. Note the codes needed to select the print mode you are interested in. Then change the constant labelled PCODE in listing to contain these codes. Reassemble, and you're ready to go. Just execute the program to select the mode of your choice.

Bear in mind that the program was specifically designed to execute in conjunction with the DEBUGGER. It is important to remember this, as the BRK instruction is used to terminate the program. In effect, this instruction relinquishes control to the DEBUGGER. If you want to use the program in some other environment, then you should change the exit logic to conform to the constraints of that environment.

However you decide to use the routine, I hope that you find it useful. Good printing, and good luck!

```
                0100 ;+++++++++++++++++++++++++++
                0110 ;+ ASSEMBLER SUBROUTINE TO +
                0120 ;+    INTERFACE  TO THE     +
                0130 ;+    ATARI 825 PRINTER     +
                0140 ;+ J.  ELLIOTT.   APRIL/81 +
                0150 ;+++++++++++++++++++++++++++
0000            0160 IOCB    *=   $0340      ;I/O CONTROL BLOCKS
0340            0170         *=   *+2
0342            0180 ICCOM   *=   *+1        ;COMMAND CODE
0343            0190         *=   *+1
0344            0200 ICBAL   *=   *+1        ;BUFFER ADDRESS LSB
0345            0210 ICBAH   *=   *+1        ;BUFFER ADDRESS MSB
0346            0220         *=   *+2
0348            0230 ICBLL   *=   *+1        ;BUFFER LENGTH LSB
0349            0240 ICBLH   *=   *+1        ;BUFFER LENGTH MSB
034A            0250 ICAX1   *=   *+1        ;AUX CONTROL BYTE 1
034B            0260 ICAX2   *=   *+1        ;AUX CONTROL BYTE 2
0008            0270 OPNOT   =    $08        ;OPEN FOR OUTPUT
0003            0280 OPEN    =    $03        ;COMMAND CODE - OPEN
000C            0290 CLOSE   =    $0C        ;COMMAND CODE - CLOSE
0009            0300 PUTREC  =    $09        ;PUT RECORD
009B            0310 EOL     =    $9B        ;END OF LINE (CR)
001B            0320 ESC     =    $1B        ;ESC CONTROL CODE
0014            0330 COND    =    $14        ;CONDENSED C.C.
0000            0340 NULL    =    $00        ;NULL CHARACTER
0060            0350 IOCB6   =    $60        ;IOCB #6 INDEX
0028            0360 BUFSZ   =    40         ;MAX RECORD SIZE
E456            0370 CIOV    =    $E456      ;CIO ENTRY VECTOR
034C            0380         *=   $0600      ;PROGRAM ORIGIN
0600 A2FF       0390         LDX  #$FF
0602 9A         0400         TXS             ;INIT THE STACK POINTER
0603 A260       0410         LDX  #IOCB6     ;OPEN P: USING #6
0605 A903       0420         LDA  #OPEN
0607 9D4203     0430         STA  ICCOM,X    ;SETUP OPEN COMMAND
060A A948       0440         LDA  #FNAME&$FF ;POINT TO THE NAME
060C 9D4403     0450         STA  ICBAL,X
060F A906       0460         LDA  #FNAME/256
0611 9D4503     0470         STA  ICBAH,X
0614 A908       0480         LDA  #OPNOT     ;OPEN FOR OUTPUT
```

www.commodore.ca

Auto-dial / Auto-answer

for Atari 400/800*

Our MICROCONNECTION™ for the Atari* comes in two versions — the RS232 MICROCONNECTION™, for use with the 850* Interface Module (Autodial/Autoanswer optional), and the Atari* buss-decoding MICROCONNECTION™, which plugs directly into the computer's data buss (Autodial optional). The MICROCONNECTION™ for the Atari* — the obvious answer.

Prices start at $199.50.

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

the microperipheral corporation

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

* Indicates trademarks of Atari, Inc.

ATARI 800/400 SOFTWARE BY MED SYSTEMS!

SOFTWARE AUTHORS!

Join the company of best-selling authors at Med Systems. We have an **established** market spanning the free world and royalties second to none. We seek **excellent** games, utilities and applications packages. Only the best are accepted! If you have authored software you feel is publishable, submit it to Med Systems, Software Review Section.

KNOSSOS You wander a gigantic, 3-D perspective cave, seeking the only door out. Somewhere, the minotaur seeks you for a grizzly meal. The cave is graphically represented **as though you are actually there!** Extensive graphics, sound effects.

Atari 400/800 16K BASIC cassette	$14.95
Atari 400/800 32K BASIC disk	$19.95

Scott Adams Adventures 1-9 for 24K Ataris. What can we say? These are some of the best!

Atari 400/800 24K cassette	$18.95 each

(The higher numbers are the harder adventures)

COMING SOON FOR THE ATARI: Rat's Revenge, Deathmaze 5000, Labyrinth, Asylum

ATARI SOFTWARE: Star Raiders $34.95, Chess $34.95, Basketball $24.95, Asteroids $34.95, Missle Command $34.95, Space Invaders $16.95, Super Breakout $34.95, Joystick Pair $16.00.

CALL FOR OUR INCREDIBLE HARDWARE PRICES!

MED SYSTEMS has been publishing and distributing software worldwide since 1979. We publish only the best! We ship within 5 days, whenever possible, usually within 2! We **don't** wait for "checks to clear." We know how hard it is to wait for that perfect program. **We even guarantee satisfaction!** If you don't like our software, return it within 14 days for a prompt, cheerful refund. If you have a problem, call us 10-6 EST. We are here to serve you.

MED SYSTEMS
Box 2674-C1, Chapel Hill, NC 27514
(919) 933-1990

www.commodore.ca

```
0616 9D4A03 0490          STA  ICAX1,X
0619 A900   0500          LDA  #$00       ;CLEAR AUX2
061B 9D4B03 0510          STA  ICAX2,X
061E 2056E4 0520          JSR  CIOV       ;DO THE OPEN
0621 3024   0530          BMI  RETURN     ;EXIT IF NO GOOD
0623 A94B   0540          LDA  #PCODE&$FF ;SETUP THE...
0625 9D4403 0550          STA  ICBAL,X
0628 A906   0560          LDA  #PCODE/256 ;...BUFFER ADDRESS
062A 9D4503 0570          STA  ICBAH,X
062D A928   0580          LDA  #BUFSZ&$FF ;SETUP MAX...
062F 9D4803 0590          STA  ICBLL,X
0632 A900   0600          LDA  #BUFSZ/256 ;...BUFFER SIZE
0634 9D4903 0610          STA  ICBLH,X
0637 A909   0620          LDA  #PUTREC    ;SETUP THE PUT...
0639 9D4203 0630          STA  ICCOM,X    ;...COMMAND CODE
063C 2056E4 0640          JSR  CIOV
063F A90C   0650          LDA  #CLOSE     ;CLOSE THE FILE
0641 9D4203 0660          STA  ICCOM,X
0644 2056E4 0670          JSR  CIOV
0647        0680 RETURN = *               ;RETURN TO DEBUGGER
0647 00     0690          BRK
            0700 ;
0648 50     0710 PNAME  .BYTE "P:",EOL
0649 3A
064A 9B
064B 1B     0720 PCODE  .BYTE ESC,COND,NULL,EOL
064C 14
064D 00
064E 9B
064F        0730        .END
```

©

ATARI™ OWNERS DEALERS PROGRAMMERS

MASTER MEMORY MAP — Lists almost every memory location that you might need with examples of what to poke or peek and what results necessary for beginning and advanced Atari owners. ***$5.95.***

TRICKY TUTORIAL TAPES — Do your programs all look alike? This series of self teaching tutorials will first show you what your machine can do, then take you step by step through the code itself. All of these "tricks" can easily be used by basic or advanced programmers. ***Excellent as Dealer Demos.***

TTT#1 — DISPLAY LISTS: Break up your screen into 3, 5, even 10 different modes of graphics and text at the same time.

TTT#2 — HORIZONTAL/VERTICAL SCROLLING: Move the information on the screen up, down, or sideways.

TTT#3 — PAGE FLIPPING: Instantly display a new screen of graphics or text by the press of a button.

TTT#4 — BASICS OF ANIMATION: For games or business presentations, animated figures or graphs are very impressive. Similar to techniques used in star raiders.TM

TTT#5 — LIGHT PEN PROGRAMS: Some single programs for use with the new light pen.

Each program takes a few hours to completely learn, includes full documentation, and costs:

$14.95** Tape or Disk! Any 3 for **$39.95

ORDER TODAY OR SEND FOR OUR COMPLETE CATALOG

SANTA CRUZ SOFTWARE

5425 Jigger Drive, Soquel, CA 95073

ATTENTION ATARI* Programmers, Authors ...

Let us Evaluate,
Develop, and Market
Your PROGRAMS.

WHY ? ? ?

Because, we Market
only to ATARI* Users.
Our staff has the
Programming, Marketing,
And Sales Experience
to turn YOUR Program
into a CERTIFIED Money maker.
If you are interested,
We are INTERESTED.

For further information WRITE:
CYGNUS MICRO SYSTEMS
P.O. Box 1203
Claremont, Ca 91711
Attn: Director Software Marketing

*Atari is a registered trademark of ATARI, Inc.

www.commodore.ca

Using The Color And Locate Instruction To Program Pong Type Games

Michael A. Greenspan

New Atari Owners may be confused (as I was) about the COLOR and SETCOLOR instructions. These two commands, and the LOCATE instruction, form the basis of the following PONG type game.

In Graphics 3, there are four color registers labeled 0, 1, 2, and 3, which are accessed by the instruction COLOR X, where X is the number of the register desired. (COLOR 4 is the same as COLOR 0; COLOR 5 is the same as COLOR 1, etc.) While COLOR determines the register used, SETCOLOR enables you to determine which of the 128 colors are used by your chosen register to draw points on the screen. Thus, since the SETCOLOR instructions are identical, the following commands will each put a dark gold on the screen at location 1,1:

10 GR 3: COLOR 1: SETCOLOR 0, 1, 2* : PLOT 1,1
10 GR 3: COLOR 2: SETCOLOR 0, 1, 2* : PLOT 1,1

Each color register has a different default color that determines the color of the points plotted in that register if no SETCOLOR 0, X, X instruction is given. Therefore plotting points in different color registers will produce different colors in the absence of SETCOLOR instructions, and identical colors if identical SETCOLOR instructions are used.

In the program below, a ball will move from left to right and a joystick is used to maneuver a paddle on the far right to intercept the ball. The paddle is plotted in color register 1, and the ball in color register 2. In order to move the ball, it is replotted in color register 4, whose default color is the same as the background color (and thus is invisible), and then replotted on the adjacent square in color register 2.

The LOCATE instruction determines if there is a hit. X and Y are the X and Y coordinates of the ball. LOCATE X + 1, Y, X tells the computer to LOCATE the point to the right of the ball and to store the *color register* of that point in Z. Since the paddle is plotted in color register 1, Z = 1 means that the ball hit the paddle.

Once you understand the use of COLOR and LOCATE to move the ball and effect a hit, it is a relatively simple matter to add boundaries, 2 or more paddles, sound, etc., etc., etc.. (Of course the same result can be accomplished by Player Missile Graphics, but that's the subject of another article.)

In the program below, A and B are the X and Y coordinates of the paddle. X and Y are the X and Y coordinates of the ball. C relates to random changes in the color of the paddle. S relates to the speed with which the ball moves.

*The SETCOLOR command instructs the computer to set the color of the points on the screen (that's the function of the 0) to color 1 (that's gold) brightness 2. A two for the first number will change the Text Window to that color. A four will change the background.

```
1 REM BY MIKE GREENSPAN
2 REM 15604 SYCAMORE LANE
3 REM ROCKVILLE, MD  20853
4 REM QUESTIONS CALL O-202 857 0350
5 REM OR H-301 924 2210
10 S=51:GRAPHICS 3
20 A=35:B=10:X=0:Y=INT(RND(0)*19)+1:C=IN
T(RND(0)*15)+1
25 REM PLOT THE PADDLE
30 COLOR 1:SETCOLOR 0,C,8:PLOT A,B:PLOT
A,B+1
35 REM MOVE THE PADDLE UP?
40 IF STICK(0)=14 THEN COLOR 4:PLOT A,B:
PLOT A,B+1:B=B-1:IF B<0 THEN B=0
50 IF STICK(0)=14 THEN GOTO 30
55 REM MOVE THE PADDLE DOWN?
60 IF STICK(0)=13 THEN COLOR 4:PLOT A,B:
PLOT A,B+1:B=B+1:IF B>19 THEN B=19
70 IF STICK(0)=13 THEN GOTO 30
75 REM  PLOT THE BALL AND HOLD IT AT  TH
AT LOCATION WHILE THE COMPUTER COUNTS FR
OM 1 TO S
80 COLOR 2:PLOT X,Y:FOR D=1 TO S:NEXT D
85 REM CHECK IF THE BALL HIT THE PADDLE
90 LOCATE X+1,Y,Z
95 REM MOVE BALL TO THE RIGHT IF IT HAS
NOT REACHED THE END OF THE ROW
100 IF Z<>1 THEN IF X<=35 THEN COLOR 4:P
LOT X,Y:X=X+1:GOTO 30
105 REM IT'S A MISS
110 IF Z<>1 THEN IF X>35 THEN MISS=MISS+
1:? "HITS-";HIT;"  MISSES-";MISS:COLOR 4
:FOR B=0 TO 19:PLOT 35,B:PLOT 36,B
120 IF Z<>1 THEN NEXT B:S=S+10:GOTO 20
125 REM IT'S A HIT
130 HIT=HIT+1:? "HITS-";HIT;"  MISSES-";
MISS:S=S-10:COLOR 4:FOR B=0 TO 19:PLOT 3
5,B:PLOT 34,B:NEXT B:GOTO 20
```

©

www.commodore.ca

Atari BASIC String Sort

Jerry White

Putting data in alphabetical order makes finding things much easier. Having numbered items in numeric order is often essential.

This Atari BASIC adaptation of the Shell-Metzner sort algorithm does both. Numbered items will be placed before alphabetic data if both are used.

This demo program can easily be modified to suit your needs. For demonstration purposes, I set up this program to handle up to 100 records. Each record or item may contain up to 30 characters.

Suppose 30 characters is just short of your requirements. Let's assume you need a 35 character record. All you would have to do is change all references of the number 30 to 35, and change all 29's to 34. You would also have to change both places where you find A$(3000) in line 130 to A$(3500). This figure is the record size multiplied by 100. If you have more than 100 records, just multiply your record size by the maximum number of records you may possibly need.

You do not have to enter all your records at one time. You also do not have to type the entire length of the dimensioned record size. Just to get a feel for the program, type in a few names or numbers. As you enter each item, REC will store the current record number. To end your data entry, just enter a null record which means just press the return key.

The data will be sorted then displayed on the screen. You can pause the display by holding the CTRL key and typing 1. Repeat this procedure to continue.

After the sorted data is displayed, you may continue by pressing the OPTION button. Again, a null entry will begin the sort phase.

Add some options of your own, such as saving the file (A$) on cassette or diskette, and you've got yourself a mini-database. You can find endless uses for manipulating data. Sorting is one of the most effective ways of making data easier to read and handle. Remember, your Atari computer can do a great deal more than entertain you.

```
100 REM ATARI BASIC STRING SORT
110 REM TUTORIAL BY JERRY WHITE
120 REM *** SETUP ***
130 DIM A$(3000),B$(30),C$(30):A$(1)=" "
:A$(3000)=" ":A$(2)=A$:B$="":C$=""
140 GRAPHICS 0:SETCOLOR 2,0,0:POKE 82,2:
GOTO 320
150 REM *** SORT A$ ***
160 T=INT(T/3)+1:FOR L1=1 TO REC-T:FOR L
2=L1 TO 1 STEP -T
170 IF A$(L2*30-29,L2*30)<=A$((L2+T)*30-
29,(L2+T)*30) THEN 210
180 C$=A$(L2*30-29,L2*30):A$(L2*30-29,L2
*30)=A$((L2+T)*30-29,(L2+T)*30)
190 A$((L2+T)*30-29,(L2+T)*30)=C$
200 NEXT L2
210 SOUND 0,REC+10-L1,10,2:NEXT L1
220 IF T>1 THEN 160
230 REM *** DISPLAY SORTED DATA ***
240 SOUND 0,0,0,0:? CHR$(125):? ,"*** SO
RTED DATA ***"
250 FOR ME=1 TO REC:? A$(ME*30-29,ME*30)
:NEXT ME
260 REM *** CONTINUE OPTION ***
270 ? :? "         PRESS OPTION TO ADD DAT
A":? "       PRESS SELECT TO END DEMO";
280 IF PEEK(53279)=3 THEN 330
290 IF PEEK(53279)=5 THEN GRAPHICS 0:END
300 GOTO 280
310 REM *** DATA ENTRY ***
320 ? :? "ENTER UP TO 100 RECORDS TO BE
SORTED:"
330 REC=REC+1:? :? "ENTER RECORD ";REC:B
$="":INPUT B$:LB=LEN(B$)
340 IF LB=0 THEN REC=REC-1:T=REC:? ,"***
 S O R T I N G ***":GOTO 160
350 A$(REC*30-29,REC*30-29+LB)=B$
360 GOTO 330
```

©

educational software for the ATARI* educational software for the APPLE II+** educational software for the ATARI* educational software for the APPLE II+** educational software for the ATARI* educational software for the APPLE II+** educational software for the ATARI* educational software for the APPLE II+** educational software for the ATARI* educational

HIDDEN WORDS

A word-search program for the ATARI*. Move your character over the letters that spell your word. Words may be read from left, right, up, down, or diagonally.

- For one or two players
- Four levels of play
- Comes with word base of the most frequently misspelled words.
- Requires JOYSTICKS

24K BASIC, Grades 1-8+ ... $17.50

SPATIAL RELATIONS

Designed to help the child learn words that indicate size and placement. Units cover: big/small, tall/short, right/left, high/low, over/under, in/out. Requires one joystick. Minimal reading. SIDE TWO REQUIRES NO READING - Use with Type-'N-Talk™ speech synthesizer.
16K BASIC, Preschool-1 .... $17.50

For the ATARI*

Type-N'-Talk™*** speech synthesizer (VOTRAX).
Let your computer talk with one of the easiest to use speech synthesizers on the market.
Requires RS-232 interface. $375.00

T.H.E.S.I.S.
P.O. Box 147 • Garden City, MI 48135
or call (313) 595-4722 for C.O.D.
Some T.H.E.S.I.S. software is now available for the APPLE II+
DEALER INQUIRIES WELCOME
Please add: $1.50 shipping/handling, $1.50 for C.O.D., Mich. res. add 4% sales tax.
WRITE FOR FREE CALALOG

*ATARI is trademark of Atari, Inc. **APPLE is trademark of Apple Computer, Inc. ***Type-'N-Talk is trademark of Votrax.

www.commodore.ca

LETTER PERFECT T.M. LJK

WORD PROCESSING FOR THE *ATARI — 800 T.M.

MAIN - MENU

CURRENT DRIVE NUMBER #1

→ Editor ←
Change Drive #
Load
Save
Merge
Screen Format
Printer
Lock
Unlock
Delete
Format Disk
Data Base Merge
Quit

Press '<' or '>' to move cursor
Press (Return) for selection

USE: EPSON MX-80

and ATARI -825

PRINTERS

EASY TO USE: LETTER PERFECT is a character orientated word processor with the user in mind. The program (machine language) is very fast. It is a menu driven program that is very easy to operate. The program is a single load program and can work with one or more disk drives. It requires a minimum of 16K of memory and a single disk drive. With the Artari 825 printer you can print text with right hand justification. You may also use different type fonts (10 and 17 character per inch) within the body of the text itself. Boldface is printed as expanded print font. Underlining can be done as well as sending Escape characters within the body of the letter itself. All the formats are a default but you can change them all to desired values if you wish. Right Margin, left margin, top of form, line spacing, etc. are easily changed. Data Base Merge works with the sister program LETTER PERFECT — DATA BASE MANAGER. User may use this program to create mailing lists, and completely develop your own data base for your personal needs. All text packed before storage to diskette for greater storage capacity. Large Buffer allows you to pick up and move up to one full page of screen text and move it to any location in the text. Merge more than one file together for easy editing. Screen Format allows you to see on the video screen exactly how the text will appear on the printer. Automatic page numbering, headers and footers are easily accomplished. This program is easy to use because of its meaningful and easily mastered commands. Fully documented with a users manual that explains in simple language 'how to' completely use the program.

All this and more, for $149.95.

