NO. 67 SPREADSHEETS
U.S. Edition: $2.50
International Edition: $3.00

Complete Programs Included

MICRO™

DEC 83

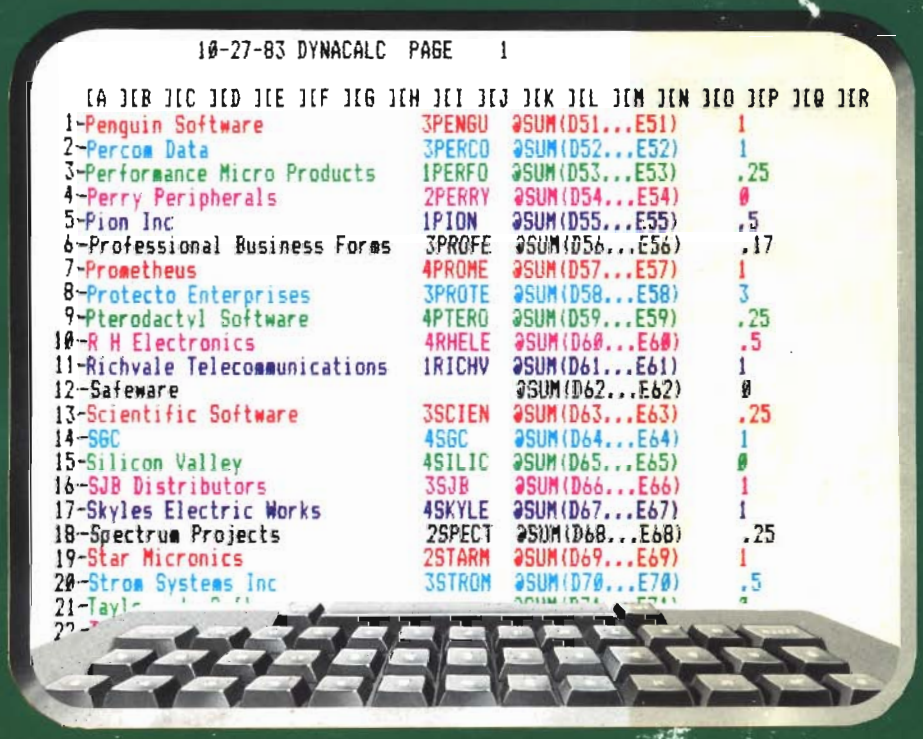
Advancing Computer Knowledge

Spread Sheets

- An in depth "how to" on using Spreadsheets

MicroCalc

- Complete Mini Spreadsheet program for Apple, Atari, Commodore and TRS-80C



10-27-83 DYNACALC PAGE 1

	[A	][B	][C	][D	][E	][F	][G	][H	][I	][J	][K	][L	][M	][N	][O	][P	][Q	][R
1	Penguin Software										3PENGU	@SUM(D51...E51)			1			
2	Percom Data										3PERCO	@SUM(D52...E52)			1			
3	Performance Micro Products										1PERFO	@SUM(D53...E53)			.25			
4	Perry Peripherals										2PERRY	@SUM(D54...E54)			0			
5	Pion Inc										1PION	@SUM(D55...E55)			.5			
6	Professional Business Forms										3PROFE	@SUM(D56...E56)			.17			
7	Prometheus										4PROME	@SUM(D57...E57)			1			
8	Protecto Enterprises										3PROTE	@SUM(D58...E58)			3			
9	Pterodactyl Software										4PTERO	@SUM(D59...E59)			.25			
10	R H Electronics										4RHELE	@SUM(D60...E60)			.5			
11	Richvale Telecommunications										1RICHV	@SUM(D61...E61)			1			
12	Safeware											@SUM(D62...E62)			0			
13	Scientific Software										3SCIEN	@SUM(D63...E63)			.25			
14	SGC										4SGC	@SUM(D64...E64)			1			
15	Silicon Valley										4SILIC	@SUM(D65...E65)			0			
16	SJB Distributors										3SJB	@SUM(D66...E66)			1			
17	Skyles Electric Works										4SKYLE	@SUM(D67...E67)			1			
18	Spectrum Projects										2SPECT	@SUM(D68...E68)			.25			
19	Star Micronics										2STARM	@SUM(D69...E69)			1			
20	Strom Systems Inc										3STROM	@SUM(D70...E70)			.5			
21	Tayl																	
22																		

Plus ... Reviews, Columns, and Much Much More

See page 29

See page 83

See page 9

C64 Alarm Clock
Master Directory for Apple
Color Computer Memory Map
Break Up for Atari

12
0 74470 11690

Calc Result

The Commodore 64™ Spreadsheet that puts you a million miles ahead

CALC RESULT. The one spreadsheet guaranteed to turn your Commodore into a powerful financial tool.
Offering you every feature found on other, more expensive programs for much less the cost.
Flexible...you can view four different areas at once.
Versatile...customize your own print formats.
Distinctive...display beautiful color graphics.
CALC RESULT *Advanced* is a three-dimensional spreadsheet with built-in HELP function and 32 pages of memory. For the Commodore 64. $149.95. For the CBM™ 8032 $199.00.
For first time users CALC RESULT *Easy* gives you a fast way to perform financial calculations—easily. For the Commodore 64. $79.95.
For a down to earth demonstration of either version visit your local dealer today.

Developed by:
handic software ab
-a company in the Datatronic group-

A Product of:

Distributed by:

DES—DATA EQUIPMENT SUPPLY
213-923-9361

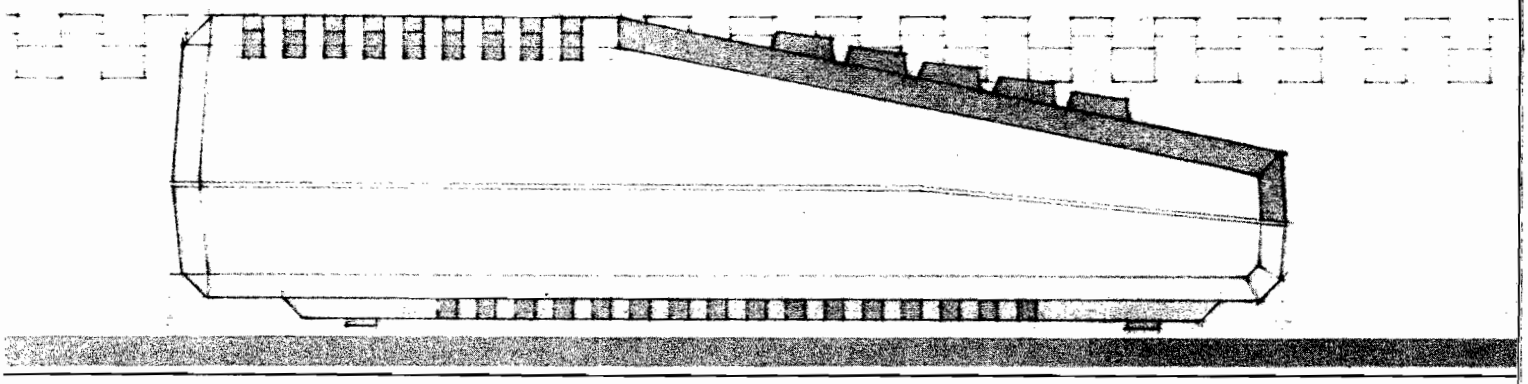
SOFTEAM
800-421-0814

COMPUTER MARKETING SERVICES, INC.
800-222-0585

WAREHOUSE 1, INC.
EASTERN U.S./800-253-5330
WESTERN U.S.-800-255-0056

BLUE SKY SOFTWARE
Ashland Office Center
Evesham & Alpha Avenues
Voorhees, NJ 08043
609/795-4025