Features:

FULL CURSOR CONTROL
Home Cursor
Scroll Page Forward
Scroll Page Backward
Pause Scroll
Scroll Line at Time
Scrolling Speed Control
Move Cursor Down
Beginning of Text

MULTIFUNCTION FORMAT LINE
Standard Formats a Default
Formats Easily Changed
Right Justification
Left Margin
Page Width
Line Spacing
Lines Per Page
Form Stop
Set Page#
Top Margin
Bottom Margin

Delete a Character
Insert a Character
Delete a Line

Insert a Line
Headers and Footers
Shift Lock and Release
Global and Local Search and Replacement
Underlining and Boldface
Automatic Centering
Horizontal Tabs
Special Print Characters
Split Catalog
Page Numbering up to 65535
Prints up to 255 Copies of Single Text File
Non Printing Text Commenting

FUNCTIONS
Delete All Text
Delete All After Cursor
Delete All Before Cursor
Delete Next Block
Delete Buffer
Move Next Block to Buffer
Add Next Block to Buffer
Insert Block From Buffer
Merge Text Files

apple®

This program also available on the Apple in 40/80 Video (Super'R' Term, Smarterm, Videx, Bit-3). You may use any printer type. The Hays Micromodem II can be used to send files. Can be Reconfigured at any time to use different printer, 80 column board, or standard 40 column video. Much, Much, More!

DEALER INQUIRIES INVITED

LJK™ ENTERPRISES Inc.
COMPUTER BASED SOFTWARE

FREE CONTROL PAGE

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

*Apple T.M. of Apple Computer Inc., — *T.M. Atari Computer Corporation.

www.commodore.ca

Dynamic Player Animation With Atari

Alan Watson

This article describes a simple technique to create dynamic players with ATARI's Player/Missile Graphics. Articles have appeared here in **COMPUTE!**, as well as *ATARI CONNECTION*, which describe how to set up P/M Graphics, bit-map players, and move them using joysticks. If you would like your airplane to face in the direction it is moving, or your players to shake their heads or move their feet, this article may help you.

The central idea is to use a string or substring to hold the bit-map description for each view or position you want your player to assume. Then, using the VAL function, poke different strings or substrings to make your player change.

As an example program, we will create a figure who "marches" raising first one foot, then the other. First, we draw and bit-map the different positions involved in marching. See Figure 1.

We will put our bit-map descriptions in DATA statements to make them easy to find should we want to make changes in any of the player positions later. It is important to use three digits for each row in each bit-map. For example, in our DATA statements, 7 will be entered 007, 66 will be entered 066, and so on. This makes it easy to find each element of the string or substring when we get ready to poke the description into memory.

Now let's get to the program itself:

LINES 100 – 150. Here strings are dimensioned. Our data is read (in groups of three digits) and put into the string P$. P$ now contains the descriptions for all three positions.

LINES 200 – 290. These lines set up P/M Graphics.
LINE 220 is our player's starting position.
LINE 230 enables double line resolution.
LINES 240 and 250 set player/missile address.
LINE 260 enables P/M Graphics.
LINE 270 clears out player memory area.
LINE 280 sets our player color to gold.
LINE 290 sets player's horizontal position.

LINES 300 – 390. These lines establish a view or position pointer to indicate which position is to be drawn. Since all our descriptions are in the string P$, we use a substring V$ to extract the position description of each "march" step as needed. Sound is added in line 385 so we can hear the steps as they are made.

LINES 400 – 470. This is the motion routine.
LINE 410 reads the joystick.
LINES 420, 430, 440, 450, and 460 check for no joystick movement or movement left, right, down, or up respectively.
LINE 470 sends the program back to the pointer to begin again with the next player position.

LINES 500 – 530. These DATA statements hold the bit-map information. Each of lines 510, 520, 530 contains a different position.

After making your way through this example, you will no doubt have ideas for expanding it or for figures of your own design. You may want to add positions which have our marching figure actually turn and march facing left or right. To do this, set up a pointer which is changed as the joystick position is read. Another idea is to use separate strings or substrings for the head and body. By concatenating the strings, you can make the player shake his head while marching or while not moving his feet at all.

If, like me, you have grown tired of moving static figures around the screen, these ideas will help you. Now your spaceships can explode in a cloud when they are hit, your figures can dance and change expressions, and your animation efforts can be more rewarding.

References:

Crawford, Chris, "Player-Missile Graphics with the Atari Personal Computer System", p. 66, **COMPUTE!**, Issue 8, January 1981.

"Player-Missile Graphics", p. 10, *The Atari Connection*, Vol. 1, No. 1, Spring 1981.

Atari 400/800 Basic Reference Manual, Atari, Inc., copyright 1980.

Bit-maps For Player Positions

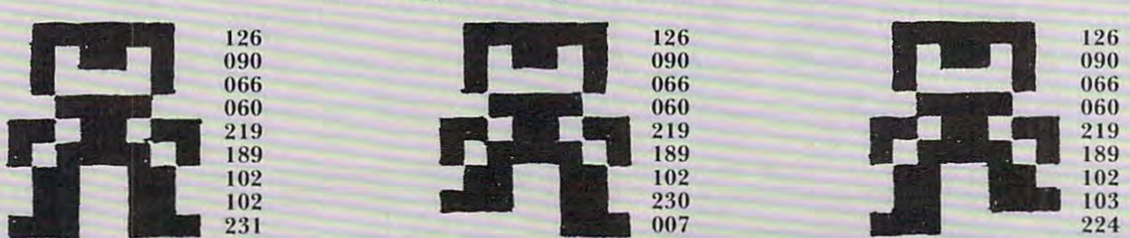


Figure 1.

www.commodore.ca

DEDICATED TO ATARI®

PREMIUM SOFTWARE FOR YOUR ATARI

ARCADE GAMES 24K disk; 16K cassette; Joystick required

Add these HIGH RESOLUTION, REAL-TIME, ANIMATED games to your software arsenal. Get FAST ACTION and FULL SOUND GRAPHICS that take advantage of the unique features of your ATARI. Enjoy challenge that requires strategy and skill.

SPACE CHASE Fly against intelligent invader clones. Arm Yourself with Nuclear Defense Charges and play with or without Defense Shields. Enjoy this action-packed multicolor space odyssey. Displays top score, number of planets saved and number of galaxies conquered. $14.95 cassette; $19.95 disk

TIMEBOMB Meet the challenge of this fast moving animated race against time, enemy aircraft and enemy bombs as you attempt to disarm timebombs set to explode ammunition depots. Avoid aircraft of varying sizes and speeds — and their bombs. Choose one of ten Day or Night Missions. Use from one to four Joysticks. Any number can play; top players listed on scoreboard. $14.95 cassette; $19.95 disk

DATA MANAGEMENT

FILE-IT With this **startup database system** you can file and manage personal information and data. Use this database system to create, sort, store, and manipulate information such as appointment calendars, address or telephone data, credit or charge card records, stock investments, medical or prescription information, hobby, coupon or other types of collection informaton...and more. With a printer you get 1 or 2 across mailing labels, disk jacket inventory covers and neatly written copy of all your data files. Comes with well documented instruction manual explaining basics of computer filing. Fast and easy to use. Holds 100+ records in 24K and over 300 in 40K. Requires minimum of 24K and one disk drive. Pinter optional. $34.95 (Disk Only)

FILE-IT 2 An **expanded database system** which extends FILE-IT, provides the following additional capabilities. User controlled data selection for creating subfiles from main data files. Random access file updating for label and financial data files. Financial entry and report generator programs provide data selection by code(s) and/or date(s). Monthly bar graph program generates visual pictures of selected data on screen/printer. Requires 1 disk drive, minimum of 24K RAM, and an 80 column printer. Supports single or multiple disk drives. Includes detailed documentation, users manual and utility programs. $49.95

EDUCATIONAL/ENTERTAINMENT

MY FIRST ALPHABET Will give your youngster an unparalleled learning experience. Complete with melodies and thirty-six professional drawings, children see pictures and hear tunes with letters and numbers of their own choosing or the ones you choose to show. Package includes a GRAPHICS EDITOR for creating, editing and VIEWING your own custom drawings in real time. Use drawings with any basic program or as part of MY FIRST ALPHABET. Instruction manual included. Minimum of 24K and disk required. $29.95

WORDGAMES This package is jam-packed with hours of fun and challenge. Wordgames contains GUESSIT, WORDJUMBLE and POSSIBLE. GUESSIT, a deductive alphabetic reasoning game for 1 or 2 players can also be used for teaching or learning dictionary look-up skills. Comes with 60 word vocabulary. WORDJUMBLE is a multiple word descrambling puzzle with play-on-word hints and mystery answers. Comes with 20 puzzles. Instructions show how to substitute your own words and clues. Use POSSIBLE as a word game tool to assist creating or playing word or letter scrambling games by showing all combinations of letters you supply. If you like word games you will love this package. 16K cassette $14.95; 24Kdisk $19.95

UTILITIES

PROGRAMMING AIDS PACKAGE I Is four utility programs to help increase programming efficiency and learn more about your computer. RENUMBER handles references and **even** variables. Generates Diagnostic Tables for programming error detection. PROGRAM DECODER, DECIMAL TO BCD and BCD TO DECIMAL programs give you a practical way of studying internal program representation and ATARI number-conversion procedures. Comes with comprehensive users manual. 16K cassette $14.95; 24K disk $19.95

SWIFTY UTILITIES Make programming time more efficient; increase programming productivity. Includes all of PROGRAMMING AIDS I plus has REM Remover, Variable Lister, Disklist, DOS Caller (access DOS Utilities with program in core), MENU/SELECTOR (to run programs in either saved or listed format) and Custom Print (for preparing condensed and indented program listings on either EPSON MX-80's or ATARI 825 printers). (Listings skip page perforations and title and number pages.) Minimum of 24K disk system required. $29.95

DISKETTE INVENTORY SYSTEM Use this system to gain control of your expanding disk/program inventory. Quickly get locations of single or multiple copies of your programs and all your valuable files. An invaluable tool, this system is easy and convenient to use and to update. 24K disk system required. $24.95 Printer suggested.

GRAPHICS EDITOR (refer to MY FIRST ALPHABET.)

TO ORDER SEND CHECK OR MONEY ORDER TO

SWIFTY SOFTWARE, INC.

P.O. BOX 641
MELVILLE, N.Y. 11747
[516] 549-9141

Atari is a registerd trademark of Atari, Inc., and all references to Atari should be so noted.

N.Y. RESIDENTS ADD 7% SALES TAX

AVATAR SOFTWARE

FOR

ATARI

HOME MANAGER

Part I

Cash Flow

This month AVATAR SOFTWARE introduces a totally integrated package for the home.

HOME MANAGER is a series of programs which cover cash flow, budget, time scheduling, filing, inventory and gas consumption. Each program allows you to input up to 200 entries per month as both income and expense, in category names that can be used as is or modified to suit your own needs. All entries are saved on disk. Categories can be viewed on the screen or can be printed.

You don't have to be a computer programmer to use this package! The computer does the work for you. Along the way your choices and inputs are carefully monitored by the program to insure that only the correct data is being used and saved.

Cash Flow, written by Jerry Falkenhan, has been tested for over six months with non-computer oriented people. The result is a nice, clean, smooth-running package that makes using the computer an enjoyable experience with useable, practical applications.

Cash Flow is a total package in itself, and is also a module that fits into the **Home Manager Package.** As these modules become available, they can be purchased and put into the binder indexes already provided for you. You can then tailor our package to your own personal needs.

Cash Flow comes in a handsome, 3-ring binder with documentation.

CASH FLOW, BINDER & DOCUMENTATION = $50.00

*****

PERSONAL QUICK EDITOR

Our **Personal Quick Editor** is the closest package to a true word processing system without the high price! The program allows you to create, edit, delete & move text. Vertical scrolling, elongated text, special text formatting abilities, centering and elongated text centering, etc., all combined into an easy to learn and use form makes the **Personal Quick Editor a real buy at only $24.95.**

2096-A Walsh Ave.
Santa Clara, Ca. 95050
Phone (408) 988-5399

www.commodore.ca

Program 1.

```
10 REM *** DYNAMIC PLAYER ANIMATION WITH ATARI ***
20 REM BY Alan Watson
30 REM June 25, 1981
100 REM *** DIMENSION STRINGS & READ PLAYER DATA ***
110 DIM P$(81),V$(27),D$(3)
120 FOR I=1 TO 27
130 READ D$
140 P$(3*I-2,3*I)=D$
150 NEXT I
200 REM *** SET P/M GRAPHICS ***
210 GRAPHICS 2+16:SETCOLOR 4,7,2
220 X=127:Y=63
230 POKE 559,46
240 I=PEEK(106)-8:POKE 54279,I
250 PMBASE=I*256
260 POKE 53277,3
270 FOR I=PMBASE+512 TO PMBASE+640:POKE I,0:NEXT I
280 POKE 704,22
290 POKE 53248,X
300 REM *** VIEW POINTER & STRING ***
310 C=C+1
320 IF C>4 THEN C=1
330 ON C GOTO 340,350,340,360
340 V$=P$(1,27):GOTO 370
350 V$=P$(28,54):GOTO 370
360 V$=P$(55,81)
370 FOR I=1 TO 9
380 POKE PMBASE+512+Y+I,VAL(V$(3*I-2,3*I))
385 IF C=2 OR C=4 THEN SOUND 0,28*I,6,9-I
390 NEXT I
400 REM *** MOTION ROUTINE ***
410 A=STICK(0)
420 IF A=15 THEN 310
430 IF A=11 THEN X=X-1:POKE 53248,X
440 IF A=7 THEN X=X+1:POKE 53248,X
450 IF A=13 THEN FOR J=11 TO 0 STEP -1:POKE PMBASE+512+Y+J,PEEK(PMBASE+511+Y+J):NEXT J:Y=Y+1
460 IF A=14 THEN FOR J=1 TO 11:POKE PMBASE+511+Y+J,PEEK(PMBASE+512+Y+J):NEXT J:Y=Y-1
470 GOTO 310
500 REM *** BIT-MAP DATA FOR EACH VIEW ***
510 DATA 126,090,066,060,219,189,102,102,231
520 DATA 126,090,066,060,219,189,102,230,007
530 DATA 126,090,066,060,219,189,102,103,224
```

©

GREAT GAMES FOR ATARI

Gin Rummy 3.0

The classic computer Gin Rummy game now available for Atari, with color graphics and sound. Plays a regulation game, and a tough game that will hold its own against anyone. Keeps score to game level. Disk version keeps a running score in disk file so you can start another session where you left off. 24K Cass. IGR/C $21.95, Disk IGR/D $26.95.

Casino Blackjack/Counter

Play at a very realistic casino table — learn to beat the house at its own game, or just play for fun following the recommended bets. You play one of five hands (the computer plays the others), and practice card counting as the cards are dealt. Choose up to 6 decks to play against, and set the dealing speed to slow, medium or fast. 16K Cass. IBJ/C $19.95, Disk IBJ/D $24.95.

Concentration

Excellent full-color graphics, great fun to play on the screen. Where *was* that sailboat you saw two turns ago? Choose up to 15 pairs of figures—a smaller game is fascinating for children, and 15 pairs will challenge anyone. For 2 players. 16K Cass. ICO/C $14.95.

Atari 3-Game Pack

These three great Atari games — Gin Rummy, Blackjack and Concentration — on a single disk, and at less than the cost of the three separately. $49.95.

ALL PROGRAMS OPERATE WITH JOYSTICKS

AT YOUR DEALER OR DIRECT FROM:

MANHATTAN SOFTWARE
POST OFFICE BOX 1063
WOODLAND HILLS, CA 91365

California residents add 6% sales tax
24-hour Visa and MasterCard order line:
(213) 704-8495

ATARI 800 SOFTWARE!!

TRS-80 EXTENDED BASIC COLOR COMPUTER SOFTWARE

TRS-80 POCKET COMPUTER SOFTWARE!!

FROM: SEBREE'S COMPUTING
456 Granite Avenue, Monrovia, Calif., 91016

ATARI 800 16K min. 40K Preferred.>>>GRAPHICS EDITOR!!!<<< NOW, both 2-D and 3-D scenes can be designed with a JOYSTICK, and then saved to disk!!!! These scenes can then be loaded in later to be edited before you save it again under another name!! All of this can be done using ANY graphics mode!!! But that's not all!!! You can save entire screens OR just individual images in 2-D OR 3-D!! You have the option of giving the new scenes different file names, a MENU of disk files is shown on the screen as you choose the file name. A 'HELP' option is included sould you have trouble with ANY operation. If you decide to use the 3-D option, you may change the 3-D view(s) of the object(s) on the screen. Uses Player-Missile Graphics for the 'cursor'. POWERFUL, AND WELL DOCUMENTED!! Disk version is recommended (--add $5.00--). W/5 Programs. ONLY $29.95 +$1.50 p&h. on Cassette.

COLOR COMPUTER EXTENDED BASIC>>>3-D C.C. GRAPHICS PACKAGE!!<<< NOW, you can get our popular 3-D Graphics Package for your Color Computer!! Design your own graphics with a Joystick and view these images from any angle you want!!! Software selectable screen resolutions, Colors, Viewing angles, Rotation, object erase & replacement, wide-angle or telephoto views, along with ALL of the required 3-D operations to change viewers location.W/listings!ONLY $24.95+$1.50 p&h.cassette

>FLIGHT SIMULATOR<Req. 24K ATARI or 16K TRS-80 COL.CMP.A Graphic Flight simulator for 1 player. You have to take-off & navigate to the next airport -watching obsticles!! Then attempt to land at the airport- if you have enough fuel! Great graphics! Different difficulty levels.Requires one joystick.Only-$17.95 +.95p&h

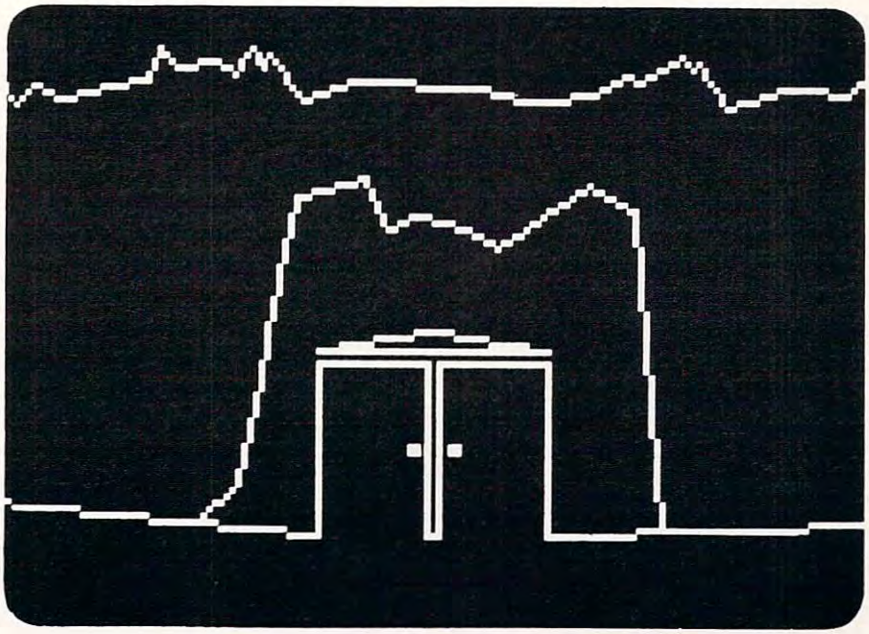
>TRIP TO JUPITER SPACE ADVENTURE!< Req. 24K ATARI or 16K COLOR COMPUTER. Launch your space craft from Earth & get on a trajectory to JUPITER! Obsticles to navigate through! Land on JUPITER and re-launch your craft & bring it to orbit and re-connect with the Mothercraft and head back to Earth!! ONLY-$18.95 + $.95 p&h

>3-D RED BARON DOGFIGHT/ FLIGHT SIMULATOR!< Req.16K ATARI or 16K C.CMP. NOW you can play this exciting 3-D simulation /game on both computers! Done in HI-RES!! You are in pursuit of the famed RED BARON, and catching up to him. If you don't shoot him down soon, his tail gunner starts shooting at you!! Out-of-the cockpit view, w/ALTIMETER, RADAR, BANKING METER, & NUMBER OF WINS. Only-$16.95 +.95 p&h

TRS-80 POCKET COMPUTER>>WUMPUS ADVENTURE!<< Now you can play our popular WUMPUS game on your P.C. With A 4 page manual and listing, on cassette--$7.95 +.95 p&h Pay by Check or Money Order (preffered).Foriegn-U.S.funds ONLY.CAL.6% sales.tax

www.commodore.ca

AND NOW BEHOLD THE ENTRANCE TO THE PLACE KNOWN AS DEVIL DWELL!



COMPUTER AGE SOFTWARE

CA001 "Atari Epson Screen Dump" is a screen dump program that dumps a screen image (up to GR.7) to the Epson proportionally.