Commodore 64 is a trademark of Commodore Business Machines

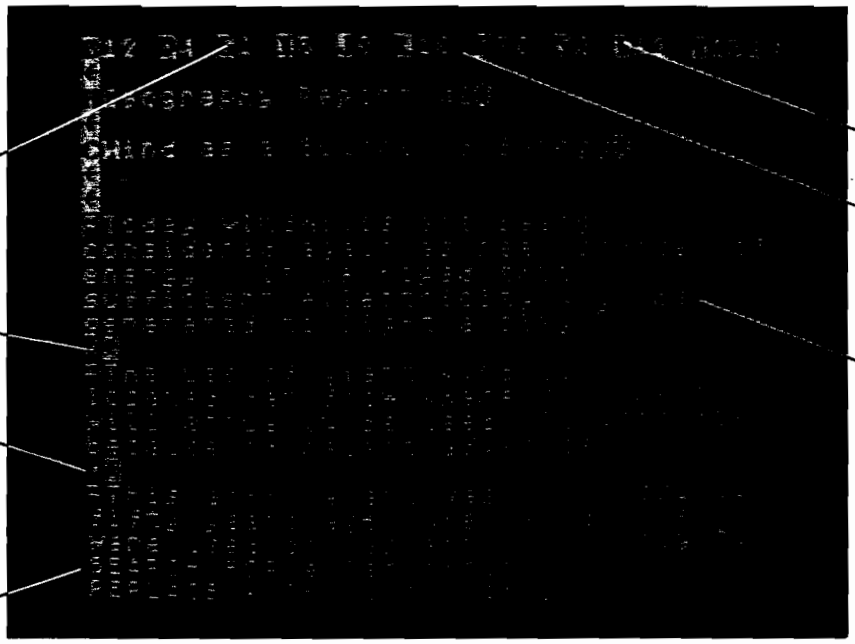


Prints with built-in format or lets you create your own: center, underline. Boldface, elongated, proportional and condensed print.

Inserts and deletes characters, lines or blocks of text.

Duplicates and moves blocks of text.

Searches for and/or replaces words or phrases, one at a time or all at once.



Changes line spacing, margins, print type, paragraph indents anywhere in the document.

Instantly reformats.

Enters text easily, with word wrap, one main menu and helpful prompts.

Creates multi-line headers and footers.

Prints double columns, form letters, multiple copies of a page or document, at the touch of a key.

Numbers pages and creates section numbers automatically.

Corrects spelling (with extra software).

Chains and merges files.

Has "print preview": you can check your copy before it's printed.

You can't find a friendlier, more powerful word processor at twice the price. New AtariWriter.™ Under $100.

Now you can do multi-featured word processing at home, simply. At a family budget price.

Our ROM-based cartridge technology means you can use new AtariWriter on any ATARI® Home Computer (even 16K) for personal and business correspondence, term papers, committee reports, mailings, etc.

It also lets you choose between cassette and disk storage systems.

One very special AtariWriter feature: you can correct as you write, without switching back and forth between Create and Edit modes.

And our memory buffer offers an "undo" command to let you change your mind, and restore text you've just deleted.

Check into our remarkable AtariWriter, and our choice of letter quality and dot matrix printers, also reasonably priced, at Atari dealers. Call 800-538-8543 for dealer nearest you. In California, call 800-672-1404.

You'll do more with Atari Home Computers.

© 1983 Atari, Inc. All rights reserved. A Warner Communications Company

THE MOMENTUM™ GROUP

Presented by ≡Gold Disk™■

Featuring the finest in **educational and home utility** software for the Apple:

ANIMATION GRAPHICS: PAINT PROGRAM
Powerful yet easy-to-use shapes creator and animator for the novice or experienced graphics artist. Apple II or IIe, 48 K. 1 drive. $35.00

ANIMATION GRAPHICS: SUPER SLIDE SHOW
Powerful slide show creator for novice or expert. Apple II or IIe, 48 K. 1 drive. $29.00

INTRODUCTORY SPECIAL: NOW THROUGH JANUARY 31, 1984: Buy both the PAINT PROGRAM and the SUPER SLIDE SHOW together for one low package price of $45.00.

SCANDURA MICRO-TUTOR II SERIES (Grades 1-5)
Interactive and individualized instruction disks with hi-res graphics and the latest advances in cognitive theory. By Dr. Joseph Scandura, specialist in structural learning.

— **Complete Tutorial Arithmetic Series** (4 disks complete). A total system to pinpoint your child's needs in arithmetic and get results. Apple II or IIe, 48 K. I drive. $135.00. With Diagnostic Pre-Test Option, $160.00.
— **Complete Language Arts Series** (8 disks complete). Helps identify the language skills needed by your child and improves them dramatically. Apple II or IIe, 48 K. 1 drive. $400.00

LOCUS MICRO-TUTOR (All Grades)
Allows an educator to build a customized course for students on any subject matter with C.A.I. instruction disks. Can also print and store test materials. No programming knowledge necessary. Apple II or IIe, 48 K. 1 drive. $59.95.

LITMAS DATA SYSTEM
High-speed system for data storage, retrieval and statistical analysis. Written by Eric Sohr, M.D., LITMAS is the only cross-indexing data base for the Apple, with up to 512 keywords possible, and up to 2048 combinations of keyword phrases possible. LITMAS' speed is derived from its unique "bit map" structure. Delightfully-easy to use, excellent manual. Apple II or IIe, 48 K. 1 drive. $145.00

BRAHMAN HOME HEALTH DISK
Comprehensive medical information program. Written by an E.R. physician, this program can help you live healthier and possibly save someone's life. Factual, inclusive, consumer-oriented . . . Covers emergencies, injuries, drugs, treatments, etc. 2 full disks with 15 programs. Apple II or IIe, 48 K. 1 drive. $79.95. Annual update available.

PRO—FORMA TAX PREPARER/TAX BREAK ANNUAL
On time and reliable since 1981 — never missed a year's update. A sophisticated tax optimizer which quickly determines the lowest tax by automatically taking into account every benefit, loophole, exclusion, adjustment and credit possible. User-oriented and very fast, typically taking 3 minutes from data input to return preparation. Program generates a detailed 7-page tax summary report indicating what form, line number and amount to be entered to obtain the very lowest tax. Also contains a unique "AUDIT POTENTIAL SCORE" which compares your return with that of a typical taxpayer with the same income, alerting you to any unusually high deductions that could trigger an audit. Apple II or IIe, 48 K. 1 drive. $129.95. Annual update available.

Distributed by MOMENTUM through your local dealer, or call national order agent for VISA/MC or C.O.D. order at **1-800-368-2260** in Maryland, 301-592-5949. Or write to MOMENTUM, 300 W. 31st St., Baltimore, Md. 21211.

30-DAY MONEY BACK GUARANTEE IF NOT SATISFIED!

APPLE IS A TRADEMARK OF APPLE COMPUTER INC.

penguin software™

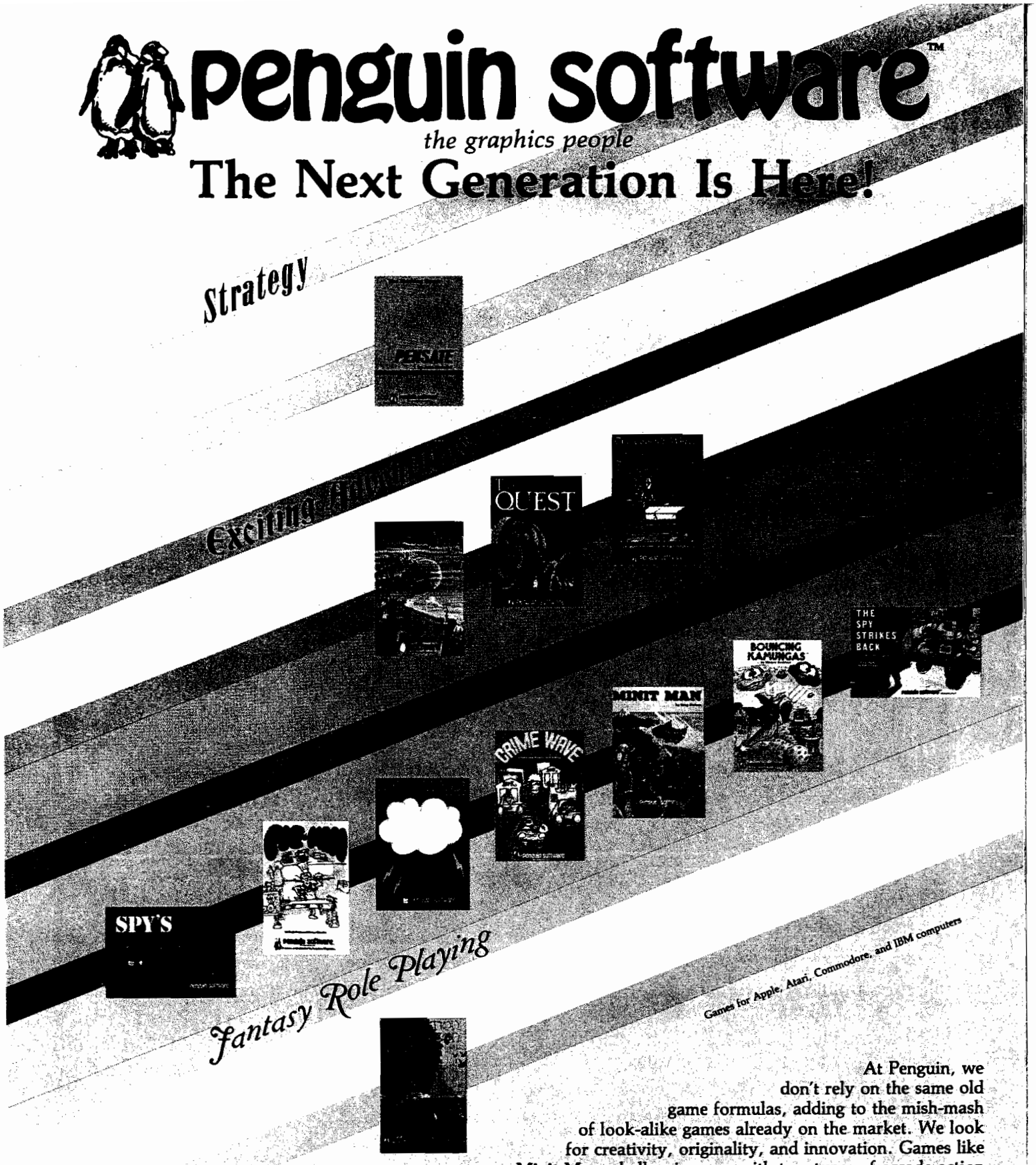
the graphics people

The Next Generation Is Here!

Strategy

Exciting Adventure

Fantasy Role Playing



Games for Apple, Atari, Commodore, and IBM computers

At Penguin, we don't rely on the same old game formulas, adding to the mish-mash of look-alike games already on the market. We look for creativity, originality, and innovation. Games like **Minit Man**, challenging you with two types of arcade action simultaneously; **Coveted Mirror**, an adventure with animation and arcade games throughout; **The Spy Strikes Back**, an arcade game with strategy and a touch of adventure; **Pensate** and **Tactic 9**, games that make you think; and **Expedition Amazon**, an intriguing and humorous fantasy game.

And Each Is Only $19.95! More Fun For Your Money From Penguin Software

Write for a free catalog

830 Fourth Avenue, Box 311 Dept. C Geneva, IL 60134 For information - Call (312) 232-1984 Dealer orders only - Call (800) 323-0116

Apple is a trademark of Apple Computer, Inc. Atari is a trademark of Atari, Inc. Commodore is a trademark of Commodore Business Machines, Inc. IBM is a trademark of International Business Machines Corp. Penguins are hard on pool toys.

Outsmart your computer.

Show your computer who's boss. Earn its respect. With a Datamost book.

No matter what age or ability level you're at, we have the right book that talks just to you. And your computer. Whether you own an Apple.* An Atari.* Or just about any brand.

All our books are incredibly easy to understand.

Which will make it incredibly easy to understand your computer.

What If You're Scared Of Books About Computers?

Don't be.

Our books are written in friendly, familiar American English. Highlighted with cartoons. And illustrations. So they're fun to read. As well as educational.

And there's over 30 books to choose from. Basic computer learning books to programming books to coloring books.

You won't be bombarded with complicated programming routines until you're ready for complicated programming routines. And no funny technical talk until you've reached the level where you don't think it's funny.

We'll tell you what you need to know. And then, when that's understood, you can take the next step toward outsmarting your computer.

Before You Spend Big Bucks On A Computer, Spend Little Bucks.

Once you've decided, kind of, almost, nearly, what type of computer you think you'd like to buy, buy a Datamost book.

It'll help you understand the Atari or the Commodore* or the Apple of your eye.

Before you spend a lot of money. Before you take your computer home.

You'll make a better purchase decision. Because you'll understand what you're doing. And what you and your computer can do together.

So there won't be any misunderstandings to ruin your new relationship.

How To Get The Most Out Of Your Computer.

Get the most out of our minds.

DATAMOST INC™

The most out of our minds.™

Datamost books for every computer.

Everybody. Every level.

Pretty smart, huh?

Datamost, Inc., 8943 Fullbright Ave., Chatsworth, CA 91311, (213) 709-1202
*Atari is a trademark of Atari Computer.*Apple is a trademark of Apple Computer.
*Commodore 64 is a trademark of Commodore Business Machines, Inc.