CA003 "Atar-Renum" is a general utility that will renumber any tokenized BASIC program that is co-resident in RAM. Requires only 3565 bytes of RAM.

CA004 "InfoFile" is a program designed to act as an electronic file cabinet. A "dynamic keyboard" moves the user quickly through this menu driven program. This is a "fast" database program. Use it to create, add, delete, edit, print, selectively search, and store your custom files. All files can be secured w/ code.

CA005 "Binary Load Cassette to Disk" is a utility that will take binary load cassette files like SPACE INVADERS (TM) and allow their transfer to disk.

CA006 "Ork Attack" has been renamed previous to release as "DEVIL DWELL." This adventure program is not easily beaten, has good graphics, and an excellent user dialogue.

CA007 Our long awaited "Smart Terminal Emulator Program" has also had a name change. We are very happy to announce that "DOWNLOADER" is now available. This fine piece of software allows you to download information to: Disk, Cassette, or Printer.

SWEDE 1 is a package of four programs (3-D, LUNAR LANDER, ALIEN ATTACK, and SPACE BATTLE) which is meant to be studied as well as enjoyed. It covers mainly the mysterious world of Player/Missile Graphics. By studying the programs you will learn how to smoothly move an object, such as a space capsule, horizontally, vertically, and diagonally. You will also learn how to make the player fire and rotate 360 degrees. Also included are sections on the Cursor, the ESCape key and conversions of other BASICs into Atari BASIC.

COMPUTER AGE SOFTWARE

9433 GEORGIA AVE.
SILVER SPRING, MD 20910
(301) 588-6565

Atari is a registered trademark of Atari, Inc.

www.commodore.ca

Editor's Note: We present in the following article, the most comprehensive Atari memory information ever published in a magazine. Because of its length, we had to make a tradeoff between source code size and magazine fit. Though small, it's quite readable and is arranged for your ease of use. Enjoy it. — RCL.

Shoot

John H. Palevich
Bethesda, MD

Editor's Note: This article provides a good game, a way to create cassette Boot tapes, and extensive material for study on the Atari's machine language techniques. RM

Shoot is a machine language arcade style game that must be initialized on a 16K or greater Atari with or without DOS, but will run on ANY Atari, even an Atari 400 with 8K of RAM!

O.K. Before I tell you everything you ever wanted to know about how you too can write machine language video games for your Atari, I'm going to let you see just such a game. Stop reading this paragraph for a moment, and go and look at program 1. Program 1 is a Basic program that takes about 6K to run. It will take the machine language program that I've encoded in the data statements and write it out onto a cassette tape. But this cassette tape is no ordinary cassette tape — it's a *Boot Tape*.

What, you may ask, is a Boot Tape? It is the name of a tape that has a machine language program on it, along with information to tell the Atari how to load it into memory and where to jump to begin execution. Space Invaders is an example of a program that Atari offers in boot tape form. You can think of a boot tape as a do-it-yourself ROM Pac, since you need not have Basic (or any other cartridge) installed in your Atari at the time you 'boot' (short for boot-strap as in "to pull oneself up by one's boot-straps") the boot tape.

So what I want you to do now is warm up the Atari, type in the program in program 1, and run it. To those of you with only 8K: sorry, you'll have to type this in on a friend's machine. Be careful with those DATA statements! When you run the program, one of four things will happen:

1. It prints the line numbers of the data statements on the screen, Beeps the bell twice, and saves a perfect copy of the Boot Tape on the cassette, and stops.

2. It prints some of the line numbers, but stops with the message "Error in line #1040"

3. It prints some of the line numbers, but stops with the message "Too many/few lines"

4. It does something else (like crash).

In case 1, you can smile and move on to the next paragraph. In case two, check the line number mentioned in the error statement against the same line in program 1. They won't be identical, so fix your mistake. In case 3, make sure that line 200 is entered correctly and also check that you've not forgotten to type in any of the data statements. In case 4, make sure that the string on line 300 is: 'hhh', reverse-video-asterisk, 'LV', reverse-video-d. If it is, then that's not the problem which means you've come up with a totally new error, so congratulate yourself and try again.

Now, take the boot tape you just wrote and go over to ANY Atari computer. Open the lid and remove the ROM pack. Turn off all the peripherals (especially all 815's, 810's, and 850's) except for the cassette recorder. Put the boot tape in the cassette recorder, rewind it, and press 'Play'. Turn off the Atari 400/800, press down on the START button, and turn the 400/800 back on. It should beep once, which is your signal to press the return key and wait. The boot tape will load into the RAM of your Atari. Once there, the cassette will stop and the game will begin!

First you will see a copyright message — must to make sure everybody knows that I wrote it — which will last for about 8 to 12 seconds. Then the message will disappear and three zeros will appear. The left (green) one is your score. The middle (red) one is your high score. The right (yellow) one is time remaining. Plug a joystick into controller jack 1 (far left) and press the start button.

Shazam! Eight rows of assorted sizes and colors of airplaned, helicopters and saucers will start flying hither and yon across the screen. Push the joystick left and right to aim the gun, press the button to fire the missile, then use the joystick to guide the missile into one of the planes. If you miss, try again. If you hit the plane it will explode and you will score some points: Helicopter — 5 points, Plane — 10 points, Saucer — 25 points. Clearing a rack of planes within 30 seconds gives you a bonus of 50 points. If you take more than 30 seconds to clear a rack of planes, the game will give you another full rack of planes immediately. For every 15 points you score you get an additional second of play time. When the timer goes to zero, your game ends, the high score is adjusted, and the program waits for you to press on the console buttons: Press START to restart the game. Press OPTION and SELECT down simultaneously to have the program make a copy of itself. If you do this, it will beep twice, wait until you've pressed return, and write a copy of itself to the cassette recorder. (THIS type of copying can be done on ANY size Atari, but first you have to have a working Boot Tape ... which is why you've spent so much time typing in those data statements!).

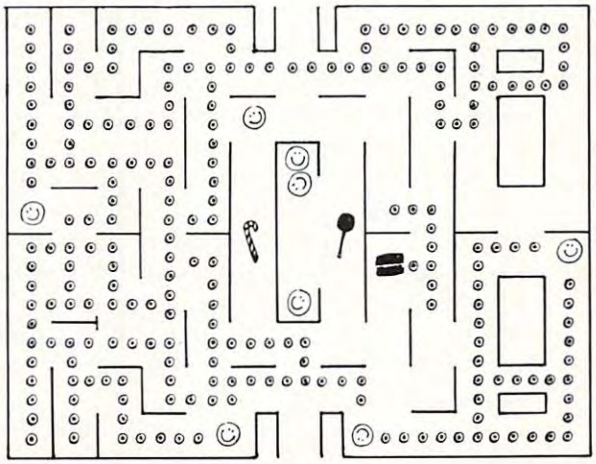
Well, if you are afraid of machine language, or don't want to program, you can stop reading this article at this point and go back to playing my game. But if you want to know how it works, read on ...

www.commodore.ca

ON-LINE systems

Presents

JAW BREAKER



For The ATARI 400/800

This arcade game takes you to the candy store for a wild game of tag with some of the rowdiest playmates you'll find anywhere. If you can eat all the sweets, the bratty kids will stop bothering you and after a quick stop and a brushing of the teeth, it's off to the store for another day of sweets and tag.

- Full Color Hi-Res Graphics
- Automatically Esculated Skill Level
- Quick Moving Animation

Jaw Breaker runs on any ATARI 400/800 with at least 24K of memory and is available now for $29.95 on disk from your local computer store or you may order directly from us.

ORDERS MAY BE CHECK, VISA, MASTERCHARGE OR C.O.D. DEALER INQUIRIES INVITED

ON-LINE SYSTEMS — 36575 Mudge Ranch Road — Coarsegold, CA 93614 — 209-683-6858

Now Take Along Your Atari* in MICROCASE

lightweight, rugged protection for your Atari* investment

Perfect for Home, School, Office

Protects your Atari* from dirt, scuffs, damage

Made from durable CODURA

Completely machine washable

Holds Atari* 800 or 400 and accessories

Insulated with 5" thick high density foam

Measures: 14" x 18" x 10"

Weighs only 3 lbs. 10 oz.

Choose: Cobalt Blue with navy trim or Desert Tan with chocolate trim

Also available Microcase II for 810 Disk Drive, Atari* Printer, Modem, Modem Connector, two power packs

Soon:
Microcase Protection for other leading microcomputers.

Casemaker
13715 Quito Road
Saratoga, CA 95070

For ordering information call toll-free: **800-543-3000** Ext. **3622**

Available at leading computer stores including:

Anderson Computers
Alabama and Tennessee

Computer Corner
Ohio

Convenience Corner Electronics
Oklahoma

Math Box
Maryland, Virginia and Washington, D.C.

On-Line Micro Centers
California, Texas, Oklahoma, and Idaho

P.C.S. Computer Service
Nevada

Quality Technology
Utah

Video Vision Store
Tennessee

*A trademark of Atari used under license Indicates trademark of ATARI Inc.

www.commodore.ca

I've included a listing of the "source code" [*assembler input, usually documented, as here — Ed.*] for SHOOT, but I bet that there won't be enough free room to print it in its entirety. It IS commented very well, and all you assembly language freaks can take it and modify it as much as you like.... You might run into some size problems, since I wrote it on a much larger computer and just transferred the assembled object code over to the Atari, but with a bit of luck, you should be able to cut it down to size. One hint, most of the Atari equates aren't used, so you needn't bother to type them.... I'd bet that you'd need a disk and at least 32K to be able to start changing the code.

In general (and here I start using slang that no one who hasn't read the Atari OS manuals will understand) there are three parts to a video game on the Atari — the setup, the main loop, and the end. In addition to these three parts are two other parts — the Vertical Blank Interrupt (VBI) and the Display List Interrupt (DLI).

The VBI occurs every sixtieth of a second and is used by the Atari to keep the realtime clock, do attract mode, update all sorts of counters, and so forth. The video game maker uses this time to move the players, update the scores, fire the shots, decrement the time left, and do sound effects.

The DLI occurs whenever the ANTIC chip reads a display list instruction that has the high bit set. The Atari OS doesn't use the DLI, but the video game maker can use it to change character sets, playfield colors, and player positioning/width on the fly. In the Super Breakout cartridge, for example, all of the bricks are 'really' the same color, but carefully placed DLI's in the display list change the hue of that color between rows of bricks. The octave bands in the Music Composer cartridge are also DLI generated.

In any event, here's the poop on SHOOT: All the flying objects, AND the gun, are the same player, Player 0. Eleven display list interrupts are used to change the color of the sky/ground, the position of the planes/gun, the color of the planes/gun, and to check if the missile hit the previous plane. Since the sky/ground has no information in it, it need not take up any memory space, so the playfield display uses only the 20 characters in the score line at the top of the screen.

The vertical blank interrupt updates the score, awards bonus time, moves the gun, and fires and moves the missile. When the user runs out of time, a flag called STOP is set to tell the main program that the game is over. If a missile is in flight or the player is scoring points, this routine will also generate the appropriate sound effects.

The main body of the program actually has very little to do. First it draws the playfield, sets up the player missile graphics, knits the VBI and DLI handlers into the operating system, displays the copyright message and zeroes the scores. Then it waits for the user to press some console buttons. START starts the game and OPTION and SELECT save a copy to tape.

When the game is started, the main loop draws a set of planes, sets Count Down Timer #3 for thirty seconds, and waits for either the user to shoot down all the planes in the current rack, or the count down timer to expire, or the game to end. If the user shoots down all the planes within thirty seconds, 50 points are added to the value of the last plane shot down. When either 30 seconds are up, or all the planes are shot down, the main loop draws a whole new set of planes. This goes on until the user runs out of time.

When the user runs out of time (and, if he can shoot a whole rack off the screen in less than ten seconds, he, or she can play forever) the final score is compared with the high score. If the final score is higher than the high score it becomes the new high score. In either event, the program loops to the wait for user input section and the user may play another game.

Well, that's SHOOT in a nutshell. I've hidden most of the gritty details in the comments to the code. Feel free to use any part of my code for a game of your own, with the provision that you don't try to sell it! I'd bet that the VBI and the DLI handlers could be used in conjunction with Basic programs that take care of the slower details. For example, you might want to use the VBI to move a Pong ball and Paddle set while the Basic program took care of the scoring, playfield generation, instructions, and so forth. Best of Luck!

Program 1: The Boot Tape Maker

```
0 DIM H$(1),B$(2),AD$(4),A$(60),BUF$(1148)
1 POKOFF=4*1024-1
2 GOTO 100
10 IF H$>="0" AND H$<="9" THEN D=ASC(H$)-48:RETURN
12 D=ASC(H$)-55:RETURN
20 H$=B$(1,1):GOSUB 10:B=D:H$=B$(2,2):GOSUB 10:B=B*16+D:
   CHECK=CHECK+B:RETURN
30 B$=AD$(1,2):GOSUB 20:AD=B:B$=AD$(3,4):GOSUB 20:AD=AD*256+B:RETURN
100 GRAPHICS 0
110 TRAP 900:LINE=1000:LSUM=0
120 READ A$:IF A$="END" THEN 200
130 ? LINE:CHECK=0:B$=A$(1,2):GOSUB 20:NOB=B
140 AD$=A$(3,6):GOSUB 30:FAD=AD
150 FOR I=1 TO NOB:B$=A$(5+2*I,6+2*I):GOSUB 20
160 M=FAD+I-1-POKOFF:BUF$(M,M)=CHR$(B):NEXT I
165 SUM=CHECK-65536*INT(CHECK/65536)
170 AD$=A$(LEN(A$)-3,LEN(A$)):GOSUB 30
180 IF SUM<>AD THEN 900
185 LSUM=LSUM+SUM:LINE=LINE+10:GOTO 120
200 IF LSUM<>125120 THEN ? "Too many/few lines":END
205 CLOSE #1
210 OPEN #1,8,128,"C:"
220 IOCB=832+16
230 POKE IOCB+2,11
240 BUF=ADR(BUF$)
250 POKE IOCB+4,BUF-(INT(BUF/256)*256)
260 POKE IOCB+5,INT(BUF/256)
270 BUFLEN=LEN(BUF$)
280 POKE IOCB+8,BUFLEN-(INT(BUFLEN/256)*256)
```

www.commodore.ca

```
290 POKE IOCB+9,INT(BUFLEN/256)
300 DUMMY=USR(ADR("hhh±LVd"),16)
310 CLOSE #1
320 END
900 ? "ERROR IN#";LINE:END
1000 DATA 18100000090010081018 60A93C8D02D3A916850AA910850B604C4E06A9
1010 DATA 181018127070704600 18F070F070F070F070F070F070F070F070F00DF0
1020 DATA 18103070F070F0704119102843293139383120 4A20482050414C45076D
1030 DATA 181048564943488090829284948696889 8C8081828384858687880 0A5A
1040 DATA 1810603838000102030405060708 7C7C0102030201FFFEFDFE00000615
1050 DATA 18107801020304050607080 90A0B000100010001000100010100 0100E9
1060 DATA 181090FF000003060C1C3C7EFF00C06030383C7EFF00181818183C077E
1070 DATA 1810A87EFF0000F820F29E90F000001F044F79090F0000010D3F7F0844
1080 DATA 1810C0180000080B0FCFE18000018247E817E000001050008FF05070D
1090 DATA 1810D80000020A0118FE0A011003190020FD190020488A48A6B0E80708
1100 DATA 1810F0BD4C108D0AD48D1AD0A6B0AD08D02901F013A9009D62109D0B70
1110 DATA 1811086D10BD78101865B185B18D1ED0E886B0BD6210187D6D109D0ACE
1120 DATA 18112062108D00D0BD57108D12D0BD83108D08D068AA6840A5B1D00B40
1130 DATA 18113808A9808D03D24C901138E90185B1A98A8D03D2A205BD00180A4A
1140 DATA 181150186901091 09D0018C91AD009A9109D0018CA4C4D11A5B7D00893
1150 DATA 18116827A6B5E886B5E00FD01EA20086B5A205BD0E1818690109900A95
1160 DATA 1811809D0E18C99A9009A9909D0E18CA4C7811A6B6E886B6E03CD00C6F
1170 DATA 18119822A20086B6A5B7D02EA205BD0E1838E90109909D0E18C99F0A8B
1180 DATA 1811B0D009A9999D0E18CA4CA311A900A2061D0D18CAD0FA290FC90AA9
1190 DATA 1811C800D004A90185B7A900854DAD78024A4AAABD8E1085B3CA8A0B72
1200 DATA 1811E00A0A0AAAA000BD921099601AC899601AE8C8C010D0F0A5B20C55
1210 DATA 1811F81865B385B28D04D0A5B4F026AAA9009D8019CAF011A5B1D00DD2
1220 DATA 1812101286B4A9FF9D80198E00D24C2A1286B44C2A12A2008E00D20A10
1230 DATA 18122886B4A5B7D016AD8402D011A5B4D00DA96285B4A5B30A0A180BE0
1240 DATA 181240698485B2A9FF85B08D1ED04CD1E7A9A88D01D2A9808D03D20E26
1250 DATA 181258A9008D00D2A9308D02D2A280A9009DFF199D7F19CAD0F7A90CB3
1260 DATA 18127000A2089DFFCFCAD0FAA92E8D2F02A9188D07D4A9038D1DD00C27
1270 DATA 181288A9100D6F028D6F028D1BD0A90085B4A90185B7A9408D0ED40A7F
1280 DATA 1812A0A9108D3102A9198D3002A9108D0102A9EA8D0002A211A03508B7
1290 DATA 1812B8A906205CE4A9C08D0ED4A9C68DC402A9368DC502A9188DC60CD2
1300 DATA 1812D002A90A8DC702A214BD3710200E1409C09DFF17CAD0F2A5130AC1
1310 DATA 1812E8186903C513D0FCA214A9009DFF17CAD0FAA9108D0518A9500C3C
1320 DATA 1813008D0C18A9908D1318A90185B7A9088D1FD0AD1FD0C901D0060A1C
1330 DATA 181318202A144C4E12C906D0EFA900A2069D0D189DFF17CAD0F7A90ADB
1340 DATA 181330918D1118A9928D1218A9908D1318A9108D0518A90085B7850952
1350 DATA 181348B185B685B5A218A00020C513C8C008D0F8A207A0D0A9038D0C95
1360 DATA 1813602A02205CE4A9C08D0ED4AD2A02D0034C4D13A008A900196C091D
1370 DATA 1813781088D0FAC900D00AA9321865B185B14C4D13A5B7F0DCA5130C73
1380 DATA 181390186902C513D0FCA200BD0718291FDD0018F005B0084CB0130959
1390 DATA 1813A8E8E006D0EC4C0813A206BDFF17291F09409D0618CAD0F34C0B64
1400 DATA 1813C008130000008EC2138CC313AD0AD22907C906B0F70A0AAABD0975
1410 DATA 1813D8D210996D10BDD3109978 10BDD41099831 0A900996210BDD50BCF
1420 DATA 1813F010AAACC213A9088DC413BDAA1099001AE8C8CEC413D0F3980D45
1430 DATA 181408AAACC31360008C0D14A88A48982A2A2A2A2903AA98299F1D087A
1440 DATA 181420F6FEA868AA98AC0D1460A220A90C9D42032056E4A9149D450B11
1450 DATA 18143803A9759D4403A9039D4203A9089D4A03A9809D4B032056E40900
1460 DATA 181450A9009D4403A9109D4503A9789D4803A9049D4903A90B9D4208D9
1470 DATA 101468032056E4A90C9D42032056E460433A9B0652
1480 DATA END
```


ATARI ™ OWNERS

SUB HUNT–It's you against the enemy subs. Blow up as many subs in the allowed time and get high score. Three levels of play. Great sound and graphics. Requires joystick 32k disk/ 19.95 24k, cass./12.95

HORSE RACE–It's a day at the races. Up to five can play. Pick horse and place bet with show, place or win!! Real odds and good graphics. 32k/disk 19.95, 24k cass./12.95

TIE FIGHTER–Try to blow up the tie fighter in the least amount of time to score!! Requires joystick. 24k disk/17.95, 16k cass./11.95

CLARKSTON SOFTWARE
10001 ELLIS RD.
CLARKSTON, MICH. 48016
CHECK OR MONEY ORDER

World Class

Your **ATARI** is a world class personal computer. But you need great software in order to exploit its capabilities. And you need information about how it all works.

IRIDIS is a series of software packages that will help you enjoy and understand your **ATARI** more fully. The programs are outstanding, just as you would expect from the people who have published 23 issues of the widely acclaimed *CURSOR Magazine* for the Pet since 1978. But **IRIDIS** is more than just a collection of excellent programs. **IRIDIS** #2 comes with a 56-page manual that has clear, detailed explanations of how each program works. The explanations tell you line-by-line what each program does, and how it does it.