MICRO™

Advancing Computer Knowledge

Spreadsheets

DEPARTMENTS

6 Editorial
102 Next Month in MICRO
104 Advertiser Index

Commodore

23 MICROCalc
Loren Wright
A mini-spreadsheet for Commodore computers

29 C1A Alarm Clock
Ian Adam
A real-time clock for C-64

33 Commodore Compass
Loren Wright
Word processing and personal accounting

35 Commodore Reviews

Color Computer

61 MICROCalc
Phil Daley
A mini-spreadsheet for TRS-80C

63 The Memory Map
A complete list of CoCo's addresses

73 CoCo Bits
John Steiner
Connect your own disk drive

76 CoCo Reviews

Apple

39 MICROCalc
Phil Daley
A mini-spreadsheet for Apple computers

41 CAT Sort
Mark Harris
Alphabetize your disk directories

47 Master Directory
Charles Hill
Organize your disk collection

56 Apple Slices
Phil Daley
A disk dump program for DOS and PASCAL

57 Apple Reviews

Atari

79 MICROCalc
Tom Marshall
A mini-spreadsheet for Atari

83 Break Up
Tom Marshall
Learn about simple animation

89 From Here to Atari
Paul Swanson
A Christmas greeting

92 Atari Reviews

97 Interface Clinic
Ralph Tenny

MICRO
P.O. Box 6502
Amherst, NH 03031
(603) 882-2...

Technical ...
...
Logan Walker

Contributing Editors
Cornelis Bongers
Dave Mahoney
John Steiner
Jim Strasma
Paul Swanson
Richard C. Vile, Jr.

Art Director/Typesetting
Nancy Lapolice

Typesetting
Lynda Feeley

Advertising
Sheila McDonough

Dealer Sales
Maureen Dubois

Circulation
Cindy Smith

Accounting
Donna M. Tripp

Publisher/Editor-in-Chief
Robert M. Tripp

Associate Publisher
Circulation
Nancy ...

MICRO (ISSN ...) is published monthly at Amherst, NH ... paid at ... postal ... Number ... subscriptions ... Form 3579 ... Other ... Northeast ... NH ... Telephone ... 347-0... Australia ... America ... Canada ... South ... Zealand ...

Copyright © 19... All Rights Reserved.

MICRO™ Editorial

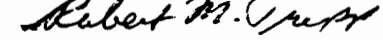
Is There MICRO After IBM?

I spent five years at a company that developed one of the first "microcomputers". It had only 8K bytes of memory, but could support 20 megabytes of disk, up to 8 keyboard/display stations, printers, modems, and much more. I helped develop the software — from absolutely nothing to a complete disk operating system with multi-user, real-time, editors, assemblers, a "high-level" language, application packages, and much more. At this time, the marketing division of the company could not figure out how to market this new product. Eventually they solved the problem by waiting for two or three years for *IBM* to catch up technologically and to produce the 3270 terminal. Then, our company emulated it! When I quit in frustration, the President spent well over an hour discussing what I felt was wrong with the company and what I would do to improve it. I suggested that he fire anyone that could spell *IBM*.

That was in 1974. Now, it almost looks as if history is repeating itself. While others have developed superior systems and lead the way, everyone is embracing *IBM*. How many of the following developments are directly related to the *IBM* announcements of the PC and the PCjr: DEC lost about 30% of its market value in a couple of days; TI announced it was discontinuing its TI99/4 completely; Atari continues to report losses in excess of 100 million dollars per quarter; Apple reduced the price on its LISA and Apple IIe; and all of the trade and financial journals speculate on *IBM*'s dominance in the personal, home, and business markets. Almost every knowledgeable person will admit that *IBM* is not the best or cheapest — but, it is *IBM*.

How does this effect you? There will be many pressures on you to consider an *IBM* as your next microcomputer, or, perhaps to immediately replace your current system. If *IBM* has the impact predicted by some, then that might be hard to resist.

How does this effect **MICRO**? **MICRO** was founded in 1977 to support the 6502 microprocessor which we felt was very good and which was not getting the attention it deserved. In 1981 we expanded coverage to the 6809 for similar reasons. I expect that many popular magazines will reduce and/or eliminate their general 6502/6809 based system coverage as *IBM*'s significance grows. **MICRO**, however, will remain true to its charter — and continue to bring you the best of the 6502 and 6809 worlds.



Robert M. Tripp
President/Editor-in-Chief

Statement of ownership, management, etc., required by the act of Congress of October 23, 1962, of MICRO, published monthly at Chelmsford, Massachusetts, for November 1983.

The name and address of the publisher is MICRO INK, 34 Chelmsford Street, Chelmsford, Massachusetts. The President/Editor-in-Chief is Robert M. Tripp of Chelmsford, Massachusetts.

The owner is THE COMPUTERIST, Chelmsford, Massachusetts and the names and addresses of stockholders owning or holding one percent or more of the total amount of stock are: Robert M. Tripp and Donna M. Tripp of Chelmsford, Massachusetts.

The known bondholders, mortgagees and other security holders owning one percent or more of the total amount of bonds, mortgages or other securities are: none.

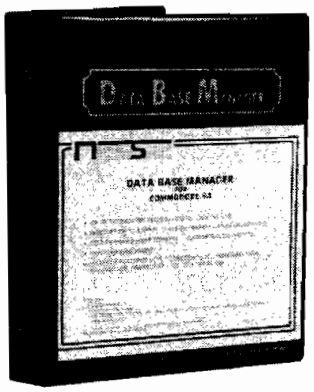
The average number of copies of each issue of this publication sold or distributed through the mails or otherwise to paid subscribers during the twelve months preceding the date shown above is: 21,857.

I certify that the statements made by me above are correct and complete.

Signed: Robert M. Tripp
President/Editor-in-Chief

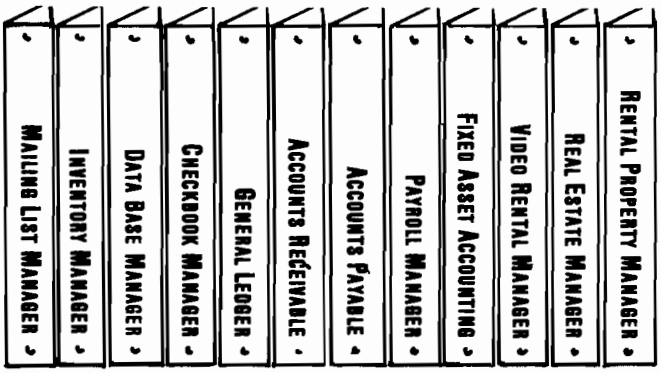
MICROSPEC

SOFTWARE MEANS BUSINESS FOR THE COMMODORE 64



When it's time to get serious, it's time to boot up MicroSpec business software. Our complete line of business software is made to give you some real applications for your Commodore 64. From data base management to full accounting software, we have the package for you.

It's attention to detail that makes our packages so beautiful and makes them stand out from the rest. We realize that most people are first time users, so we designed all our packages to be completely menu driven and user prompted for each input. We also know that most people use only one disk drive, so we designed all our packages to virtually eliminate disk swapping. Other features like non destructive input routines really make our software easy to use. But all this doesn't restrict you. Pure random access file structure maximizes your disk capacity and allows you to bring up any record for viewing in less than a second.



In our efforts to put together the best packages available, we worked on more than the software. We took the same approach with the documentation as the software. We made it complete and easily understood for the first time user. We even provide sample reports in many cases.

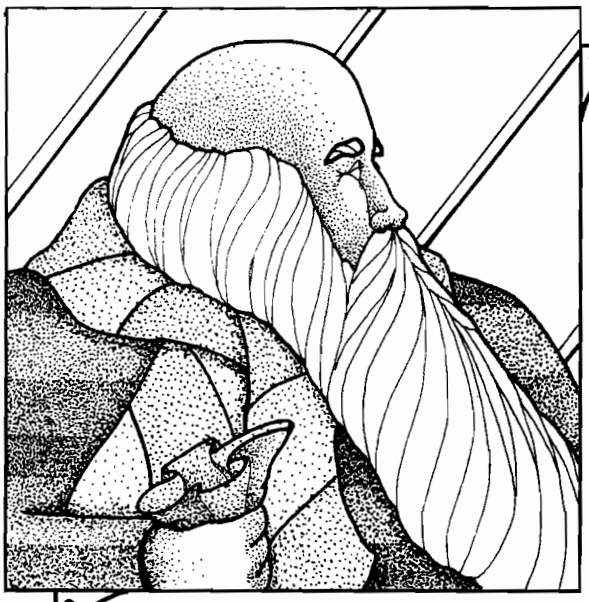
The Demonstration Package, which shows how each program runs, is available for $19.95. So, if you're serious about your 64, call or write for a complete brochure or **go right down to your nearest computer retailer for a demonstration.**

WHEN YOU AND YOUR 64 ARE READY TO GET DOWN TO BUSINESS GIVE US A CALL

MICROSPEC

P.O. BOX 863085 • PLANO, TX 75086

(214) 867-1333



VICTORY SOFTWARE

WOULD LIKE TO WISH OUR CUSTOMERS

H • A • P • P • Y H O L I D A Y S

AND THANK THEM FOR THEIR PATRONAGE THROUGHOUT THE YEAR.

FEATURING PROGRAMS FOR THE **VIC-20** AND THE **COMMODORE 64.**

BOUNTY HUNTER $24.95*

An adventure in the Old West. Journey back with us into the days of Jessie James and Billy the Kid where the only form of justice was a loaded revolver and a hangman's noose. In this full-length text adventure, you play the role of Bounty Hunter, battling against ruthless outlaws, hostile Indians, wild animals and the elements of the wilderness with only your wits and your six gun. Average solving time: 20-30 hours. If you love adventures, this one is a real treat.
Available for **COMMODORE 64** and the **VIC-20** (with 8K or 16K expander). Available on **TAPE** or **DISK**. Played with **JOYSTICK**.

KONGO KONG $24.95*

Climb ladders, avoid the barrels the crazy ape is rolling at you, and rescue the damsel. Commodore 64 version features 4 different screens!
Available for **COMMODORE 64** and **VIC-20**. Available on **TAPE** or **DISK**. Played with **JOYSTICK**.

GRAVE ROBBERS $19.95*

Introducing the first GRAPHIC ADVENTURE ever available for the VIC-20 or COMMODORE 64! With realistic audio-visual effects, you explore an old deserted graveyard and actually see the perils that lie beyond.
Available for **COMMODORE 64** and **VIC-20**. Available on **TAPE** or **DISK**. Played with **KEYBOARD**.

CHOMPER MAN $24.95*

Don't let the bullies catch you as you gobble the goodies! This program has 8 screens and still fits in the standard memory.
Available for **COMMODORE 64** and **VIC-20**. Available on **TAPE** or **DISK**. Played with **JOYSTICK** or **KEYBOARD**.

THE • EARTH • WARRIOR • SERIES

METAMORPHOSIS $24.95*

You stumbled into the nest of the Cyglorx and find yourself fighting off robot tanks guarding the Cyglorx eggs. You think you have everything under control and then the eggs start hatching.
Available for **COMMODORE 64** and **VIC-20**. Available on **TAPE** or **DISK**. Played with **JOYSTICK**

CREATOR'S REVENGE $24.95*

The creator assembled a massive army of robots and insects to take revenge on the earth. Destroy insects, get treasures, and get the neutron bomb deactivator. Battle robots and destroy the neutron bomb before it annihilates your city. Miss and you must face the mutants. Features 4 different screens.
Available for **COMMODORE 64.** Available on **TAPE** or **DISK**. Played with **JOYSTICK**.

LABYRINTH OF THE CREATOR $24.95*

Journey into the most complex and dangerous fortress ever built by the creator. You will encounter deadly robots, skulls, lakes, avalanches, false creators, and a creature who roams 256 rooms relentlessly pursuing you.
Available for **COMMODORE 64**. Available on **TAPE** or **DISK**. Played with **JOYSTICK**.

ILLUSTRATIONS: ELIZABETH HAUCK

*$3.00 additional for disk.

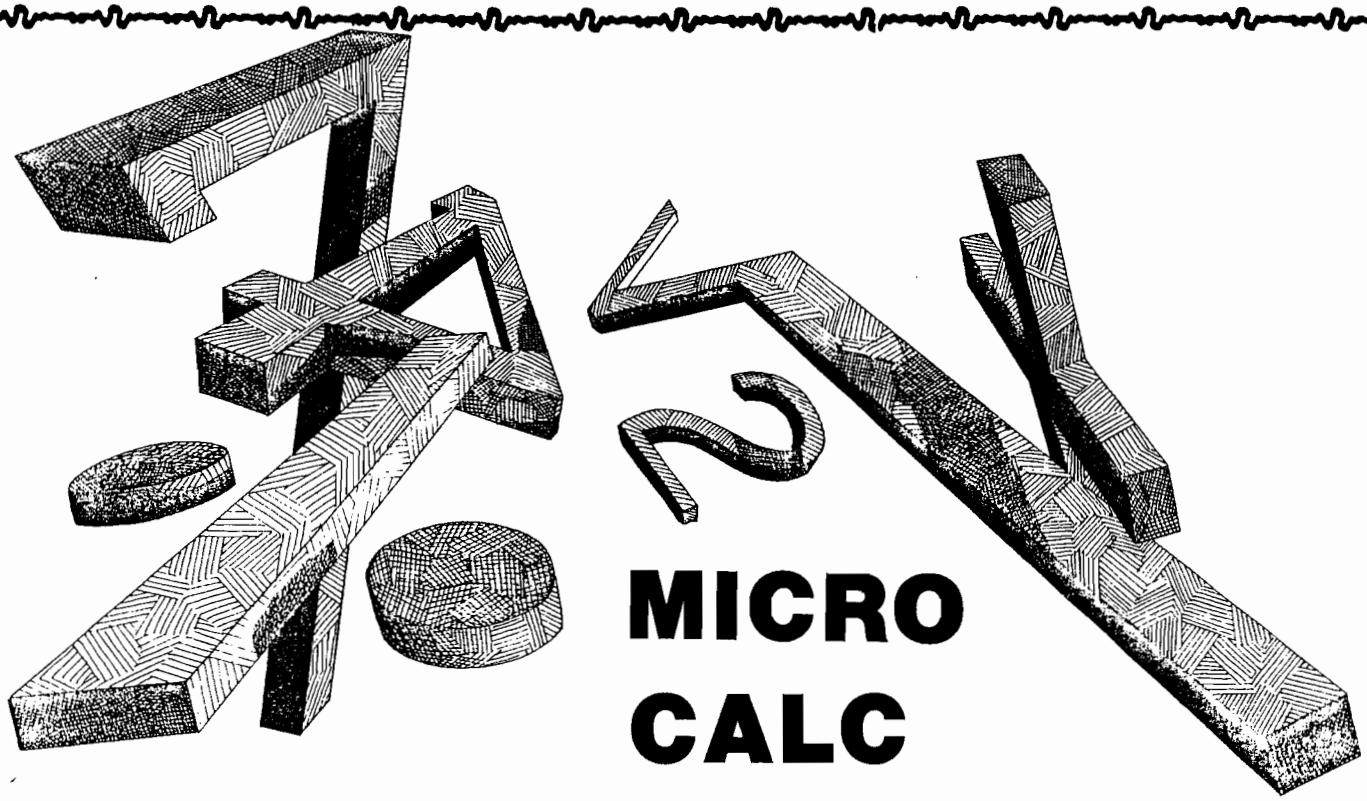
Check your LOCAL DEALER or order directly.
ORDERING: We accept personal checks, money orders, VISA, and MasterCard. Charge orders please include number and expiration date.
OVERSEAS ORDER: Please use charge, or have check payable through a U.S. bank.
CANADIAN CUSTOMERS: If you wish to write a check drawn through a Canadian bank, please multiply the total order by 1.25 for proper conversion.
Add $1.50 postage and handling per order. PA residents please add 6% sales tax.