IRIDIS and your **ATARI**: A winning team. World Class!

IRIDIS #2 - Fondedit and Knotwork programs.
Includes 56 page User Manual.
$15.95 Cassette, $18.95 disk.
Mastercharge and Visa welcome.

Published By: THE CODE WORKS
Box 550
Goleta, CA 93116
805-683-1585

A Revolutionary Concept In Software For The ATARI* 400 and 800 Computers

The Interactive Storybook

Sammy The Sea Serpent
A Storybook Program For Children Ages 4 to 7.

Sammy The Sea Serpent
is the story of an imaginery sea creature who is lost and trying to find his way home. The story is read aloud to your child by a professional actress. While the tale is being told, the child uses the joystick to help Sammy out of some tight spots.

The A side of the cassette contains the interactive story; the B side contains games that the child plays with Sammy.

The program uses voice, sound effects, music, color and mixed graphics.

Sammy The Sea Serpent
can be used with either the ATARI 400 or 800 and requires 16K. It is available in cassette format only. Price is $16.95 plus $2.00 shipping and handling.

Also available at fine computer stores.

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

ATARI is the registered trademark of ATARI, Inc.

www.commodore.ca

```
                 ;-*-MIDAS-*-
                          .ENABLE LC
                 .NLIST SEQ
                 .TITLE SHOOT
                 .SBTTL by John H. Palevich
                 ;
                 ;
                 ;       "Shoot" -- inspired by the Atari VCS "Air Sea Battle" cartridge.
                 ;
                 ;       Feel free to modify & give away.  Don't even think of selling this.
                 ;       O.K.  If you want to assemble this on your Atari, you'll have to
                 ; change some things around.  Most changes can be inferred from looking
                 ; at the code that this assembler produced and figuring out what you'd
                 ; have to type to get the Atari to do the same thing.
                 ;
                 ;       Some hints, though:
                 ;               ^       means /256
                 ;               .*      means *
                 ;               .ASCII  means .BYTE
                 ;               . (in expressions) means the value of the program counter
                 ;
                 ;               Lables shouldn't have ':'s.
                 ;
                 ;       Colleen Operating System Equate File
                 ;
                 ;
E000             CHHORG  =       $E000   ;CHARACTER SET
E400             VECTBL  =       $E400   ;VECTOR TABLE
E480             VCTABL  =       $E480   ;RAM VECTOR INITIAL VALUE TABLE
E4A6             CIOORG  =       $E4A6   ;CENTRAL I/O HANDLER
E6D5             INTORG  =       $E6D5   ;INTERRUPT HANDLER
E944             SIOORG  =       $E944   ;SERIAL I/O HANDLER
EDEA             DSKORG  =       $EDEA   ;DISK HANDLER
EE78             PRNORG  =       $EE78   ;PRINTER HANDLER
EF41             CASORG  =       $EF41   ;CASSETTE HANDLER
F0E3             MONORG  =       $F0E3   ;MONITOR/POWER UP MODULE
F3E4             KBDORG  =       $F3E4   ;KEYBOARD/DISPLAY HANDLER
                 ;
                 ;
                 ;
                 ;       VECTOR TABLE
                 ;
                 ;       HANLDER ENTRY POINTS ARE CALLED OUT IN THE FOLLOWING VECTOR
                 ;       TABLE.  THESE ARE THE ADDRESSES MINUS ONE.
                 ;
                 ;       Example for Editor
                 ;
                 ;       E400    Open
                 ;          2    Close
                 ;          4    Get
                 ;          6    Put
                 ;          8    Status
                 ;          A    Special
                 ;          C    Jump to Power on initialization routine
                 ;
                 ;
E400             EDITRV  =       $E400   ;EDITOR
E410             SCRENV  =       $E410   ;TELEVISION SCREEN
E420             KEYBDV  =       $E420   ;KEYBOARD
E430             PRINTV  =       $E430   ;PRINTER
E440             CASETV  =       $E440   ;CASSETTE
                 ;
                 ;       JUMP VECTOR TABLE
                 ;
                 ;THE FOLLOWING IS A TABLE OF JUMP INSTRUCTIONS
                 ;TO VARIOUS ENTRY POINTS IN THE OPERATING SYSTEM
                 ;
E450             DISKIV  =       $E450   ;disk initialization
E453             DSKINV  =       $E453   ;disk interface
E456             CIOV    =       $E456   ;central input output routine
                 ;
0000             ICZERO  =       $0      ;proper values for X
0010             ICONE   =       $10     ;when calling CIOV for each
0020             ICTWO   =       $20     ;of the IO control Blocks
0030             ICTHRE  =       $30
0040             ICFOUR  =       $40
0050             ICFIVE  =       $50
0060             ICSIX   =       $60
0070             ICSEVE  =       $70
                 ;
E459             SIOV    =       $E459   ;serial input output routine
E45C             SETVBV  =       $E45C   ;set system timers routine
                 ; With respect to SETVBV, the call sequence is
                 ; X - MSB of vector/timer
                 ; Y - LSB of vector/timer
                 ; A - # of vector to hack
0001             SETMR1  =       1       ;Timer  1
0002             SETMR2  =       2       ;       2
0003             SETMR3  =       3       ;       3
0004             SETMR4  =       4       ;       4
0005             SETMR5  =       5       ;       5
0006             SETIMM  =       6       ;Immediate VBLANK
0007             SETDEF  =       7       ;Deffered VBLANK
                 ;
E45F             SYSVBV  =       $E45F   ;SYSTEM VERTICAL BLANK CALCULATIONS
E462             XITVBL  =       $E462   ;EXIT VERTICAL BLANK CALCULATIONS
E465             SIOINV  =       $E465   ;SERIAL INPUT OUTPUT INITIALIZATION
```

```
0008             WARMST  =       $0008   ;WARM START FLAG
0009             BOOTQ   =       $0009   ;SUCCESSFUL BOOT FLAG <WAS BOOT?>
000A             DOSVEC  =       $000A   ;DISK SOFTWARE START FLAG
000C             DOSINI  =       $000C   ;DISK SOFTWARE INIT ADDRESS
000E             APPMHI  =       $000E   ;APPLICATIONS MEMORY HI LIMIT
                 ;
                 ;       CLEARED ON A COLD OR WARM START
0010             INTZBS  =       $0010   ;INTERRUPT HANDLER
0010             POKMSK  =       $0010   ;SYSTEM MASK FOR POKEY IRQ HANDLER
0011             BRKKEY  =       $0011   ;BREAK KEY FLAG
0012             RTCLOCK =       $0012   ;REAL TIME CLOCK (IN 16 MSEC UNITS)
                 ;
0015             BUFADR  =       $0015   ;INDIRECT BUFFER ADDRESS REGISTER
                 ;
0017             ICCOMT  =       $0017   ;COMMAND FOR VECTOR
                 ;
0018             DSKFMS  =       $0018   ;DISK FILE MANAGER POINTER
                 ;
001A             DSKUTL  =       $001A   ;DISK UTILITIES POINTER
                 ;
001C             PTIMOT  =       $001C   ;PRINTER TIME OUT REGISTER
001D             PBPNT   =       $001D   ;PRINTER BUFFER POINTER
001E             PBUFSZ  =       $001E   ;PRINT BUFFER SIZE
001F             PTEMP   =       $001F   ;TEMPORARY REGISTER
                 ;
0020             ZIOCB   =       $0020   ;ZERO PAGE I/O CONTROL BLOCK
0010             IOCBSZ  =       16      ;NUMBER OF BYTES PER IOCB
0080             MAXIOC  =       8*IOCBSZ        ;LENGTH OF THE IOCB AREA
0020             IOCBAS  =       $0020
0020             ICHIDZ  =       $0020   ;HANDLER INDEX NUMBER (FF == IOCB FREE)
0021             ICDNOZ  =       $0021   ;DEVICE NUMBER (DRIVE NUMBER)
0022             ICCOMZ  =       $0022   ;COMMAND CODE
0023             ICSTAZ  =       $0023   ;STATUS OF LAST IOCB ACTION
0024             ICBALZ  =       $0024   ;BUFFER ADDRESS LOW BYTE
0025             ICBAHZ  =       $0025   ;BUFFER ADDRESS HIGH BYTE
0026             ICPTLZ  =       $0026   ;PUT BYTE ROUTINE ADDRESS - 1
0027             ICPTHZ  =       $0027   ;
0028             ICBLLZ  =       $0028   ;BUFFER LENGTH LOW BYTE
0029             ICBLHZ  =       $0029
002A             ICAX1Z  =       $002A   ;AUXILIARY INFORMATION FIRST BYTE
002B             ICAX2Z  =       $002B
002C             ICSPRZ  =       $002C   ;TWO SPARE BYTES (CIO LOCAL USE)
002E             ICIDNO  =       ICSPRZ+2        ;ICOB NUMBER X 16
002F             CIOCHR  =       ICSPRZ+3        ;CHARACTER BYTE FOR CURRENT OPERATION
                 ;
0030             STATUS  =       $0030   ;INTERNAL STATUS STORAGE
0031             CHKSUM  =       $0031   ;CHECKSUM (SINGLE BYTE SUM WITH CARRY)
0032             BUFRLO  =       $0032   ;POINTER TO DATA BUFFER (LO BYTE)
0033             BUFRHI  =       $0033   ;POINTER TO DATA BUFFER (HI BYTE)
0034             BFENLO  =       $0034   ;NEXT BYTE PAST END OF DATA BUFFER (LO BYTE)
0035             BFENHI  =       $0035   ;NEXT BYTE PAST END OF DATA BUFFER (HI BYTE)
0036             CRETRY  =       $0036   ;NUMBER OF COMMAND FRAME RETRIES
0037             DRETRY  =       $0037   ;NUMBER OF DEVICE RETRIES
0038             BUFRFL  =       $0038   ;DATA BUFFER FULL FLAG
0039             RECVDN  =       $0039   ;RECIEVE DONE FLAG
003A             XMTDON  =       $003A   ;TRANSMISSION DONE FLAG
003B             CHKSNT  =       $003B   ;CHECKSUM SENT FLAG
003C             NOCKSM  =       $003C   ;NO CHECKSUM FOLLOWS DATA FLAG
                 ;
                 ;
003D             BPTR    =       $003D
003E             FTYPE   =       $003E
003F             FEOF    =       $003F
0040             FREQ    =       $0040
0041             SOUNDR  =       $0041   ;NOISY I/O FLAG (ZERO IS QUIET)
0042             CRITIC  =       $0042   ;DEFINES CRITICAL SECTION (CRITICAL IF NON-ZERO)
                 ;
0043             FMSZPG  =       $0043   ;TOTAL OF 7 BYTES FOR DISK FILE MANAGER ZERO PAGE
                 ;
                 ;
004A             CKEY    =       $004A   ;FLAG SET WHEN GAME START PRESSED
004B             CASSBT  =       $004B   ;CASSETTE BOOT FLAG
004C             DSTAT   =       $004C   ;DISPLAY STATUS
                 ;
004D             ATRACT  =       $004D   ;ATRACT FLAG
004E             DRKMSK  =       $004E   ;DARK ATRACT FLAG
004F             COLRSH  =       $004F   ;ATRACT COLOR SHIFTER (EOR'D WITH PLAYFIELD COLORS)
                 ;
                 ;LEDGE   =       2       ;LMARGN'S VALUE AT COLD START
                 ;REDGE   =       39      ; .
0050             TMPCHR  =       $0050
0051             HOLD1   =       $0051
0052             LMARGN  =       $0052   ;LEFT MARGIN (SET TO ONE AT POWER ON)
0053             RMARGN  =       $0053   ;RIGHT MARGIN (SET TO ONE AT POWER ON)
0054             ROWCRS  =       $0054   ;CURSOR COUNTERS
0055             COLCRS  =       $0055
0057             DINDEX  =       $0057
0058             SAVMSC  =       $0058
005A             OLDROW  =       $005A
005B             OLDCOL  =       $005B
005D             OLDCHR  =       $005D   ;DATA UNDER CURSOR
005E             OLDADR  =       $005E
0060             NEWROW  =       $0060   ;POINT DRAW GOES TO
0061             NEWCOL  =       $0061
0063             LOGCOL  =       $0063   ;POINTS AT COLUMN IN LOGICAL LINE
0064             ADRESS  =       $0064
0066             MLTTMP  =       $0066
0068             SAVADR  =       $0068
006A             RAMTOP  =       $006A   ;RAM SIZE DEFINED BY POWER ON LOGIC
```

```
                 ;
0290             TXTROW  =       $0290   ;Text rowcrs
0291             TXTCOL  =       $0291   ;Text colcrs
0293             TINDEX  =       $0293   ;Text index
0294             TXTMSC  =       $0294   ;fools convert into new msc
0296             TXTOLD  =       $0296   ;oldrow and oldcol for text (etc.)
02A2             ESCFLG  =       $02A2   ;Escape flag
02B2             LOGMAP  =       $02B2   ;Logical line start bit map
02B6             INVFLAG =       $02B6   ;Inverse video flag (toggled by Atari key)
02B7             FILFLG  =       $02B7   ;Fill flag for draw
02BB             SCRFLG  =       $02BB   ;Set if scroll occures
02BE             SHFLOK  =       $02BE   ;Shift lock
02BF             BOTSCR  =       $02BF   ;Bottom of screen: 24 Norm, 4 Split.
                 ;
                 ;
                 ;       COLORS
                 ;
                 ;
02C0             PCOLR0  =       $02C0   ;P0 COLOR
02C1             PCOLR1  =       $02C1   ;P1 COLOR
02C2             PCOLR2  =       $02C2   ;P2 COLOR
02C3             PCOLR3  =       $02C3   ;P3 COLOR
02C4             COLOR0  =       $02C4   ;COLOR 0
02C5             COLOR1  =       $02C5   ;COLOR 1
02C6             COLOR2  =       $02C6   ;COLOR 2
02C7             COLOR3  =       $02C7   ;COLOR 3
02C8             COLOR4  =       $02C8   ;COLOR 4
                 ;
                 ;
                 ;       GLOBAL VARIABLES
                 ;
                 ;
02E4             RAMSIZ  =       $02E4   ;RAM SIZE (HI BYTE ONLY)
02E5             MEMTOP  =       $02E5   ;TOP OF AVAILABLE USER MEMORY
02E7             MEMLO   =       $02E7   ,BOTTOM OF AVAILABLE USER MEMORY
02EA             DVSTAT  =       $02EA   ;STATUS BUFFER
                 ;
02F0             CRSINH  =       $02F0   ;CURSOR INHIBIT (00 = CURSOR ON)
02F1             KEYDEL  =       $02F1   ;Key delay
02F3             CHACT   =       $02F3   ;CHACTL REGISTER RAM
02F4             CHBAS   =       $02F4   ;CHBAS REGISTER RAM
02FD             FILDAT  =       $02FD   ;RIGHT FILL DATA (DRAW)
02FB             ATACHR  =       $02FB   ;Atascii character
02FC             CH      =       $02FC   ;global variable for keyboard
02FE             DSPFLA  =       $02FE   ;DISPLAY FLAG: DISPLAYS CNTLS IF NON ZERO;
02FF             SSFLAG  =       $02FF   ;Start/stop flag for paging (CNTL 1). Cleared by Brea
                 ;
                 ;       Page three RAM assignments
                 ;
                 ;
                 ;       Device control blocks
                 ;       (SIO)
0300             DCB     =       $0300   ;Device control block
0300             DDEVIC  =       $0300   ;Peripheral Unit 1 bus I.D. number
0301             DUNIT   =       $0301   ;Unit number
0302             DCOMND  =       $0302   ;Bus command
0303             DSTATS  =       $0303   ;Command Type/status return
0304             DBUFLO  =       $0304   ;Data buffe pointe low
0305             DBUFHI  =       $0305
0306             DTIMLO  =       $0306   ;Device time out in 1 second units
0308             DBYTLO  =       $0308   ;Number of bytes to be transvered low byte
0309             DBYTHI  =       $0309
030A             DAUX1   =       $030A   ;Command Aux byte 1
030B             DAUX2   =       $030B
                 ;
0340             IOCB    =       $0340
0340             ICHID   =       $0340   ;Handler index number (FF = IOCB free)
0341             ICDNO   =       $0341   ;Device number (drive number)
0342             ICCOM   =       $0342   ;Command code
0343             ICSTA   =       $0343   ;Status of last IOCB action
0344             ICBAL   =       $0344   ;Buffer address low byte
0345             ICBAH   =       $0345
0346             ICPTL   =       $0346   ;Put byte routine address - 1
0347             ICPTH   =       $0347
0348             ICBLL   =       $0348   ;Buffer length low byte
0349             ICBLH   =       $0349
034A             ICAX1   =       $034A   ;Auxiliary information first byte
034B             ICAX2   =       $034B
034C             ICSPR   =       $034C   ;four spare bytes
                 ;
03C0             PRNBUF  =       $03C0   ;Printer buffer (40 bytes)
                 ;       (21 spare bytes)
                 ;
                 ;       Page Four Ram Assignments
03FD             CASBUF  =       $03FD   ;Cassette Buffer (131 bytes)
                 ;
0480             USAREA  =       $0480   ; (0480 thru 05FF for the user)
                 ;                         (except for floating point...)
                 ;
                 ;       FLOATING POINT ROM ROUTINES
                 ;
                 ;
                 ;       IF CARRY IS USED THEN CARRY CLEAR => NO ERROR, CARRY SET => ERROR
                 ;
D800             AFP     =       $D800   ;ASCII -> FLOATING POINT (FP)
                 ;                       INBUFF + CIX -> FRO, CIX, CARRY
D8E6             FASC    =       $D8E6   ;FP -> ASCII    FRO -> FOR,FOO+1, CARRY
D9AA             IFP     =       $D9AA   ;INTEGER -> FP
```