VICTORY SOFTWARE INC.
7 Valley Brook Road
Paoli, Pennsylvania 19301
(215) 296-3787

VICTORY SOFTWARE

VISA

master charge



MICRO CALC

by Loren Wright

Micro Calc — What is it?

Micro Calc is a calculation program, not entirely unlike the spreadsheet programs described elsewhere in this issue. It is much simpler than a program such as VisiCalc, and that simplicity results in both advantages and disadvantages. This is not a spreadsheet program, so it is limited to much simpler calculations. However, as you will see from some of the examples presented later, there are many applications for such a quick calculational aide. All you need to know is the rules for BASIC arithmetic expressions.

In this issue we offer a ten-line version for an unexpanded VIC-20 with cassette. We also offer a 15-line version for the TRS-80 Color Computer, 20-line versions for the Commodore 64, PET, and Atari 400/800/1200, and a 23-line version for the Apple.

Haven't I seen this before?

The ten-line version was first presented in the March, 1982, issue of MICRO. A number of typographical errors in that listing have been corrected, and there have been several improvements. The VIC-20 version now includes the following additional features:

- ✓ multiple statements on a line
- ✓ convenient implementation of programmable function keys
- ✓ optional zeroing of user variables

The Apple, Commodore 64, and PET versions have added:

- ✓ multiple statements on a line
- ✓ function key implementation (C-64)
- ✓ optional zeroing of user variables
- ✓ disk support, with file name display
- ✓ error trapping (Apple)
- ✓ a total of 20 lines for calculations
- ✓ separate comment lines, one opposite each calculation line

The Atari version, presented here for the first time, allows *limited* use of IF...THEN, FOR...NEXT, and other BASIC constructions. The Color Computer version, also new, provides 15 lines for calculation, multiple statement capability, and file name display.

How to Use Micro Calc

See the article in each section of the magazine for listings and specific instructions. Below are general instructions.

RUN the program. The screen will fill with a sample screen. This is designed to calculate the monthly payment on an installment loan. On the Commodore 64 and the VIC-20, press the F7 key; on the others press the "@" key. The cursor will disappear for a few seconds, and then a number will appear next to the P? on the last line. This is the monthly payment calculated on an $8000 loan for 48 months at 11.9%. You may now move the cursor to the end of any line and delete and retype to try a different calculation. See what happens if the loan goes for only 36 months, or at only 9.9%, or if you decide to borrow $10000.

There are two kinds of statements allowed — *assignment* and *value request*. An assignment takes the following form:

[variable] = [BASIC numeric expression]

where [variable] is any single-letter floating-point variable name.

A value request takes the following form:

[variable]?

Typical assignments include:

```
X=5
A=X+3
J=SIN(X+3*A)
P=Y=5
```

Assignments may be combined on a single line by using semicolons or colons (see instructions for your implementation):

```
A=5:P=3.14159265:Z=TI
```

```
A=
M=
I=
I=I/1200
D=(1-(1+I)^-M)/I
P=A/D
P=INT(P*100+.5)/100
P?
```

PAYMENT: Calculates monthly payment, given starting balance A, number of months M, and annual interest rate I.

```
K=
F=3280.8336*K
F?
M=INT(F/5280)
G=F-M*5280
F=INT(G)
I=INT((G-F)*12+.5)
M?
F?
I?
```

METRIC CONVERSION: Converts kilometers to miles, feet, and nearest inch.

```
T=
U=
C=
P=3.14159265
V=180-T-U
V=P*V/180
U=P*U/180
B=SIN(U)*C/SIN(V)
B?
```

SOLVE TRIANGLE: Calculate a second side of triangle, given two angles (in degrees) and included side.

```
A=
B=
V=
V=3.14159265*V/180
D=A^2+B^2
E=2*A*B*COS(V)
C=SQR(D-E)
C?
```

SOLVE TRIANGLE: Calculate third side of triangle, given two sides and included angle.

Value requests may not be combined with any other statement on one line.

How it works

The Micro Calc program is written almost entirely in BASIC. None of the floating-point variables named with a single letter is used in the program itself. This allows the user all 26 of these variables on the screen. When the F7 or "@" key is pressed each assignment statement is POKEd into a special area of memory called the input buffer. Then a BASIC ROM routine is called to tokenize the expression. Finally another ROM routine that assigns variables (the BASIC LET function) is called to evaluate the expressions. With the Commodore versions, the machine code is only 48 bytes.

The Atari version works a little differently. It uses an alternate screen on which you do your typing. Then, when you press the calculate key ("@"), the lines you have typed are copied to the actual BASIC screen (which is kept hidden from the user) and RETURNs are executed on each line to execute the statements in the immediate mode.

Atari BASIC is quite different from the BASICs on the other computers. The discussion below applies primarily to these other computers. Many of the things discussed will not work on the Atari. The Atari version has extra powers, such as IF...THEN and FOR...NEXT support, though. See the Atari section for details.

When to Clear the Variables

The latest version of Micro Calc allows you to clear the variables at your discretion. This process is only done automatically when the screen is cleared or or when a screen is SAVEd or LOADed. What are the advantages? If you type in the screen marked "DISTRIBUTE", you will see a good reason why the variables aren't cleared automatically on each calculation. Notice that the line labeled "BALANCE" at the top of the screen assigns a value to the variable B. This is where you type in the starting balance for your loan. The final calculation results in a new value for B. If you now go to the end of the top line and delete it entirely, the calculation will be performed using the B calculated in the *previous* calculation. Without automatic recalculation, variable M acts as a counter, incrementing once each time the calculation is repeated. If the first line is left intact, though, the same calculation will be repeated, and, assuming nothing is changed, all the variables except M will come up with the same values as the previous time. If you want to zero M, you can just hit the zeroing key (F8 on C-64 and VIC, double quote on the Apple, and CLEAR on the CoCo), or you can explicitly assign M a value of zero as part of a multiple statement on the first line.