COMPUTE! The Journal For Progressive Computing

www.commodore.ca

```
E468     SENDEV  =       $E468   ;SEND ENABLE ROUTINE
E46B     INTINV  =       $E46B   ;INTERRUPT HANDLER INITIALIZATION
E46E     CIOINV  =       $E46E   ;CENTRAL INPUT OUTPUT INITIALIZATION
E471     BLKBDV  =       $E471   ;BLACKBOARD MODE
E474     WARMSV  =       $E474   ;WARM START ENTRY POINT
E477     COLDSV  =       $E477   ;COLD START ENTRY POINT
E47A     RBLOKV  =       $E47A   ;CASSETTE READ BLOCK ENTRY POINT VECTOR
E47D     CSOPIV  =       $E47D   ;CASSETTE OPEN FOR INPUT VECTOR
         ;VCTABL =       $E480
         ;
         ;
         ;       OPERATING SYSTEM EQUATES
         ;
         ;       COMMAND CODES FOR IOCBS
0003     OPEN    =       3       ;OPEN FOR INPUT/OUTPUT
0005     GETREC  =       5       ;GET RECORD (TEXT)
0007     GETCHR  =       7       ;GET CHARACTER(S)
0009     PUTREC  =       9       ;PUT RECORD (TEXT)
000B     PUTCHR  =       $B      ;PUT CHARACTER(S)
000C     CLOSE   =       $C      ;CLOSE DEVICE
000D     STATIS  =       $D      ;STATUS REQUEST
000E     SPECIL  =       $E      ;BEGINNING OF SPECIAL ENTRY COMMANDS
         ;
         ;       SPECIAL ENTRY COMMANDS
0011     DRAWLN  =       $11     ;DRAW LINE
0012     FILLIN  =       $12     ;DRAW LINE WITH RIGHT FILL
0020     RENAME  =       $20     ;RENAME DISK FILE
0021     DELETE  =       $21     ;DELETE DISK FILE
0022     FORMAT  =       $22     ;FORMAT
0023     LOCKFL  =       $23     ;LOCK FILE TO READ ONLY
0024     UNLOCK  =       $24     ;UNLOCK LOCKED FILE
0025     POINT   =       $25     ;POINT SECTOR
0026     NOTE    =       $26     ;NOTE SECTOR
00FF     IOCFRE  =       $FF     ;IOCB "FREE"
         ;
         ;       AUX1 EQUATES
         ;       () INDICATES WHICH DEVICES USE BIT
0001     APPEND  =       $1      ;OPEN FOR WRITE APPEND (D), OR SCREEN READ (E)
0002     DIRECT  =       $2      ;OPEN FOR DIRECTORY ACCESS (D)
0004     OPNIN   =       $4      ;OPEN FOR INPUT (ALL DEVICES)
0008     OPNOT   =       $8      ;OPEN FOR OUTPUT (ALL DEVICES)
000C     OPNINO  =       OPNIN+OPNOT     ;OPEN FOR INPUT AND OUTPUT (ALL DEVICES)
0010     MXDMOD  =       $10     ;OPEN FOR MIXED MODE (E,S)
0020     INSCLR  =       $20     ;OPEN WITHOUT CLEARING SCREEN (E,S)
         ;
         ;       DEVICE NAMES
         ;
0045     SCREDT  =       'E      ;SCREEN EDITOR (R/W)
004B     KBD     =       'K      ;KEYBOARD (R ONLY)
0053     DISPLY  =       'S      ;SCREEN DISPLAY (R/W)
0050     PRINTR  =       'P      ;PRINTER (W ONLY)
0043     CASSET  =       'C      ;CASSETTE
004D     MODEM   =       'M      ;MODEM
0044     DISK    =       'D      ;DISK
         ;
         ;
         ;
         ;       OPERATING SYSTEM STATUS CODES
         ;
0001     SUCCES  =       $01     ;SUCCESSFUL OPERATION
         ;
0080     BRKABT  =       $80     ;BREAK KEY ABORT
0081     PRVOPN  =       $81     ;IOCB ALREADY OPEN
0082     NONDEV  =       $82     ;NON-EXISTANT DEVICE
0083     WRONLY  =       $83     ;IOCB OPENED FOR WRITE ONLY
0084     NVALID  =       $84     ;INVALID COMMAND
0085     NOTOPN  =       $85     ;DEVICE OR FILE NOT OPEN
0086     BADIOC  =       $86     ;INVALID IOCB NUMBER
0087     RDONLY  =       $87     ;IOCB OPENED FOR READ ONLY
0088     EOFERR  =       $88     ;END OF FILE
0089     TRNRCD  =       $89     ;TRUNCATED RECORD
008A     TIMOUT  =       $8A     ;PERIPHERAL TIME OUT
008B     DNACK   =       $8B     ;DEVICE DOES NOT ACKNOWLEDGE COMMAND
008C     FRMERR  =       $8C     ;SERIAL BUS FRAMING ERROR
008D     CRSROR  =       $8D     ;CURSOR OVERRANGE
008E     OVRRUN  =       $8E     ;SERIAL BUS DATA OVERRUN
008F     CHKERR  =       $8F     ;SERIAL BUS CHECKSUM ERROR
         ;
0090     DERRER  =       $90     ;PERIPHERAL DEVICE ERROR (OPERATION NOT COMPLETED)
0091     BADMOD  =       $91     ;BAD SCREEN MODE NUMBER
0092     FNCNOT  =       $92     ;FUNCTION NOT IMPLEMENTED IN HANDLER
0093     SCRMEM  =       $93     ;INSUFICIENT MEMORY FOR SCREEN MODE
         ;
         ;
         ;
         ;
         ;       PAGE ZERO RAM ASSIGNMENTS
         ;
0000     LINZBS  =       $0000   ;LINBUG RAM (WILL BE REPLACED BY MONITOR RAM)
         ;
         ;       THESE LOCATIONS ARE NOT CLEARED
0002     CASINI  =       $0002   ;CASSETTE INIT LOCATION
0004     RAMLO   =       $0004   ;RAM POINTER FOR MEMORY TEST
0006     TRAMSZ  =       $0006   ;TEMPORARY REGISTER FOR RAM SIZE
0007     TSTDAT  =       $0007   ;RAM TEST DATA REGISTER
         ;
         ;       CLEARED ON COLD START ONLY
```

```
006B     BUFCNT  =       $006B   ;BUFFER COUNT
006C     BUFSTR  =       $006C   ;EDITOR GETCH POINTER
006E     BITMSK  =       $006E   ;BIT MASK
         ; LOTS OF RANDOM TEMPS
007B     SWPFLG  =       $007B   ;NON-0 IF TXT AND REGULAR RAM IS SWAPPED
007C     HOLDCH  =       $007C   ;CH IS MOVED HERE IN KGETCH BEfore CNTL & SHIFT PROC
         ;
         ;
         ;       80 - FF ARE FOR FP, USER, FMS AND DOS
         ;
         ;
         ;       PAGE 1 -- STACK
         ;
         ;
         ;       PAGE TWO RAM ASSIGNMENTS
         ;
0200     INTABS  =       $0200   ;INTERRUPT RAM
0200     VDSLST  =       $0200   ;DISPLAY LIST NMI VECTOR
0202     VPRCED  =       $0202   ;PROCEED LINE IRQ VECTOR
0204     VINTER  =       $0204   ;INTERRUPT LINE IRQ VECTOR
0206     VBREAK  =       $0206   ;SOFTWARE BREAK (00) INSTRUCTION IRQ VECTOR
0208     VKEYBD  =       $0208   ;POKEY KEYBOARD IRQ VECTOR
020A     VSERIN  =       $020A   ;POKEY SERIAL INPUT READY IRQ
020C     VSEROR  =       $020C   ;POKEY SERIAL OUTPUT READY IRQ
020E     VSEROC  =       $020E   ;POKEY SERIAL OUTPUT COMPLETE IRQ
0210     VTIMR1  =       $0210   ;POKEY TIMER 1 IRQ
0212     VTIMR2  =       $0212   ;POKEY TIMER 2 IRQ
0214     VTIMR4  =       $0214   ;POKEY TIMER 4 IRQ
0216     VIMIRQ  =       $0216   ;IMMEDIATE IRQ VECTOR
0218     CDTMV1  =       $0218   ;COUNT DOWN TIMER 1
021A     CDTMV2  =       $021A   ;COUNT DOWN TIMER 2
021C     CDTMV3  =       $021C   ;COUNT DOWN TIMER 3
021E     CDTMV4  =       $021E   ;COUNT DOWN TIMER 4
0220     CDTMV5  =       $0220   ;COUNT DOWN TIMER 5
0222     VVBLKI  =       $0222   ;IMMEDIATE VERTICAL BLANK NMI VECTOR
0224     VVBLKD  =       $0224   ;DEFERRED VERTICAL BLANK NMI VECTOR
0226     CDTMA1  =       $0226   ;COUNT DOWN TIMER 1 JSR ADDRESS
0228     CDTMA2  =       $0228   ;COUNT DOWN TIMER 2 JSR ADDRESS
022A     CDTMF3  =       $022A   ;COUNT DOWN TIMER 3 FLAG
022B     SRTIMR  =       $022B   ;SOFTWARE REPEAT TIMER
022C     CDTMF4  =       $022C   ;COUNT DOWN TIMER 4 FLAG
022E     CDTMF5  =       $022E   ;COUNT DOWN TIMER 5 FLAG
022F     SDMCTL  =       $022F   ;SAVE DMACTL REGISTER
0230     SDLSTL  =       $0230   ;SAVE DISPLAY LIST LOW BYTE
0231     SDLSTH  =       $0231   ;SAVE DISPLAY LIST HIGH BYTE
0232     SSKCTL  =       $0232   ;SKCTL REGISTER RAM
         ;
0234     LPENH   =       $0234   ;LIGHT PEN HORIZONTAL VALUE
0235     LPENV   =       $0235   ;LIGHT PEN VERTICAL VALUE
         ;
026F     GPRIOR  =       $26F    ;Global priority cell
         ;
         ;
         ;       POTENTIOMETERS
         ;
0270     PADDL0  =       $0270
0271     PADDL1  =       $0271
0272     PADDL2  =       $0272
0273     PADDL3  =       $0273
0274     PADDL4  =       $0274
0275     PADDL5  =       $0275
0276     PADDL6  =       $0276
0277     PADDL7  =       $0277
         ;
         ;
         ;       JOYSTICKS
         ;
         ;
0278     STICK0  =       $0278
0279     STICK1  =       $0279
027A     STICK2  =       $027A
027B     STICK3  =       $027B
         ;
         ;
         ;       PADDLE TRIGGER
         ;
         ;
027C     PTRIG0  =       $027C
027D     PTRIG1  =       $027D
027E     PTRIG2  =       $027E
027F     PTRIG3  =       $027F
0280     PTRIG4  =       $0280
0281     PTRIG5  =       $0281
0282     PTRIG6  =       $0282
0283     PTRIG7  =       $0283
         ;
         ;
         ;       JOYSTICK TRIGGER
         ;
         ;
0284     STRIG0  =       $0284
0285     STRIG1  =       $0285
0286     STRIG2  =       $0286
0287     STRIG3  =       $0287
         ;
         ;       Many random OS variables, the following were commented
```

```
         ;                       0-$FFFF (LSB, MSB) IN FR0,FR0+1->FR0
D92D     FPI     =       $D92D   ;FP -> INTEGER  FR0 -> FR0,FR0+1, CARRY
DA60     FSUB    =       $DA60   ;FR0 <- FR0 - FR1, CARRY
DA66     FADD    =       $DA66   ;FR0 <- FR0 + FR1  ,CARRY
DADB     FMUL    =       $DADB   ;FR0 <- FR0 * FR1  ,CARRY
DB28     FDIV    =       $DB28   ;FR0 <- FR0 / FR1  ,CARRY
DD89     FLD0R   =       $DD89   ;FLOATING LOAD REG0        FR0 <- (X,Y)
DD8D     FLD0P   =       $DD8D   ;                          FR0 <- (FLPTR)
DD98     FLD1R   =       $DD98   ;                          FR1 <- (X,Y)
DD9C     FLD1P   =       $DD9C   ;                          FR1 <- (FLPTR)
DDA7     FST0R   =       $DDA7   ;FLOATING STORE REG0  (X,Y) <- FR0
DDAB     FST0P   =       $DDAB   ;                  (FLPTR)<- FR0
DDB6     FMOVE   =       $DDB6   ;FR1 <- FR0
DD40     PLYEVL  =       $DD40   ;FR0 <- P(Z) = SUM(I = N TO 0) (A(I) *Z**I)  CARRY
         ;
         ;               INPUT:  (X,Y) = A(N), A(N-1)...A(0) -> PLYARG
         ;                       ACC   = # OF COEFFICIENTS = DEGREE + 1
         ;                       FR0   = Z
DDC0     EXP     =       $DDC0   ;FR0 <- E**FR0 = EXP10(FR0 * LOG10(E))  CARRY
DDCC     EXP10   =       $DDCC   ;FR0 <- 10**FR0 CARRY
DECD     LOG     =       $DECD   ;FR0 <- LN(FR0) = LOG10(FR0) / LOG10(E) CARRY
DED1     LOG10   =       $DED1   ;FR0 <- LOG10(FR0)       CARRY
         ;
         ;
         ;       THE FOLLOWING ARE IN THE BASIC CARTRIDGE:
         ;
         ;
BD81     SIN     =       $BD81   ;FR0 <- SIN(FR0) DEGFLG=0 => RADS, 6=>DEG. CARRY
         ;
BD73     COS     =       $BD73   ;FR0 <- COS(FR0)         CARRY
BD43     ATAN    =       $BD43   ;FR0 <- ATN(FR0)         CARRY
BEB1     SQR     =       $BEB1   ;FR0 <- SQUAREROOT(FR0) CARRY
         ;
         ;
         ;       FLOATING POINT ROUTINES ZERO PAGE (NEEDED ONLY IF F.P.
         ;               ROUTINES ARE CALLED)
00D4     FR0     =       $00D4   ;FP REG0
00E0     FR1     =       $00E0   ;FP REG1
00F2     CIX     =       $00F2   ;CURRENT INPUT INDEX
00F3     INBUFF  =       $00F3   ;POINTS TO USER'S LINE INPUT BUFFER
00FB     RADFLG  =       $00FB   ;0 = RADIANS, 6 = DEGREES
00FC     FLTPTR  =       $00FC   ;POINTS TO USERS FLOATING POINT NUMBER
         ;
         ;
         ;       FLOATING POINT ROUTINES' NON-ZP RAM
         ;
         ;       (057E to 05FF)
         ;
0580     LBUFF   =       $0580   ;LINE BUFFER
05E0     PLYARG  =       LBUFF+$60       ;POLYNOMILA ARGUMENTS
         ;
         ;
         ;
         ;
         ;       COLLEEN MNEMONICS
         ;
         ;
D200     POKEY   =       $D200   ;VBLANK ACTION:         DESCRIPTION:
D200     POT0    =       POKEY+0 ;POT0-->PADDL0          0-227 IN RAM CELL
D201     POT1    =       POKEY+1 ;POT1-->PADDL1          0-227 IN RAM CELL
D202     POT2    =       POKEY+2 ;POT2-->PADDL2          0-227 IN RAM CELL
D203     POT3    =       POKEY+3 ;POT3-->PADDL3          0-227 IN RAM CELL
D204     POT4    =       POKEY+4 ;POT4-->PADDL4          0-227 IN RAM CELL
D205     POT5    =       POKEY+5 ;POT5-->PADDL5          0-227 IN RAM CELL
D206     POT6    =       POKEY+6 ;POT6-->PADDL6          0-227 IN RAM CELL
D207     POT7    =       POKEY+7 ;POT7-->PADDL7          0-227 IN RAM CELL
D208     ALLPOT  =       POKEY+8 ;???
D209     KBCODE  =       POKEY+9
D20A     RANDOM  =       POKEY+10
D20B     POTGO   =       POKEY+11        ;Strobed
         ;               n/a
D20D     SERIN   =       POKEY+13
D20E     IRQST   =       POKEY+14
D20F     SKSTAT  =       POKEY+15
         ;
D200     AUDF1   =       POKEY+0
D201     AUDC1   =       POKEY+1
D202     AUDF2   =       POKEY+2
D203     AUDC2   =       POKEY+3
D204     AUDF3   =       POKEY+4
D205     AUDC3   =       POKEY+5
D206     AUDF4   =       POKEY+6
D207     AUDC4   =       POKEY+7
D208     AUDCT   =       POKEY+8 ;NONE                   AUDCTL<--[SIO]
D209     STIMER  =       POKEY+9
D20A     SKRES   =       POKEY+10        ;NONE           SKRES<--[SIO]
         ;
D20B     POTGO   =       POKEY+11
D20D     SEROUT  =       POKEY+13        ;NONE           SEROUT<--[SIO]
D20E     IRQEN   =       POKEY+14        ;POKMSK-->IRQEN (AFFECTED BY OPEN S: OR E:)
D20F     SKCTL   =       POKEY+15        ;SSKCTL-->SKCTL         SSKCTL<--[SIO]
         ;
D000     CTIA    =       $D000
D000     HPOSP0  =       CTIA+0
D001     HPOSP1  =       CTIA+1
```

COMPUTE! The Journal For Progressive Computing™

www.commodore.ca

```
        D002        HPOSP2  =     CTIA+2
        D003        HPOSP3  =     CTIA+3
        D004        HPOSM0  =     CTIA+4
        D005        HPOSM1  =     CTIA+5
        D006        HPOSM2  =     CTIA+6
        D007        HPOSM3  =     CTIA+7
        D008        SIZEP0  =     CTIA+8
        D009        SIZEP1  =     CTIA+9
        D00A        SIZEP2  =     CTIA+10
        D00B        SIZEP3  =     CTIA+11
        D00C        SIZEM   =     CTIA+12
        D00D        GRAFP0  =     CTIA+13
        D00E        GRAFP1  =     CTIA+14
        D00F        GRAFP2  =     CTIA+15
        D010        GRAFP3  =     CTIA+16
        D011        GRAFM   =     CTIA+17
        D012        COLPM0  =     CTIA+18  ;PCOLR0-->COLPM0      WITH ATTRACT
MODE
        D013        COLPM1  =     CTIA+19  ;ETC.N
        D014        COLPM2  =     CTIA+20
        D015        COLPM3  =     CTIA+21
        D016        COLPF0  =     CTIA+22
        D017        COLPF1  =     CTIA+23
        D018        COLPF2  =     CTIA+24
        D019        COLPF3  =     CTIA+25
        D01A        COLBK   =     CTIA+26
        D01B        PRIOR   =     CTIA+27
        D01C        VDELAY  =     CTIA+28
        D01D        GRACTL  =     CTIA+29
        D01E        HITCLR  =     CTIA+30
        D01F        CONSOL  =     CTIA+31  ;$08-->CONSOL         TURN OFF SPEA
KER
                    ;
        D000        M0PF    =     CTIA+0
        D002        M2PF    =     CTIA+2
        D003        M3PF    =     CTIA+3
        D004        P0PF    =     CTIA+4
        D005        P1PF    =     CTIA+5
        D006        P2PF    =     CTIA+6
        D007        P3PF    =     CTIA+7
        D008        M0PL    =     CTIA+8
        D009        M1PL    =     CTIA+9
        D00A        M2PL    =     CTIA+10
        D00B        M3PL    =     CTIA+11
        D00C        P0PL    =     CTIA+12
        D00D        P1PL    =     CTIA+13
        D00E        P2PL    =     CTIA+14
        D00F        P3PL    =     CTIA+15
        D010        TRIG0   =     CTIA+16          ;TRIG0-->STRIG1
        D011        TRIG1   =     CTIA+17          ;ETC.
        D012        TRIG2   =     CTIA+18
        D013        TRIG3   =     CTIA+19
        D014        PAL     =     CTIA+20
                    ;
        D400        ANTIC   =     $D400
        D400        DMACTL  =     ANTIC+0          ;DMACTL<--SDMCTL   ON OPEN S: OR
E:
        D401        CHARCTL =     ANTIC+1          ;CHACTL<--CHACT    ON OPEN S: OR
E:
        D402        DLISTL  =     ANTIC+2          ;DLISTL<--SDLSTL   ON OPEN S: OR
E:
        D403        DLISTH  =     ANTIC+3          ;DLISTH<--SDLSTH   ON OPEN S: OR
E:
        D404        HSCROL  =     ANTIC+4
        D405        VSCROL  =     ANTIC+5
        D407        PMBASE  =     ANTIC+7
        D409        CHBASE  =     ANTIC+9          ;CHBASE<--CHBAS    ON OPEN S: OR
E:
        D40A        WSYNC   =     ANTIC+10
        D40B        VCOUNT  =     ANTIC+11
        D40C        PENH    =     ANTIC+12
        D40D        PENV    =     ANTIC+13
        D40E        NMIEN   =     ANTIC+14         ;NMIEN<--40        POWER ON AND
[SETVBV]
        D40F        NMIRES  =     ANTIC+15         ;STROBED
        D40F        NMIST   =     ANTIC+15
                    ;
                    ;      Lots and lots of unofficial
                    ;      Mnemonics for display list instructions,
                    ;      as well as other bit patterns.
                    ;
                    ;      DL prefix implies display list
                    ;      instruction, naturally
                    ;
        0000        DLBL1   =     0       ; One blank line
        0010        DLBL2   =     $10     ; Two blank lines
        0020        DLBL3   =     $20     ; Three
        0030        DLBL4   =     $30     ; Four
        0040        DLBL5   =     $40     ; Five
        0050        DLBL6   =     $50     ; Six
        0060        DLBL7   =     $60     ; Seven
        0070        DLBL8   =     $70     ; Eight blank lines

        0080        DLINT   =     $80     ; Generate DisplayListInterrupt when this
                                          ; instruction is interpreted.
        0001        DLJMP   =     $1      ; Tells Antic Chip to jump to contents of
next two bytes
        0041        DLJVB   =     $41     ; Same as DLJMP but also halts ANTIC
'till next verticle blank
```

```
                                   ;        End of Atari Equates.
                                   ;
                                   ;    The rest of this is (C)1981 John H. Palevich
                                   ;
                                   ;
                1000               ORG    =    $1000       ;Must be at least 3K of memory
                                                           ;above this  point. . . .
                1000               .=ORG

                                               ;Once in the actual game you can make
                                               ;identical copies of it by pressing OPTION
                                               ;and SELECT down simultaniously between
                                               ;plays.  The Atari will Feep twice and you
                                               ;should press play & record and then RETURN
                                               ;to make a copy.  After one copy is made
                                               ;the game starts all over again.

           1000             PST=.
                                   ;
                                   ;    This is the Boot tape header table.
                                   ;

           1000   00               .BYTE   0                  ;Traditional
           1001   09               .BYTE   PND-PST+127/128  ;# of 128 byte blocks in program
           1002   00   10          .ADDR   PST                ;Start of place to load program
           1004   08   10          .ADDR   PINIT              ;Place to jump after loading
                                                              ;program

                                   ; ENTRY POINT FOR MULTI-STAGE BOOT PROCESS.

           1006   18               CLC
           1007   60               RTS

                                   ; ENTRY POINT FOR FIRST TIME INITIALIZATION
           1008   A9   3C          PINIT:  LDA     #$3C
           100A   8D   02   D3             STA     PACTL   ;turn off cassete motor

           100D   A9   16                  LDA     #RESTRT&$FF    ;Shove the restrt vector into
           100F   85   0A                  STA     DOSVEC         ;the DOS vector.
           1011   A9   10                  LDA     #RESTRT^
           1013   85   0B                  STA     DOSVEC+1
           1015   60                       RTS

                                   ;       WARMSTART ENTRY POINT
                                   ;
           1016   4C   4E   12     RESTRT: JMP BEGIN

                                   ;
                                   ; --------------------
                                   ; Look, up in the sky!  Its a helicopter, it's a plane, it's
                                   ; a saucer, no, it's Jack's wonderful 'SHOOT' game.
                                   ;

                                   ; Zero-Page-Variables
                                   ;(None of these variables Have to be on the zero page, but
                                   ;putting them here speeds up and shortens the code)
                                   ;
                 0080              DLIC    =     $80 ;Counter of display list interrupts
                 0081              DSCOR   =     $81 ;Delta score.
                 0082              MX      =     $82 ;Missile X coordinate
                 0083              MDX     =     $83 ;Missile X velocity
                 0084              MY      =     $84 ;Missile Y position (Zero means no missile
                                                     ;on the screen.)
                 0085              DSEC    =     $85 ;mod 15 counter for score to second conversio
n
                 0086              JIFF    =     $86 ;mod 60 counter for jiffy to second conversio
n
                 0087              STOP    =     $87 ;Non zero means ignore player's button.  Used

                                                     ;to keep him from firing unless we want him t
o.

                                   ;
                                   ; Data Tables
                                   ;
                                   ; MDLIST -- My display list. Consists of the standard-24-
blank-scan-lines,
                                   ; 8 scan lines of text (the score and copyright line)
                                   ; 11 sets of 16 scan lines with a display list interrupt
                                   ; occuring at the start of each pair
                                   ; and a final jump & wait for vertical blank instruction
                                   ;
           1019   70   70   70     MDLIST:  .BYTE   DLBL8,DLBL8,DLBL8
           101C   46                        .BYTE   DLLDM+DLPF6      ;C
           101D   00   18                   .ADDR   SCORLN
           101F   F0   70                   .BYTE   DLBL8+DLINT,DLBL8        ;1,2
           1021   F0   70                   .BYTE   DLBL8+DLINT,DLBL8        ;3,4
           1023   F0   70                   .BYTE   DLBL8+DLINT,DLBL8        ;5,6
           1025   F0   70                   .BYTE   DLBL8+DLINT,DLBL8        ;7,8
           1027   F0   70                   .BYTE   DLBL8+DLINT,DLBL8        ;9,10
           1029   F0   70                   .BYTE   DLBL8+DLINT,DLBL8        ;11,12
           102B   F0   70                   .BYTE   DLBL8+DLINT,DLBL8        ;13,14
           102D   F0   70                   .BYTE   DLBL8+DLINT,DLBL8        ;15,16
```

```
                           ;
                           ; JTAB -- missile's X velocity
                           ;       Used to convert from the joystick direction to
                           ; the speed and direction of the missile.  Used by VBLI.
                           ;

108E   00   01   FF        JTAB:    .BYTE   0,1,255,0
1091   00

                           ;
                           ; GUNTAB -- Picture of gun in all three
                           ; directions.
                           ;
                           ;........
                           ;......++
                           ;.....++.
                           ;....++..
                           ;...+++..
                           ;..++++..
                           ;.++++++.
                           ;++++++++
1092   00   03   06        GUNTAB:  .BYTE   0,3,6,$C,$1C,$3C,$7E,$FF
1095   0C   1C   3C
1098   7E   FF

                           ;........
                           ;++......
                           ;.++.....
                           ;..++....
                           ;..+++...
                           ;..++++..
                           ;.++++++.
                           ;++++++++
109A   00   C0   60                 .BYTE   0,$C0,$60,$30,$38,$3C,$7E,$FF
109D   30   38   3C
10A0   7E   FF

                           ;........
                           ;...++...
                           ;...++...
                           ;...++...
                           ;...++...
                           ;..++++..
                           ;.++++++.
                           ;++++++++
10A2   00   18   18                 .BYTE   0,$18,$18,$18,$18,$3C,$7E,$FF
10A5   18   18   3C
10A8   7E   FF

                           ;
                           ;  Pattern defn's.
                           ;
                           ;  Used by DPLANE to draw the planes.  Note that
                           ; the top two lines must be blank or else all sorts
                           ; of messy chopping effects will occur.
                           ;

                           ;
                           ; Pattern Zero is a helicopter flying left.
                           ;

                           ;........
                           ;........
                           ;+++++...
                           ;..+.....
                           ;++++..+.
                           ;+..++++.
                           ;+..+....
                           ;++++....
                           ;
10AA   00   00   F8        PTOL:    .BYTE   0,0,$F8,$20,$F2,$9E,$90,$F0
10AD   20   F2   9E
10B0   90   F0

                           ;........          helicopter flying right.
                           ;........
                           ;...+++++
                           ;.....+..
                           ;.+..++++
                           ;.++++..+
                           ;....+..+
                           ;....++++
10B2   00   00   1F                 .BYTE   0,0,$1F,4,$4F,$79,9,$F
10B5   04   4F   79
10B8   09   0F

                           ;........          airplane flying left.
                           ;........
                           ;.......+
                           ;....++.+
                           ;..++++++
                           ;.+++++++
                           ;...++...
                           ;........
10BA   00   00   01                 .BYTE   0,0,1,$0D,$3F,$7F,$18,0
10BD   0D   3F   7F
10C0   18   00

                           ;........          airplane flying right
                           ;........
```

COMPUTE! The Journal For Progressive Computing™

www.commodore.ca

```
        0040            DLLDM   =       $40     ; Tells Antic chip to start retrieving
data from memory referenced by the next t
        0020            DLVSCR  =       $20     ; (sort of) enables vertical scroll
        0010            DLHSCR  =       $10     ; enable horizontal scroll
                        ;
                        ;       Playfield Instructions
                        ;
        000F            DLPF15  =       15      ; 320 dots  2 colors  1 scan line
        000E            DLPF14  =       14      ; 160       4         1
        000D            DLPF13  =       13      ; 160       4         2
        000C            DLPF12  =       12      ; 160       2         1
        000B            DLPF11  =       11      ; 160       2         2
        000A            DLPF10  =       10      ; 80        4         4
        0009            DLPF9   =       9       ; 80        2         4
        0008            DLPF8   =       8       ; 40        4         8

                        ; Character PF types

        0007            DLPF7   =       7       ; 20 chars  5 colors  16 scan lines
        0006            DLPF6   =       6       ; 20        5         8
        0005            DLPF5   =       5       ; 40        5         16 (hairy 4 color cha
racters)
        0004            DLPF4   =       4       ; 40        5         8
        0003            DLPF3   =       3       ; 40        2         10 (Text processing)
        0002            DLPF2   =       2       ; 40        2         8  (Normal text mode)

                        ; PF1 & PF0 are special. . . .

                        ;
                        ;       Player & Missile Offsets
                        ;       Denoted by PM prefix
                        ;
        0024            PMLF    =       $24     ;Left side of screen
        00DD            PMRT    =       $DD     ;Right side of screen
        0180            PMDM    =       $180    ;Missile offset for double line resolution
        0200            PMDP0   =       $200    ;Player 0
        0280            PMDP1   =       $280    ;Player 1
        0300            PMDP2   =       $300    ;Player 2
        0380            PMDP3   =       $380    ;Player 3
        0300            PMSM    =       2*PMDM  ;Missile offset for single line resolution
        0400            PMSP0   =       2*PMDP0 ;Player 0
        0500            PMSP1   =       2*PMDP1 ;       1
        0600            PMSP2   =       2*PMDP2 ;       2
        0700            PMSP3   =       2*PMDP3 ;       3

                        ;
                        ;       Colors, denoted by CL
                        ;
        0000            CLGREY  =       0       ;grey
        0010            CLGOLD  =       $10     ;gold
        0020            CLORNG  =       $20     ;orange
        0030            CLRED   =       $30     ;red-orange
        0040            CLPINK  =       $40     ;pink
        0050            CLPURP  =       $50     ;purple
        0060            CLPURB  =       $60     ;purple blue
        0070            CLPBLU  =       $70     ;blue 1
        0080            CLBLUE  =       $80     ;blue 2
        0090            CLLBLU  =       $90     ;light blue
        00A0            CLTURQ  =       $A0     ;turquoise
        00B0            CLGBLU  =       $B0     ;green blue
        00C0            CLGREN  =       $C0     ;green
        00D0            CLYGRN  =       $D0     ;yellow green
        00E0            CLOGRN  =       $E0     ;orange green
        00F0            CLLTOR  =       $F0     ;light orange
                        ;       Bit Masks for console switches
                        ;
        0001            SWSTRT  =       $1      ;Start
        0002            SWSEL   =       $2      ;Select
        0004            SWOPT   =       $4      ;Option

                        ;       Joystick bit masks

        0008            JYR     =       8       ;Right
        0004            JYL     =       4       ;Left
        0002            JYB     =       2       ;Back
        0001            JYF     =       1       ;Forward

        D300            PIA     =       $D300
        D300            PORTA   =       PIA+0   ;PORTA-->STICK0,1     X-Y CONTROLLE
RS
        D301            PORTB   =       PIA+1   ;PORTB-->STICK2,3
        D302            PACTL   =       PIA+2   ;NONE                 PACTL<--3C [I
NIT]
        D303            PBCTL   =       PIA+3   ;NONE                 PBCTL<--3C [I
NIT]
        0030            ASCZER  =       '0      ;Ascii zero
        003A            COLON   =       $3A     ;Ascii Colon
        009B            EOL     =       $9B     ;End of Record

                        ;       The following are subroutines

        F385            PUTLIN  =       $F385   ;X -- Lo byte
                                                ;Y -- Hi byte
                                                ;of line

        E7D1            XVB     =       $E7D1   ;System VBLANK exit routine
                        ;
                        ;
                        ;
```

```
102F  F0  70            .BYTE   DLBL8+DLINT,DLBL8       ;17,18
1031  F0  70            .BYTE   DLBL8+DLINT,DLBL8       ;19,20
1033  F0  70            .BYTE   DLBL8+DLINT,DLBL8       ;21,22
1035  41                .BYTE   DLJVB
1036  19  10            .ADDR   MDLIST

                ;
                ; Message table

1038  28  43  29  TMSG: .ASCII"(C)1981 J H PALEVICH" ;Horrible bugs will infest the code
103B  31  39  38
103E  31  20  4A
1041  20  48  20
1044  50  41  4C
1047  45  56  49
104A  43  48

                                        ;of who-so-ever changes this copyright
                                        ;message to something else!

                ;
                ; The next many tables are for the display list interrupt handler.
                ; They are all 11 bytes long, so that there is one byte in each
                ; table for each display list interrupt.  The first DLI uses the
                ; first byte in the table, and so forth, so in the TCOLB table,
                ; for instance, the top of the sky is black, then the first DLI
                ; makes it turn CLBLUE, the second CLLBLU, and so on.
                ;

                ;
                ; TCOLB - table of colors for backround
                ;  The sky starts out black and turns light blue, then
                ; the backround turns green for the 'ground' under the gun.

104C  80  90  82  TCOLB:  .BYTE   CLBLUE,CLLBLU,CLBLUE+2
104F  92  84  94          .BYTE   CLLBLU+2,CLBLUE+4,CLLBLU+4
1052  86  96  88          .BYTE   CLBLUE+6,CLLBLU+6,CLBLUE+8
1055  98  C8              .BYTE   CLLBLU+8,CLGREN+8

                ;
                ; TCOLP - COLOR PLAYER ZERO
                ; Player zero is the planes in the sky and the gun on the ground
                ;

1057  08  18      TCOLP:  .BYTE   CLGREY+8,CLGOLD+8
1059  28  38              .BYTE   CLORNG+8,CLRED+8
105B  48  58              .BYTE   CLPINK+8,CLPURP+8
105D  68  78              .BYTE   CLPURB+8,CLPBLU+8
105F  80  38  38          .BYTE   CLBLUE,CLRED+8,CLRED+8

                ;
                ; TPX -- Players X position
                ;       The gun is always stuck at 124.
                ;       DPLANE takes care of the planes' x position,
                ;       DLIH sets the X position to zero to kill the
                ; plane and get it off the screen.
                ;

1062  00  01  02  TPX:    .BYTE   0,1,2,3,4,5,6,7,8,124,124
1065  03  04  05
1068  06  07  08
106B  7C  7C

                ;
                ; TDX -- Player's velocity.
                ;       The gun is always zero, so it will not drift off
                ; to either side.
                ;

106D  01  02  03  TDX:    .BYTE   1,2,3,2,1,255,254,253
1070  02  01  FF
1073  FE  FD
1075  FE  00  00          .BYTE   254,0,0

                ;
                ; TVAL -- PLAYER'S POINT-VALUE
                ;
                ;       Set by DPLANE
                ;       Used by DLIH
                ;

1078  01  02  03  TVAL:   .BYTE   1,2,3,4,5,6,7,8,9,10,11
107B  04  05  06
107E  07  08  09
1081  0A  0B

                ;
                ; TWID -- width of the player
                ;
                ;       0 -- normal (two GR. 0 characters) wide
                ;       1 -- double (four Gr. 0 characters) wide
                ;       The gun is double width
                ;

1083  00  01  00  TWID:   .BYTE   0,1,0,1,0,1,0,1,0,1,1
1086  01  00  01
1089  00  01  00
108C  01  01

                ;
                ; That's the end of the tables used by the DLIH
                ; routine.
                ;
```

```
                        ;+.......
                        ;+.++....
                        ;++++++..
                        ;+++++++.
                        ;...++...
                        ;........
10C2  00  00  80                .BYTE   0,0,$80,$80,$FC,$FE,$18,0
10C5  80  FC  FE
10C8  18  00

                        ;........       saucer flying left, right, hovering, etc.
                        ;...++...
                        ;..+..+..
                        ;.++++++.
                        ;+......+
                        ;.++++++.
                        ;........
                        ;........
10CA  00  18  24                .BYTE   0,$18,$24,$7E,$81,$7E,0,0
10CD  7E  81  7E
10D0  00  00

                        ;
                        ; TPLANE DX, VAL, WID, PAT table
                        ;
                        ;       TPLANE is a table used by DPLANE to
                        ; create planes.  What DPLANE does is pick a random number
                        ; between zero and five and use that number to fill in the
                        ; TDX, TVAL, TWID and player zero slots for that kind of plane.
                        ;
                        ;       There is no real reason to stop with only six
                        ; kinds of planes. . . .
                        ;

10D2  01  05  00        TPLANE: .BYTE   1,5,0,8         ;Helicopter, left
10D5  08
10D6  FF  05  00                .BYTE   $FF,5,0,0       ; "         ", right
10D9  00
10DA  02  0A  01                .BYTE   2,10,1,24       ;Airplane, left
10DD  18
10DE  FE  0A  01                .BYTE   $FE,10,1,16 ; " " , right
10E1  10
10E2  03  19  00                .BYTE   3,25,0,32       ;Saucer, left
10E5  20
10E6  FD  19  00                .BYTE   $FD,25,0,32 ;Saucer, right
10E9  20

                        ; Player missile table equates
                        ;
      1800              MPMBAS  =       ORG+$800        ;Player missile base --
                                                        ;Leave 2K for the main program

      1800              SCORLN  =       MPMBAS          ;Score line (uses 20 bytes)

      0060              GUNOFF  =       96              ;Gun offset -- Y position of top
                                                        ;of gun.

      1A60              GUNPOS  =       MPMBAS+PMDP0+GUNOFF ;Memory location of
                                                        ;top of gun in player0 memory.

      1800              SCORE   =       SCORLN          ;Address of Score (six digets)
      1807              HISCOR  =       SCORLN+7        ;Address of High score (six digets)
      180E              TIME    =       SCORLN+14       ;Address of Time remaining (six digets)

                        ;
                        ;
                        ; This is the DLI handler --
                        ;       it is called every time the Antic chip fetches a display
                        ; list byte with the high bit set.  It saves the A and X registers,
                        ; changes the sky & color & width & position of the next plane,
                        ; and checks to see if the previous plane has been hit.  If so, it
                        ; moves that plane off the screen and adds the point value of that
                        ; plane to the DSCOR variable.  Then it restores X & A and returns.
                        ;
                        ;

10EA  48                DLIH:   PHA                     ;Save A
10EB  8A                        TXA
10EC  48                        PHA                     ;Save X
10ED  A6  B0                    LDX     DLIC            ;Change the sky's color as fast as
10EF  E8                        INX                     ;we can,
10F0  BD  4C  10                LDA     TCOLB,X         ;but,
10F3  8D  0A  D4                STA     WSYNC           ;wait untill we can do it without
10F6  8D  1A  D0                STA     COLBK           ;the user noticing it.

10F9  A6  B0                    LDX     DLIC
10FB  AD  08  D0                LDA     MOPL            ;Did missile 0 hit anything?
10FE  29  01                    AND     #1              ;like P0?
1100  F0  13                    BEQ     DLIH2           ;No.
1102  A9  00                    LDA     #0              ;Yes:
1104  9D  62  10                STA     TPX,X           ;So zero x position,
1107  9D  6D  10                STA     TDX,X           ;and x velocity,
110A  BD  78  10                LDA     TVAL,X          ;and add the value of
110D  18                        CLC                     ;that player to DSCOR
110E  65  B1                    ADC     DSCOR
1110  85  B1                    STA     DSCOR
1112  8D  1E  D0                STA     HITCLR          ;And clear the hit registers.

1115  E8                DLIH2:  INX                     ;In any event, update:
1116  86  B0                    STX     DLIC
1118  BD  62  10                LDA     TPX,X           ;plane's X position.
```