Making a Decision without IF...THEN

The "DEC-TO-HEX" screen demonstrates how to make decisions without using IF...THEN (which is not allowed in Micro Calc). The problem we want to solve is how to get the same screen to work on both signed and unsigned decimal integers. There are two ways to look at a 16-bit binary number. If *unsigned* arithmetic is used, all 16 bits are used, so 1111 1111 1111 1111 is considered to be the equivalent of the decimal number 65535. If *signed* arithmetic is used, the most significant bit indicates whether the number is positive or negative. If the bit is on, the number is negative and the absolute value is determined by taking the two's complement. This same binary number that is 65535 in unsigned arithmetic is −1 in signed arithmetic.

The solution is to test for positive or negative within an arithmetic expression. This is done in the line labeled "SIGNED". The expression D<0 tests whether the original decimal number is negative. If it is, −1 is assigned to the expression, it's multiplied by −16, and 16 is added to the value of H, which is negative. What this really accomplishes is taking the two's complement of the most significant hex digit whenever the original decimal number is negative. The other three hex digits are calculated properly, whether the calculation is signed or unsigned. The Apple and Atari assign 1, instead of −1, to a true statement, so your calculations should reflect the difference. In this example, you would type H=H+(D<0)*16 for the Apple or Atari. This decision making capability is used similarly in the "HEX-TO-DEC" screen. The variable S is used as a flag: if it is less than 0, then the result is calculated as signed; if it is 0 or greater, then the result is calculated as unsigned. The same change must be made for Atari or Apple screens. Other applications of this decision-making ability would be testing a divisor to avoid a fatal ?DIVISION BY ZERO ERROR, and testing a counter to see if it has arrived at a specified maximum.

Getting More into Less Space

The Color Computer and VIC-20 versions of Micro Calc offer less space for calculation due to memory or screen-size limitations. On the VIC-20, each line is only 20 characters long, and on both computers there are fewer lines available. Two techniques may be used to get around these limitations.

Multiple statements may be used to perform two short assignments on the same line. For instance, in the "HEX-TO-DEC" screen, the statements D=D+J*16 and D=D+K may be combined into one line by separating them with a colon (semicolon on Commodore machines): D=D+J*16:D=D+K.

Statements that are too long to fit on one line may be broken into two separate statements by using an *intermediate result.* For instance, the statement J=INT(I*D*B*100+.5)/100 may be replaced (as is it was in the "DISTRIBUTE" screen) with two separate statements: J=I*D*B and J=INT(J*100+.5)/100.

Micro Calc Program Description

Notes on all programs

Of necessity, all of the comments in the following description do not necessarily apply to all of the programs. The reader is cautioned to take such comments as 'color' to apply only if your computer has the specified function.

Initialization (A)

The screen is cleared and the border and screen color set. The call to subroutine, READs in the bytes of the machine language program from the DATA statements and POKEs them into memory. A number of constants are defined, including the carriage return, delete, and other control characters. The number of lines is set and the arrays are dimensioned accordingly. A subroutine is called, which fills the arrays from the remaining DATA statements to make the sample

```
START BAL       B=
DAYS/PER.       D=
PAYMENT         P=
ANN %           I=
DAILY DEC       I=I/36500
INTEREST        J=I*D*B
(ROUND)         J=INT(J*100+.5)/100
TO PRINC        C=P-J
(ROUND)         C=INT(C*100+.5)/100
COUNTER         M=M+1
TO INT.         J?
TO PRINC.       C?
# PERIODS       M?
                B=B-C
NEW BAL         B?
```

The following screens require more than ten lines. See the text for techniques to squeeze more assignments into less space.

DISTRIBUTE: Calculate distribution of monthly payment to interest and principal. Enter the requested values for the starting balance B, days/period D, payment P, and annual interest rate I. To continue beyond the first month, perform the calculation once, then delete the first line. The new balance will be retained as the starting balance for the next calculation. The counter M will increment once for each calculation.

```
DEC INPUT       D=

MS DIGIT        H=INT(D/4096):I=D-H*4096
                J=INT(I/256):K=I-J*256
                L=INT(K/16)
LS DIGIT        M=K-L*16
SIGNED?         H=H-(D<0)*16

10=A            H?
11=B            J?
12=C            L?
13=D            M?
14=E
15=F
```

(Commodore use ;)

(Atari and Apple use: H = H + (D < 0)•16)

DEC-TO-HEX: Calculate hexadecimal equivalent of decimal integer in the range −32768 to 65535.

```
MS DIGIT        H=
                I=
                J=
LS DIGIT        K=
                D=H*4096
                D=D+I*256
                D=D+J*16
                D=D+K
SIGNED?         D=D+(H<0)*65536
DECIMAL         D?
A=10
B=11
C=12
D=13
E=14
F=15
```

(Atari and Apple use: D = D − (H < 0)•65536)

HEX-TO-DEC: Calculate decimal equivalent, given four-digit hex number. To interpret as negative number, precede first digit with - sign.

screen. The sample screen is displayed. Then the cursor is positioned for the first line, and a branch is made to enter the normal loop. If you want to skip loading the sample screen, delete the appropriate line.

Main Program Loop (B)

In general, the program consists of testing for characters from the keyboard. Certain characters are considered to be *control* characters and must be dispensed with in special ways. Some of these involve branching to subroutines; others are dealt with immediately. Characters that aren't control characters are either accepted and added to the end of the current line, or they are rejected. After each character is processed, the flow usually goes back to the main loop. If the next character begins a new line, then a branch is made to reposition the cursor.

Whenever the screen is cleared, the arrays are cleared and the screen blanked. If the new line contains a value request statement, then the line is blanked out to remove the printed value. The current line is printed, followed by the cursor. Characters from the keyboard are processed. Control characters are tested and other characters are added to the current line.

If the character causes the length of the line to be exceeded, then a cursor down or return is executed. Then the line number counter is incremented and tested. If maximum lines has been exceeded, then the necessary adjustments are made to start work on the first line. Otherwise, the cursor is positioned at the beginning of the next line.

Delete is handled by checking for an empty line. The necessary screen display parameters are set, and strings are adjusted with the LEFT$( ) function.

The up-cursor character is handled in the following manner. If the new line is 0 then the line counter is set to maximum lines, and the cursor is adjusted accordingly. Otherwise, the cursor is moved up one line.

The calculation command branches to a subroutine which handles the calculation and printing the results. The cursor is positioned at the end of the top line after the calculation.

Comment Field Handling (C)

The operation here is very similar to that of the calculation field. Things are simpler, though, since nearly every character is is allowed. Everything is printed in light green, and the remainder of each line is left unreversed.

Input Subroutine (D)

This is called by the main editor program. The various control characters are tested, then for the other characters. Acceptable ones cause return, while unacceptable ones fall through to get another character. This continues until an acceptable character is received.

Calculation Processing (E)

The calculation process may take several seconds, depending on the screen contents. Each line is examined with value request statements handled by one subroutine. The requested variable is placed into the appropriate element of the string array. An illegal line, with fewer than three characters, is skipped. Other lines are handled by the subroutine where the values are assigned by the machine language routine.