COMPUTE! The Journal For Progressive Computing™

www.commodore.ca

```
111B  18              CLC
111C  7D  6D  10      ADC   TDX,X
111F  9D  62  10      STA   TPX,X

1122  8D  00  D0      STA   HPOSP0
1125  BD  57  10      LDA   TCOLP,X   ;plane's color
1128  8D  12  D0      STA   COLPM0
112B  BD  83  10      LDA   TWID,X    ;plane's width
112E  8D  08  D0      STA   SIZEP0
1131  68              PLA             ;restore X
1132  AA              TAX
1133  68              PLA             ;restore A
1134  40              RTI             ;and return.

                ;
                ; VBLH -- vertical blank interrupt
                ;       gets called 60 times a second when the TV gun has just
                ; retraced the screen and the Antic chip is starting to read the
                ; display list all over again.
                ;
                ;       It updates the player's score & time-left counter,
                ; updates the missile if it is in flight, fires it if it is not
                ; in flight and if the joystick's button is pressed, resets the
                ; counter that the display list interrupt handler uses to tell
                ; where it is, and kills the missile if it hit a plane or went
                ; off the top of the screen.
                ;

1135  A5  B1      VBLH:   LDA   DSCOR     ;Check if there are any unscored
1137  D0  08              BNE   VBL5      ;points left. Yes.
1139  A9  80              LDA   #$80      ;No.
113B  8D  03  D2          STA   AUDC2     ;Turn off point sound
113E  4C  90  11          JMP   VBL7

1141  38          VBL5:   SEC             ;Subtract one from the unscored
1142  E9  01              SBC   #1        ;points counter.
1144  85  B1              STA   DSCOR

1146  A9  8A              LDA   #$8A      ;Turn on the point sound.
1148  8D  03  D2          STA   AUDC2

114B  A2  05              LDX   #5        ;Add one to the SCORE counter.
114D  BD  00  18  VBL6:   LDA   SCORE,X   ;Which by an amazing coincidence
1150  18                  CLC             ;is in a human readable form.
1151  69  01              ADC   #1
1153  09  10              ORA   #$10      ;If it was a space make it a '1'.
1155  9D  00  18          STA   SCORE,X
1158  C9  1A              CMP   #$1A      ;Greater than a '9'?
115A  D0  09              BNE   VBLH5     ;No.
115C  A9  10              LDA   #$10      ;Yes. Set to '0' and add one to
115E  9D  00  18          STA   SCORE,X   ;the next most signifigant diget.
1161  CA                  DEX
1162  4C  4D  11          JMP   VBL6
1165  A5  B7      VBLH5:  LDA   STOP      ;No free time if
1167  D0  27              BNE   VBL7      ;we're STOPed.

1169  A6  B5              LDX   DSEC      ;Has he scored 15 points in a row
116B  E8                  INX             ;yet?
116C  86  B5              STX   DSEC
116E  E0  0F              CPX   #15
1170  D0  1E              BNE   VBL7      ;Nope.
1172  A2  00              LDX   #0        ;Yes.  Reset this counter, then, and
1174  86  B5              STX   DSEC

1176  A2  05              LDX   #5        ;give this man a second of free time.
1178  BD  0E  18  VBL8:   LDA   TIME,X    ;(this routine is just like the VBL6
117B  18                  CLC             ;one, only it fiddles with numbers that
117C  69  01              ADC   #1        ;are a different color. . . .)
117E  09  90              ORA   #$90
1180  9D  0E  18          STA   TIME,X
1183  C9  9A              CMP   #$9A
1185  90  09              BCC   VBL7
1187  A9  90              LDA   #$90
1189  9D  0E  18          STA   TIME,X
118C  CA                  DEX
118D  4C  78  11          JMP   VBL8

1190  A6  B6      VBL7:   LDX   JIFF      ;Have 60 jiffies (a jiffy is used
1192  E8                  INX             ;by Pet owners and other people
1193  86  B6              STX   JIFF      ;to denote a 60th of a second) elapsed?
1195  E0  3C              CPX   #60
1197  D0  22              BNE   VBL9      ;No!
1199  A2  00              LDX   #0        ;Yes.  Reset jiffy counter
119B  86  B6              STX   JIFF

119D  A5  B7              LDA   STOP      ;Has time stopped?
119F  D0  2E              BNE   VBL12     ;Yes.

11A1  A2  05              LDX   #5        ;No, so we shall take away a second from
11A3  BD  0E  18  VBL10:  LDA   TIME,X    ;the user's time (har har har)
11A6  38                  SEC             ;in just the same way as we added one,
11A7  E9  01              SBC   #1        ;except for a few changes.
11A9  09  90              ORA   #$90
11AB  9D  0E  18          STA   TIME,X

11AE  C9  9F              CMP   #$9F      ;Like this check for borrow in place
11B0  D0  09              BNE   VBL9      ;of a check for carry.
11B2  A9  99              LDA   #$99
11B4  9D  0E  18          STA   TIME,X
```

```
                ;
                ; Main program starts here!!!
                ;

124E  A9  A8      BEGIN:  LDA   #$A8      ;Missile sound is pure at volume 8
1250  8D  01  D2          STA   AUDC1
1253  A9  80              LDA   #$80      ;Score sound is a silent fuzz.
1255  8D  03  D2          STA   AUDC2

1258  A9  00              LDA   #0        ;missile frequency is ultrasonic
125A  8D  00  D2          STA   AUDF1

125D  A9  30              LDA   #$30      ;score frequency is a high fuzz.
125F  8D  02  D2          STA   AUDF2

                ;
                ; Erase player missile memory space.
                ;

1262  A2  80              LDX   #128
1264  A9  00              LDA   #0
1266  9D  FF  19  CLOOP:  STA   MPMBAS+PMDP0-1,X   ;We only use Player zero
1269  9D  7F  19          STA   MPMBAS+PMDM-1,X    ;and the missiles.
126C  CA                  DEX
126D  D0  F7              BNE   CLOOP

126F  A9  00              LDA   #0        ;Move all the players and missiles
1271  A2  08              LDX   #8        ;off the screen.
1273  9D  FF  CF  PLOOP:  STA   HPOSP0-1,X
1276  CA                  DEX
1277  D0  FA              BNE   PLOOP

1279  A9  2E              LDA   #$2E      ;Enable PM DMA and a normal playfield.
127B  8D  2F  02          STA   SDMCTL
127E  A9  18              LDA   #MPMBAS^  ;Set up the pointer to the Player
1280  8D  07  D4          STA   PMBASE    ;missile defuns.
1283  A9  03              LDA   #3        ;Tell the CTIA to expect PM DMA
1285  8D  1D  D0          STA   GRACTL

1288  A9  10              LDA   #$10      ;Enable fifth player so the
128A  0D  6F  02          ORA   GPRIOR    ;missiles will be COLOR3
128D  8D  6F  02          STA   GPRIOR
1290  8D  1B  D0          STA   PRIOR

1293  A9  00              LDA   #0        ;Zero the missile
1295  85  B4              STA   MY

1297  A9  01              LDA   #1        ;Stop the player from fireing.
1299  85  B7              STA   STOP

129B  A9  40              LDA   #$40      ;Disable DLI's
129D  8D  0E  D4          STA   NMIEN
12A0  A9  10              LDA   #MDLIST^  ;Set up the vector to our
12A2  8D  31  02          STA   SDLSTH    ;display list.
12A5  A9  19              LDA   #MDLIST&$FF
12A7  8D  30  02          STA   SDLSTL
12AA  A9  10              LDA   #DLIH^    ;Set up the vector to our
12AC  8D  01  02          STA   VDBLST+1  ;DLI handler
12AF  A9  EA              LDA   #DLIH&$FF
12B1  8D  00  02          STA   VDBLST
12B4  A2  11              LDX   #VBLH^    ;Load the address of our
12B6  A0  35              LDY   #VBLH&$FF ;verticle blank interrupt
12B8  A9  06              LDA   #SETIMM   ;handler into X & Y and call
12BA  20  5C  E4          JSR   SETVBV    ;SETVBV to set up the VBLI vector.

12BD  A9  C0              LDA   #$C0      ;Enable DLI's
12BF  8D  0E  D4          STA   NMIEN

12C2  A9  C6              LDA   #CLGREN+6 ;Color 0 (score) is green
12C4  8D  C4  02          STA   COLOR0

12C7  A9  36              LDA   #CLRED+6  ;Color 1 (high score) is red.
12C9  8D  C5  02          STA   COLOR1

12CC  A9  18              LDA   #CLGOLD+8 ;Color 2 (time) is gold.
12CE  8D  C6  02          STA   COLOR2

12D1  A9  0A              LDA   #CLGREY+10 ;Color 3 (missiles) is white.
12D3  8D  C7  02          STA   COLOR3

                ;
                ; Write out the copyright message.
                ;

12D6  A2  14              LDX   #20       ;For the twenty charaters in the message,
12D8  BD  37  10  COPYR1: LDA   TMSG-1,X  ;load a character,
12DB  20  0E  14          JSR   ASTOIN    ;Call ASTOIN to get it to internal
12DE  09  C0              ORA   #$C0      ;form, change it to color 3, and
12E0  9D  FF  17          STA   SCORLN-1,X ;store it on the play field.
12E3  CA                  DEX
12E4  D0  F2              BNE   COPYR1
12E6  A5  13              LDA   RTCLOC+1  ;get the real time (4 second ticks)
12E8  18                  CLC
12E9  69  03              ADC   #3        ;8 to 12 seconds of glory

12EB  C5  13      COPYW:  CMP   RTCLOC+1  ;Wait a while 'till user's read it.
12ED  D0  FC              BNE   COPYW
```

```
13B2  18                  CLC
13B3  65  B1              ADC   DSCOR
13B5  85  B1              STA   DSCOR
13B7  4C  4D  13          JMP   REDRAW    ;and a new round.

13BA  A5  B7      MAIN4:  LDA   STOP      ;Check if player is stopped.
13BC  F0  DC              BEQ   MAIN      ;Nope.
                                          ;Yes:
                ;
                ; Player died, so update high score
                ; and restart.
                ;

138E  A5  13              LDA   RTCLOC+1  ;Wait 4 to 8 seconds
1390  18                  CLC
1391  69  02              ADC   #2
1393  C5  13      HIW:    CMP   RTCLOC+1  ;for points to drain.
1395  D0  FC              BNE   HIW

1397  A2  00              LDX   #0        ;Check his high score against
1399  BD  07  18  HICHK:  LDA   HISCOR,X  ;his current score.
139C  29  1F              AND   #$1F      ;If his high score is anywhere
139E  DD  00  18          CMP   SCORE,X   ;less than his current score
13A1  F0  05              BEQ   HICHK1    ;Equal
13A3  B0  08              BCS   NOHI      ;HI > score. Stop.
13A5  4C  B0  13          JMP   NEWHI     ;HI < score.
13A8  E8          HICHK1: INX             ;NOTE: We check from left to right,
13A9  E0  06              CPX   #6        ;unlike the routines in VBIH.
13AB  D0  EC              BNE   HICHK
13AD  4C  08  13  NOHI:   JMP   REPEAT    ;So go and restart the game.

                ;
                ; Copy new high score

13B0  A2  06      NEWHI:  LDX   #6        ;Copy all six digets of the
13B2  BD  FF  17  NEWHI1: LDA   SCORE-1,X ;score into the high score,
13B5  29  1F              AND   #$1F      ;changing the color at the
13B7  09  40              ORA   #$40      ;same time.
13B9  9D  06  18          STA   HISCOR-1,X
13BC  CA                  DEX
13BD  D0  F3              BNE   NEWHI1
13BF  4C  08  13          JMP   REPEAT    ;Then restart the game.

                ;
                ; DPLANE -- pick and draw a plane
                ;

13C2  00          XTEMP:  .BYTE 0         ;Temporary variables
13C3  00          YTEMP:  .BYTE 0         ;Used to save state of
13C4  00          TEMPA:  .BYTE 0         ;the processer.

13C5  8E  C2  13  DPLANE: STX   XTEMP     ;Save X
13C8  8C  C3  13          STY   YTEMP     ;Save Y
13CB  AD  0A  D2  DPLAN1: LDA   RANDOM    ;Get a random number
13CE  29  07              AND   #7        ;reduce it to 0-7
13D0  C9  06              CMP   #6        ;Is it > 6 ?
13D2  B0  F7              BCS   DPLAN1    ;Yes.  Pick another one.
13D4  0A                  ASL   A         ;No.  Multiply by 4
13D5  0A                  ASL   A
13D6  AA                  TAX
13D7  BD  D2  10          LDA   TPLANE,X  ;And use it to index into
13DA  99  6D  10          STA   TDX,Y     ;The tplane table for
13DD  BD  D3  10          LDA   TPLANE+1,X ;the velocity, the value,
13E0  99  78  10          STA   TVAL,Y    ;the width, and the
13E3  BD  D4  10          LDA   TPLANE+2,X ;index into the plane
13E6  99  83  10          STA   TWID,Y    ;picture table.
13E9  A9  00              LDA   #0
13EB  99  62  10          STA   TPX,Y     ;Set the X position to 0.
13EE  BD  D5  10          LDA   TPLANE+3,X ;Get the index into the
13F1  AA                  TAX             ;picture table.
13F2  AC  C2  13          LDY   XTEMP
13F5  A9  08              LDA   #8
13F7  8D  C4  13          STA   TEMPA

13FA  BD  AA  10  DPLAN3: LDA   PTOL,X    ;Copy eight bytes from
13FD  99  00  1A          STA   MPMBAS+PMDP0,Y ;the picture table to
1400  E8                  INX             ;the player's defun space.
1401  C8                  INY
1402  CE  C4  13          DEC   TEMPA
1405  D0  F3              BNE   DPLAN3
1407  98                  TYA             ;Restore the processor
1408  AA                  TAX             ;state, and return
1409  AC  C3  13          LDY   YTEMP     ;(X has had 8 added to it
140C  60                  RTS             ;so that the next plane
                                          ;drawn will be one position
                                          ;lower in the player's defun).

                ;
                ; Ascii character in A --> Screen internal character in A
                ; state saved and restored.
                ;
                ; (slightly modified from the internal
                ; Atari OS routine & uses ROM tables)

140D  00          ASTEMP: .BYTE 0         ;Used to save the state of Y

140E  8C  0D  14  ASTOIN: STY   ASTEMP    ;Save Y.
1411  A8                  TAY             ;Copy A into Y
1412  8A                  TXA
1413  48                  PHA             ;Save X.
1414  98                  TYA
```

COMPUTE! The Journal For Progressive Computing

www.commodore.ca

```
11B7  CA                DEX
11B8  4C  A3  11        JMP     VBL10

11BB  A9  00    VBL9:   LDA     #0          ;Has the user run out of time??
11BD  A2  06            LDX     #6          ;If so, OR-ing together all of his
11BF  1D  00  18 VBL11: ORA     TIME-1,X    ;time-left digets should give a zero
11C2  CA                DEX
11C3  D0  FA            BNE     VBL11

11C5  29  0F            AND     #$F
11C7  C9  00            CMP     #0          ;Does it?
11C9  D0  04            BNE     VBL12       ;No.
11CB  A9  01            LDA     #1          ;Yes.  Stop time!!!
11CD  85  B7            STA     STOP

11CF  A9  00    VBL12:  LDA     #0          ;Store a zero into the ATRACT flag
11D1  85  4D            STA     ATRACT      ;to keep the Atari from futzing with
                                            ;our screen colors. . . .  Of course
                                            ;this means that the user might end
                                            ;up with the game field permanently
                                            ;embossed on his TV screen. . . .

11D3  AD  78  02        LDA     STICK0      ;Take STICK0
11D6  4A                LSR     A
11D7  4A                LSR     A           ;Divide it by 4
11D8  AA                TAX                 ;Use that number to look up the
11D9  BD  8E  10        LDA     JTAB,X      ;direction the missile should
11DC  85  B3            STA     MDX         ;travel in.
11DE  CA                DEX                 ; Then subtract one from that
11DF  8A                TXA                 ; (it better not be 0. . . .)
11E0  0A                ASL     A           ;And multiply by eight
11E1  0A                ASL     A           ;to get an index into the
11E2  0A                ASL     A           ;table of the gun pictures
11E3  AA                TAX
11E4  A0  00            LDY     #0
11E6  BD  92  10 GUNDLP: LDA    GUNTAB,X    ;Copy the picture of the gun
11E9  99  60  1A        STA     GUNPOS,Y    ;into player zero.  Use two
11EC  C8                INY                 ;bytes for each byte in the
11ED  99  60  1A        STA     GUNPOS,Y    ;piture table so the gun is
11F0  E8                INX                 ;16 dots (32 scan lines) high.
11F1  C8                INY
11F2  C0  10            CPY     #16
11F4  D0  F0            BNE     GUNDLP

11F6  A5  B2            LDA     MX          ;Now update missile's X position.
11F8  18                CLC
11F9  65  B3            ADC     MDX
11FB  85  B2            STA     MX
11FD  8D  04  D0        STA     HPOSM0

1200  A5  B4            LDA     MY          ;Missile Y
1202  F0  26            BEQ     VCONT       ;No missile.
1204  AA                TAX
1205  A9  00            LDA     #0          ;Erase old missile
1207  9D  80  19        STA     MPMBAS+PMDM,X

120A  CA                DEX                 ;Hit top of screen?
120B  F0  11            BEQ     VMDIE       ;Yes.
                                            ;No.
120D  A5  B1            LDA     DSCOR       ;Hit an airplane with the missile?
120F  D0  12            BNE     VMHIT       ;No.
1211  86  B4            STX     MY
1213  A9  FF            LDA     #$FF        ;Draw new missile

1215  9D  80  19        STA     MPMBAS+PMDM,X

1218  8E  00  D2        STX     AUDF1       ;fweep sound effect
121B  4C  2A  12        JMP     VCONT

121E  86  B4    VMDIE:  STX     MY          ;Zero the missile's Y coordinate
1220  4C  2A  12        JMP     VCONT       ;to kill the missile.

1223  A2  00    VMHIT:  LDX     #0          ;Since we hit something we should
1225  8E  00  D2        STX     AUDF1       ;silence the sound register
1228  86  B4            STX     MY          ;and zero the missile.

122A  A5  B7    VCONT:  LDA     STOP        ;Stoped?
122C  D0  16            BNE     VCONT2      ;Yes.

122E  AD  84  02        LDA     STRIG0      ;Check if human wants to fire
1231  D0  11            BNE     VCONT2      ;No.

1233  A5  B4            LDA     MY          ;Check if he CAN fire.
1235  D0  0D            BNE     VCONT2      ;Can't

1237  A9  62            LDA     #GUNOFF+2   ;Set the Y coordinate to just
1239  85  B4            STA     MY          ;above the muzzle of the gun.
123B  A5  B3            LDA     MDX         ;To get the X coordinate,
123D  0A                ASL     A           ;multiply MDX (the direction the
123E  0A                ASL     A           ;gun is pointing) by 4
123F  18                CLC                 ;and add to 132 (which is the
1240  69  84            ADC     #132        ;CENTER of the Gun)
1242  85  B2            STA     MX

1244  A9  FF    VCONT2: LDA     #$FF        ;Reset DLI counter
1246  85  B0            STA     DLIC
1248  8D  1E  D0        STA     HITCLR      ;Zero hits.
124B  4C  D1  E7        JMP     XVB         ;Jump to the OS's exit vblank routine.
```

```
                ;
                ; Erase & initialize score line
                ;
12EF  A2  14            LDX     #20         ;Fill score line with spaces.
12F1  A9  00            LDA     #0
12F3  9D  FF  17 ERASES: STA    SCORLN-1,X
12F6  CA                DEX
12F7  D0  FA            BNE     ERASES

12F9  A9  10            LDA     #$10        ;Make score '     0'
12FB  8D  05  18        STA     SCORE+5

12FE  A9  50            LDA     #$50        ;Make high score '     0'
1300  8D  0C  18        STA     HISCOR+5

1303  A9  90            LDA     #$90        ;Make time '     0'
1305  8D  13  18        STA     TIME+5

1308  A9  01    REPEAT: LDA     #1          ;Stop the player (just to make
130A  85  B7            STA     STOP        ;sure!)
130C  A9  08            LDA     #$08        ;Set up to read the consol
130E  8D  1F  D0        STA     CONSOL      ;switches

1311  AD  1F  D0 WAIT:  LDA     CONSOL
1314  C9  01            CMP     #SWSTRT     ;The other two switches -- Select
1316  D0  06            BNE     WAIT1       ;and option, are not pressed down.

1318  20  2A  14        JSR     MAKETP      ;If they are pressed down, make a
                                            ;copy of this whole program
131B  4C  4E  12        JMP     BEGIN       ;And reset since the sound registers
                                            ;will be messed up. . . .

131E  C9  06    WAIT1:  CMP     #6          ;Is the start switch pressed?
1320  D0  EF            BNE     WAIT        ;Nope. . . .

1322  A9  00            LDA     #0          ;Yes.  Start game (!)
1324  A2  06            LDX     #6          ;Erase time and score but not
1326  9D  00  18 RESTR2: STA    TIME-1,X    ;high score.
1329  9D  FF  17        STA     SCORE-1,X
132C  CA                DEX
132D  D0  F7            BNE     RESTR2

132F  A9  91            LDA     #$91        ;Set the time left to '   120'
1331  8D  11  18        STA     TIME+3
1334  A9  92            LDA     #$92
1336  8D  12  18        STA     TIME+4
1339  A9  90            LDA     #$90
133B  8D  13  18        STA     TIME+5

133E  A9  10            LDA     #$10        ;Set the score to '     0'
1340  8D  05  18        STA     SCORE+5

1343  A9  00            LDA     #0          ;Let the player shoot.
1345  85  B7            STA     STOP
1347  85  B1            STA     DSCOR       ;Clear out the vblank
1349  85  B6            STA     JIFF        ;counters.
134B  85  B5            STA     DSEC

                ;
                ;       Set up the PM graphics.
                ;

134D  A2  18    REDRAW: LDX     #$18        ;Start at line 18.
134F  A0  00            LDY     #0
1351  20  C5  13 CLOOP1: JSR    DPLANE      ;Draw 8 planes.
1354  C8                INY
1355  C0  08            CPY     #8
1357  D0  F8            BNE     CLOOP1

                ;
                ; Set up 30 second count down timer (used to time rounds)
                ;
1359  A2  07            LDX     #7          ;30 secs = 1800 jiffies.
135B  A0  D0            LDY     #208
135D  A9  03            LDA     #3          ;CDT # 3.
135F  8D  2A  02        STA     CDTMF3
1362  20  5C  E4        JSR     SETVBV

1365  A9  C0            LDA     #$C0        ;Re-enable DLI's
1367  8D  0E  D4        STA     NMIEN

136A  AD  2A  02 MAIN:  LDA     CDTMF3      ;Main loop -- if 30 seconds are
136D  D0  03            BNE     MAIN2       ;up, draw another wave of planes.
136F  4C  4D  13        JMP     REDRAW

1372  A0  08    MAIN2:  LDY     #8          ;Check if all the planes have
1374  A9  00            LDA     #0          ;been shot down (i.e. their
1376  19  6C  10 MAIN3: ORA     TDX-1,Y     ;velocities are all zero)
1379  88                DEY
137A  D0  FA            BNE     MAIN3
137C  C9  00            CMP     #0
137E  D0  0A            BNE     MAIN4

1380  A9  32            LDA     #50         ;If so, player gets 50 points
```

```
1415  2A                ROL     A           ;Get bits 6 & 5 of character
1416  2A                ROL     A
1417  2A                ROL     A
1418  2A                ROL     A
1419  29  03            AND     #3
141B  AA                TAX
141C  98                TYA
141D  29  9F            AND     #$9F        ;Zero bits 6 & 5 of character
141F  1D  F6  FE        ORA     $FEF6,X     ;and fill them with the bits
                                            ;from the ROM table, giving
                                            ;the internal (screen) code.

1422  A8                TAY                 ;Restore X & Y registers.
1423  68                PLA
1424  AA                TAX
1425  98                TYA
1426  AC  0D  14        LDY     ASTEMP
1429  60                RTS

                ;
                ;       Boot tape writer.  Writes out a Boot tape and returns.
                ;

142A  A2  20    MAKETP: LDX     #ICTWO      ;Choose IOCB two
142C  A9  0C            LDA     #CLOSE      ;Close it.
142E  9D  42  03        STA     ICCOM,X
1431  20  56  E4        JSR     CIOV

1434  A9  14            LDA     #CAS^       ;Open the C: device.
1436  9D  45  03        STA     ICBAH,X
1439  A9  75            LDA     #CAS&$FF
143B  9D  44  03        STA     ICBAL,X
143E  A9  03            LDA     #OPEN
1440  9D  42  03        STA     ICCOM,X
1443  A9  08            LDA     #OPNOT      ;for output
1445  9D  4A  03        STA     ICAX1,X
1448  A9  80            LDA     #$80        ;short IRQ
144A  9D  4B  03        STA     ICAX2,X
144D  20  56  E4        JSR     CIOV

1450  A9  00            LDA     #PST&$FF    ;Write out the program in
1452  9D  44  03        STA     ICBAL,X     ;one Fell Swoop (using a
1455  A9  10            LDA     #PST^       ;block putchar)
1457  9D  45  03        STA     ICBAH,X
145A  A9  78            LDA     #<PND-PST>&$FF
145C  9D  48  03        STA     ICBLL,X
145F  A9  04            LDA     #<PND-PST>^
1461  9D  49  03        STA     ICBLH,X
1464  A9  0B            LDA     #PUTCHR
1466  9D  42  03        STA     ICCOM,X
1469  20  56  E4        JSR     CIOV

146C  A9  0C            LDA     #CLOSE      ;Close cassette buffer
146E  9D  42  03        STA     ICCOM,X
1471  20  56  E4        JSR     CIOV
1474  60                RTS
1475  43  3A    CAS:    .ASCII  'C:'        ;Name of cassette device.
1477  9B                .BYTE   EOL

      1478      PND = .                     ;Used by MAKETP to figure
                                            ;out what to save. Must
                                            ;point to just after last
                                            ;byte of program.

      0000              .END
```

©

ATARI 800/400
Users and Dealers

We have software for your computer including:

MUSIC1 and Christmas MUSIC — for the Atari Music Composer Cartridge - 24K $24.95 Disk, $19.95 Tape.

MENUMAKR — to make and store GRAPHICS O text and graphics screens - 16K $19.95 Disk, 8K $14.95 Tape.

HANGMAN and MATHFAKS — for educational help for your children, 24K Each - Disk $24.95, Tape $19.95.

Word Search Puzzle Maker, Printer Required - 24K $24.95 Disk $19.95 Tape.

And More!!! (Ask about our Word Processor.)

- We also sell Epson, Atari, Dysan, Verbatim, and more. -

Computer's Voice

2370 ELLA DR. - DEPT. C-9
FLINT, MI. 48504 - (313) 238-5585

COMPUTE! The Journal For Progressive Computing™

www.commodore.ca

THE STAR MODEM

From Livermore Data Systems

RS232 MODEM	SALE $128
IEEE 488 MODEM	**SALE $199**
RS232 CCITT	$170
IEEE 488 CCITT	$280

STAR Modem is the price performance leader with a full 2 YEAR FACTORY WARRANTY.

MIPLOT Intelligent Plotter
by Watanabe Instruments (Digiplot)

SPECIAL $1195

Has all intelligent functions for producing graphs and drawings including 8 vector and 4 character commands. Solid and broken lines can be specified. Character generator for alpha, numeric, and symbols. Characters can be rotated in 4 orientations, and can be 16 sizes. Coordinate axes drawn by specifying graduation interval and number of repetitions. Parallel ASCII interface. 11 x 17 paper.

We carry Apple II+ from Bell & Howell

REVERSAL (Spracklen) 32K Apple	28.00
Super FORTH 48K Apple	40.00
Energy Miser for PET, Apple, or Zenith	24.50
Data Manager (Lutus) 24K Apple	40.00
Histo-Graph (Boyd) 48K Apple	24.50
Data-Graph (Boyd) 48K Apple	40.00
Apple II User's Guide (Osborne)	12.00
Introduction to Pascal (Sybex)	10.30
Pascal Handbook (Sybex)	12.00
Graphics Cookbook for Apple	9.90
Musical Applications of Micros (Chamberlin)	20.00
Basic FORTRAN (Coan)	7.25

SPECIALS

EPSON MX-80 Printer	
EPSON MX-80 F/T Printer	
EPSON MX-70 Printer	
EPSON MX-100 Printer	
Centronics 739 Printer	750
STARWRITER Daisy Wheel Printer	1445
Leedex Video 100 12" Monitor	129
ZENITH DATA SYSTEMS	
Z19 Video Terminal	729
Z89 with 48K	2150
Extra 16K RAM	115
Z-47 Dual 8" Drive	2775
SYM-1	209
SYM BAS-1 BASIC or RAE-1/2 Assembler	85
KTM-2/80 Synertek Video Board	349

PROGRAM YOUR OWN EPROMS

Branding Iron for PET/CBM — $79
EPROM Programmer with software for all ROM versions. Includes all necessary hardware and software to program or copy 2716 and 2532 EPROMS.

WORDCRAFT 80 Word Processor-CBM — $300
Extremely comprehensive word processing package for 8032. Allows you to view full 117 character line on screen by using "window" approach. Supports all major printers.

Dow Jones Portfolio Management System $120
Online access to quotes and data for 6000 stocks from Barrons, WSJ, and Dow Jones News Service. Maintains portfolio accounting including tax records.

Legal Time Accounting — $500
Allows automatic processing of matters by client, lawyer, and activity. Provides reports on client data, aging analysis, activity revenue, and productivity. Interfaces with Wordcraft 80.

EBS Business System for CBM — $600
Provides extremely comprehensive integrated inventory and accounts receivable package including invoices, packing slips, mail labels, statements, deposit slips, and 17 reports.

EASTERN HOUSE SOFTWARE

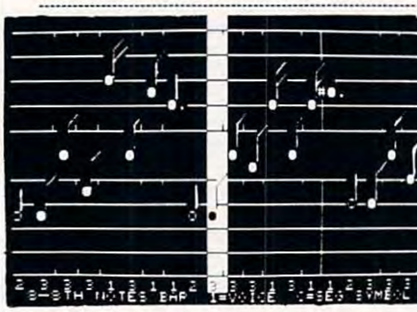
ROM Rabbit	42.50
Cassette Rabbit	25.50
Macro Assembler and Text Editor	42.50
MAE Macro Assembler and Editor	144.00

6502	7.45	10/6.95	50/6.55	100/6.15
6502A	8.40	10/7.95	50/7.35	100/6.90
6520 PIA	5.15	10/4.90	50/4.45	100/4.15
6522 VIA	6.45	10/6.10	50/5.75	100/5.45
6532	7.90	10/7.40	50/7.00	100/6.60
2114-L200 ns RAM		3.75	25/3.50	100/3.25
2114-L300 ns RAM		3.15	25/2.90	100/2.65
2716 EPROM		7.00	5/6.45	10/5.90
2532 EPROM				14.50
6116 Hitachi 2K x 8 CMOS STATIC RAM				14.50
4116 200ns				8 for $20
Zero Insertion Force 24 pin Socket				$2.00

CASSETTES — AGFA PE-611 PREMIUM
High output, low noise, 5 screw housing, labels.

C-10	10/5.65	50/25.00	100/48.00
C-30	10/7.30	50/34.00	100/66.00

All other lengths available. Write for price list.



4 PART HARMONY MUSIC SYSTEM for PET
The Visible Music Monitor, by Frank Levinson, allows you to easily enter, display, edit, and play 4 part harmony music. Includes whole notes thru 64ths (with dotted and triplets), tempo change, key signature, transpose, etc. The KL-4M unit includes D to A converter and amplifier (add your own speaker).
KL-4M Music Board with VMM Program — $59.90

commodore

CBM-PET SPECIALS

FREE — **Up to $380 free merchandise with purchase of one of following CBM-PET items:**

		FREE
8032 32K, 80 x 25 CRT, business keyboard	1495	320
Micro Mainframe – available soon		
8050 Dual Disk Drive – 1 megabyte capacity	1795	380
8250 Dual Disk Drive – 2 megabyte available soon		
2031 Single Disk Drive	695	135
4016 full size graphics keyboard	995	200
4032 full size graphics keyboard	1295	275
4040 Dual Disk Drive	1295	275
4022 Tractor Feed Printer	795	160
C2N External Cassette Deck	75	13
VIC 20 Personal Computer	300	25
Used CBM/PET Computers		CALL

***** EDUCATIONAL DISCOUNTS *****
Buy 2 PET/CBM Computers, receive 1 FREE

WordPro 3+ – 32K CBM, disk, printer	250
WordPro 4+ – 8032, disk, printer	335
OZZ Data Base System for CBM 8032	335
VISICALC for PET or ATARI	170
SM-KIT - Super PET ROM Utilitites	40
Programmers Toolkit – PET ROM Utilities	34.90
PET Spacemaker II ROM Switch	36.00
2 Meter PET to IEEE or IEEE to IEEE Cable	40.00
Dust Cover for PET	6.90
IEEE-Parallel Printer Interface for PET	110.00
IEEE-RS232 Printer Interface for PET	120.00
The PET Revealed*	17.00
Library of PET Subroutines	17.00

WRITE FOR SYSTEM PRICES

DISK SPECIALS

SCOTCH (3M) 5¼"	10/2.85	50/2.75	100/2.65
SCOTCH (3M) 8"	10/2.90	50/2.80	100/2.70
Verbatim 5¼"	10/2.45	50/2.40	100/2.35
(add 1.00 for 5¼" Verbatim plastic storage box)			
Verbatim 8" Dbl. Dens.	10/3.45	50/3.35	100/3.25
BASF 5¼"	10/2.40	20/2.35	100/2.30

WRITE for prices on WABASH Disks

WE STOCK MAXELL DISKS

Diskette Storage Pages		10 for 3.95
Disk Library Cases	8" - 2.85	5" - 2.15

ATARI 800 $745
All Atari Modules 20% OFF

ATARI EDUCATIONAL PLAN — **Write for details.**

A P Products 15% OFF
A P Hobby-Blox 15% OFF

ALL BOOK and SOFTWARE PRICES DISCOUNTED

The 8086 Book (Osborne)	14.00
Z8000 Assembly Language Programming	16.90
PET Personal Computer Guide (Osborne)	12.75
PET and the IEEE-488 Bus (Osborne)	13.60
6502 Assembly Language (Osborne)	14.45
Programming the 6502 (Zaks)	10.45
6502 Applications Book (Zaks)	10.45
6502 Software Cookbook (Scelbi)	9.45
CP/M Handbook (w/ MP/M) Zaks	11.85
Practical BASIC Programs (Osborne)	13.60
Some Common BASIC Programs (Osborne)	12.75

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.

www.commodore.ca

SuperGraphics

by John Fluharty **$30**

SuperGraphics provides machine language extensions 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 that 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.

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.

SuperGraphics commands include GRAPHIC, TEXT, RVS, SET, DRAW, FILL, PLOT, MOVE, PRINT, CSET, CMOVE, DISPLAY, PUT, SWAP, PAUSE, and SOUND.

Please specify machine type and ROM version, disk or tape.

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	$85
RAM/ROM - - 8K	120
Battery Backup Option	30

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.

Available soon:

Metacompiler for FORTH **$30**

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

SINGLE DISK DRIVE FOR PET/CBM

PEDISK II from cgrs Microtech is a new system ready to plug into large keyboard PET/CBM systems. The package offers speed,reliability, and IBM compatibility.

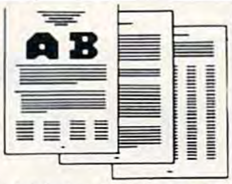
Complete System Prices with DOS and cable:

5" 40 track, 1 drive, 143K	$495
5" 80 track, 1 drive, 286K	690
8" IBM 3740 format, 77 track, 250K	995

SOFTPACK-1 from Competitive Software 25
16 games and utilities for PET

MICROREVERSI for PET by Michael Riley 10
Super Machine Language Version of Othello

PAPER-MATE 60 COMMAND WORD PROCESSOR



Paper-Mate is a full-featured word processor for $29.00 by Michael Riley. Paper-Mate incorporates 60 commands to give you full screen editing with graphics for all 16k or 32K PETs, all printers, and disk or tape drives. It also includes most features of the CBM WordPro III, plus many additional features.

For writing text, Paper-Mate has a definable keyboard so you can use 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, PET graphics as well as text can be used. Paper-Mate can send any ASC11 code over any secondary address to any printer.

Paper-Mate works on 16K or 32K PETs with any ROM, cassette or disk, and CBM or non-CBM printers. An 8K version is in the planning.

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

On Tape (with manual):	**$40.00**
On Disk (with manual):	**$42.00**
Manual Separate:	**$ 1.00**

KMMM Pascal for PET/CBM $75

A subset of standard Pascal with extensions • Machine Language Pascal Source Editor • Machine Language P-Code Compiler • P-Code Interpreter (for debugging and learning) • P-Code to machine language translator for optimized object code • Run-time package • Floating point capability • User manual and sample programs • Includes source code editor

Specify ROM version (16K minimum), disk or tape.

EARL for PET (disk file based) $65

Editor, Assembler, Relocater, 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.

SUPERSORT by James Strasma $35

Supersort 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.

FLEX-FILE

For CBM/PET Computers

Self Calculating DATA BASE
REPORT WRITER
MAILING LIST

Flex File is a set of flexible, friendly programs to allow you to set up and maintain a data base as well as print files with a versatile Report Writer or a Mail Label routine. Programmers will find it easy to add subroutines to their own programs to make use of Data Base files.

RANDOM ACCESS DATA BASE

Record size limit is 250 characters. The number of records per disk is limited only by the size of each record and the amount of 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 from a specified field. The Find command locates any record when you enter all (or a portion of) the desired key field. Field lengths can vary from record to record provided the sum of the fields does not exceed the size of the record. This allows maximum packing of information. The file can be sorted by any field. Any field can be specified as a key field at any time. Sequential files from other programs can be converted to random files, and random can be converted to sequential. Maximum record size, fields per record, and order of fields can be changed at any time.

MAILING LABELS

When record size is 127 characters (typical for mailing list), each disk can handle over 1000 records (about 2800 with the 8050 drive). Labels can be printed any number of labels across, and in any column position. Any number of fields can be printed on a label in any order, and two or three fields can be joined together on one line (like first name, last name, and title). A "type of customer" field allows selective printing.

REPORT WRITER

The contents of any field can be placed in any column. Numerics can be decimal point justified and rounded to any accuracy. Any column can be defined as a series of mathematical functions performed on other columns. These functions may include +, −, x, /, %, and various log and trig functions. Results of operations such as running total may be passed from row to row. At the end of the report a total and/or average can be calculated for any column. Complete record selection, including field within range, pattern match, and logical functions can be specified individually or in combination with other parameters.

Flex File was developed by Michael Riley.

Flex File System **$60**

Specify machine size (32K recommended) and ROM type for both disk and computer.

PERSONAL SOFTWARE

Microchess	17.00
Checker King	17.00
Gammon Gambler	17.00
Time Trek	13.45
Bridge Partner	13.45

HAYDEN SOFTWARE

Complex Mathematics	12.70
Engineering Mathematics	12.70
General Mathematics	12.70
MCAP Circuit Analysis Program	21.00
Energy Miser	24.50

CBM SOFTWARE

Legal Time Accounting Package	
Medical Accounting Package	
Complete General Accounting Package	
Comprehensive Investment Analysis Package	
Dow Jones Portfolio Management	$135
Personal Tax Calculator	65
Tax Preparation System	445
Information Retrieval and Management Aid	400
Wordcraft 80 Wordprocessor Package	325
Pascal Development Package	295
Assembler Development Package	99

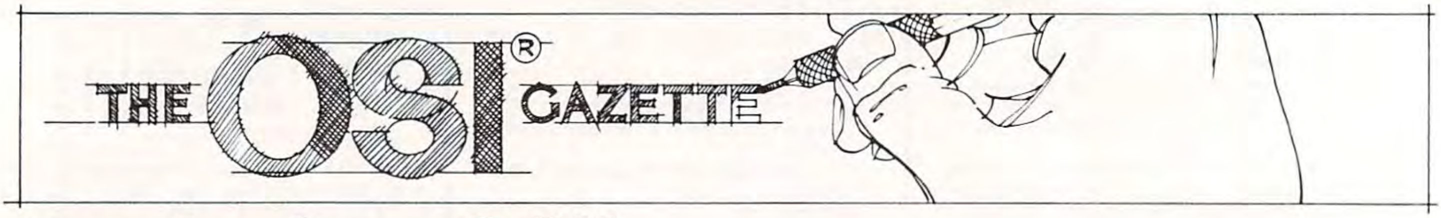
252 Bethlehem Pike
Colmar, PA 18915

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.

www.commodore.ca



Exploring OSI's Video Routine

Kerry Lourash
Decatur, IL

Welcome to the BASIC-in-ROM Explorers' Club! On our journey through the Fill-the-Buffer routine, we had to bypass a tour of the Video routine at $BF2D. Now we are ready to unravel the mysteries of the routine that makes objects appear and disappear on the screen.

The Video routine (VR) is a section of machine language code located in BASIC-in-ROM at $BF2D-BFFC. Input from the keyboard and the LOAD routine and output from the SAVE, PRINT, LIST, etc. routines are fed to the VR, which displays the information on the screen.

This is what the VR does:

1. Prints text on the screen.

2. Does automatic carriage return (CR) and line feed (LF) when the end of the video line is reached.

3. Scrolls the screen.

4. Slows printing rate, if necessary, for compatibility with printers or other slow ipherals.

Preparing For Our Journey

The format of our map (see Fig. 1) is the same as that of our first trip (**COMPUTE!** #12, p. 90). I've shown subroutines immediately after the point where they are called, instead of in numerical order. Addresses at the left are part of the main routine and indented addresses are subroutines.

The result approximates an outline of the VR. Machine language addresses have been retained so ML readers can pinpoint and disassemble any part of the routine for more information. BASIC-oriented readers should consider the addresses as line numbers. Most assembly language mnemonics have been replaced by explanations of what is happening. The few mnemonics that are used have their BASIC equivalents listed in the heading of the chart.

All numbers are hexadecimal unless specified otherwise.

The VR uses several locations in RAM and ROM:

0200 - Holds address of the video memory location where current character will be printed.

0201 - Temporary storage for character to be printed.

0202 - Storage for A register while A, X, and Y are pushed on the stack. Also holds the number of bytes to be scrolled in the last page of video memory.

0206 - TV delay loop value.

0207 - 020E Scroll-one-byte subroutine.

BFFB - Holds number of last page of video memory for C1P(D3).

BFFC - Holds number of last page of video memory for C2P(D7).

FFE0 - Cursor "home" position; C1P=65, C2P=3F.

FFE1 - Characters/line-1; C1P=17, C2P=3F.

FFE2 - Video memory size; 0=1K, 1=2K.

D000 - D3FF C1P video memory.

D000 - D7FF C2P video memory.

Both the Fill-the-Buffer routine and the video routine generate an automatic CR/LF, but the two functions shouldn't be confused. Unlike the FTB, whose "terminal width" counter is in RAM,(loc. 0F), the VR has its character/line permanently set in the monitor (loc. FFE1). If you set the terminal width at less than the char./line value, the FTB will tell the VR to do a CR/LF before the VR does one automatically. However, if you set the terminal width greater than the video line length, the VR will still be triggered at 24 or 64 (decimal) characters, and the video line length will not be longer. You may see a CR/LF at seemingly random intervals. The intervals are not random; both FTB and VR are doing CR/LFs independently of each other. Another difference is that the VR doesn't generate nulls after its CR/LF, as the FTB can. A third difference is that the actual CR/LF subroutines are located in the VR. When you hit the RETURN key or the FTB does a CR/LF, the FTB is sending a CR and a LF character to the VR.

I'd also like to clear up the definition of a few terms, such as "high" and "low" bytes and "pages." The address D365 is a two-byte address. D3 is the high byte and 65 is the low byte. A page contains 256 (dec.) or 0100 (hex) bytes. Notice that the high byte is also the page number (0000-00FF is zero

www.commodore.ca