Assignment Handling (F)

Each assignment line is POKEd into the input buffer, character by character. Whenever a colon (semicolon) is encountered, that line is processed (a zero is POKEd for the Commodore programs). When the end of the line is reached, the machine-language routine is executed, and the RETURN goes back to the calling routine.

Get Character (G)

The GET function accepts any character from the keyboard. If there is no character, then the program loops until there is.

File Handling (H)

The subroutine handles SAVEing the calculation and comment arrays. Similarily, the LOAD subroutine handles loading these arrays from tape or disk. Some of the programs check for disk errors and print error messages or allows a new file name.

The prompt portion sets up a number of variables according to the responses. A disk or tape variable is set and appropriate file name strings established to either READ or WRITE a sequential file. Then the proper channel is opened and the arrays either read from or written to the output device. (The Commodore routine includes an additional subroutine to reread the machine-language routine back into the cassette buffer, since all or part of it was destroyed during the file operation.) Then the screen is redisplayed, containing the old contents if it was a SAVE operation, and the new contents if it was a LOAD.

Value Request Processing (I)

As each line is processed in the subroutine, only lines ending in a "?" are sent here. Therefore, the first character is the variable name. Each letter is converted into a number from 1 to 26. Flow of control is passed by the ON...GOSUB structure with 26 possible branches. The rest of this subroutine consists of small subroutines, one for each letter of the alphabet. The value of the named variable is assigned to the appropriate element of the value array.

Screen Print With Values (J)

This causes the whole screen to be printed with values shown after each "?" The values are converted to a string using the STR$( ) function, and the remainder of the line is filled out with the appropriate number of reversed underline characters.

Read Machine Language Routine (K)

The DATA statements contain the individual bytes of the machine-language program. The READ statement is used for each byte, and it is POKEd into succesive bytes of memory. This routine is called once at the start of the program, and (for Commodore) again whenever a LOAD or SAVE operation takes place.

Print Screen (L)

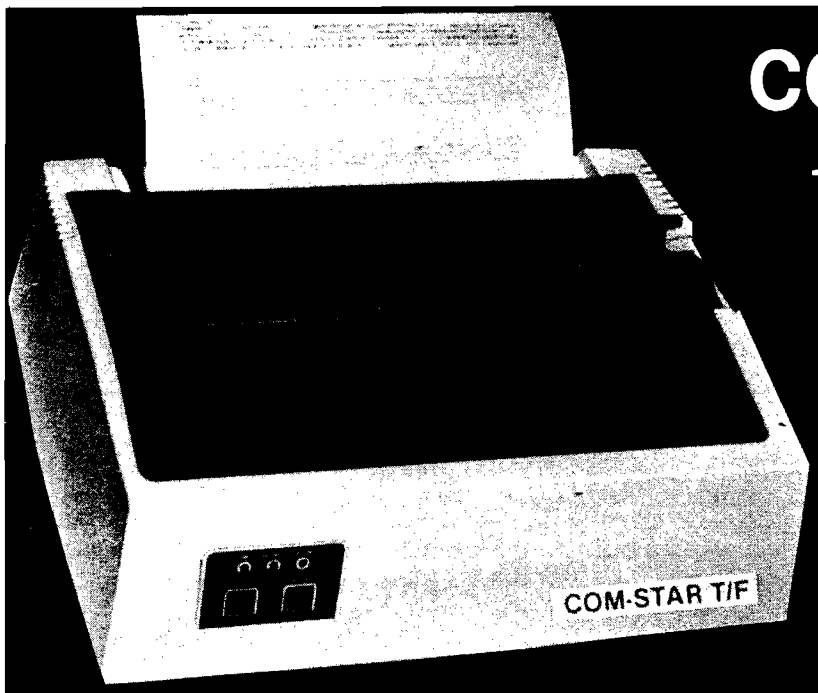
The first time the subroutine is called, it prints the standard start-up screen. Later, the screen is printed with current array values and comments. Each screen line is printed according to the contents of the arrays. The current file name, if any, is printed at the bottom of the screen.

Clear User Variables (M)

Each of the user variables is set to zero. This routine is used when using successive calculations, such as A = A + 1, to start over with different values.

The rest of the program consists of the DATA statements used for the machine-language program and the initial screen contents. MICRO™

80 COLUMN PRINTER SALE—$149.00*



COM-STAR T/F

Tractor
Friction
Printer

only $219

(Parallel or Serial)

• Lowest price quality tractor friction printer in the U.S.A. • Fast 80 characters per second
• 40, 46, 66, 80, 96, or 132 characters per line spacing • Prints labels, letters, graphs, and tables
• List your programs • Print out data from modem services

*STX-80 COLUMN PRINTER—$149.00

Prints full 80 columns. Super silent operation, 60 CPS, prints Hi-resolution graphics and block graphics, expanded character set, exceptionally clear characters, fantastic print quality, uses inexpensive thermal roll paper!

DELUXE COMSTAR T/F PRINTER—$219.00

The Comstar T/F is an excellent addition to any micro-computer system. (Interfaces are available for Apple, VIC-20, Commodore-64, Pet, Atari 400 and 800, and Hewlett Packard). At only $219 the Comstar gives you print quality and features found only on printers costing twice as much. Compare these features.

• **BI-DIRECTIONAL PRINTING** with a LOGIC SEEKING CARRIAGE CONTROL for higher through-put in actual text printing. 80 characters per second.

• **PRINTING VERSATILITY:** standard 96 ASCII character set plus block graphics and International scripts. An EPROM character generator includes up to 224 characters.

• **INTERFACE FLEXIBILITY:** Centronics is standard. Options include EIA RS232C, 20mA Current Loop.

• **LONG LIFE PRINT HEAD:** 100 million character life expectancy.

• **THREE SELECTABLE LINE SPACINGS:** 6, 8 or 12 lines per inch.

• **THREE SELECTABLE CHARACTER PITCHES:** • 10, 12 or 16.5 characters per inch. 132 columns maximum. Double-width font also is standard for each character pitch.

• **PROGRAMMABLE LINE FEED:** programmable length from 1/144 to 255/144 inches.

• **VERTICAL FORMAT CONTROL:** programmable form length up to 127 lines, useful for short or over-sized preprinted forms.

• **FRICTION AND TRACTOR FEED:** will accept single sheet paper.

• **224 TOTAL CHARACTERS**

• **USES STANDARD SIZE PAPER**

If you want more try —

Premium Quality COMSTAR T/F SUPER-10X PRINTER—$299.00

More Features Than RX-80

For $299 you get all of the features of the Comstar T/F plus 10" carriage 120 cps, 9 x 9 dot matrix with double strike capability for 18 x 18 dot matrix. High resolution bit image (120 x 144 dot matrix), underlining, backspacing, left and right margin settings, true lower descenders, with super and subscripts, and prints standard, Italic, Block Graphics, special characters, plus 2K of user definable characters. For the ultimate in price performance the Comstar T/F Super 10" leads the pack!

Double Immediate Replacement Warranty

We have doubled the normal 90 day warranty to 180 days. Therefore if your printer fails within "180 days" from the date of purchase you simply send your printer to us via United Parcel Service, prepaid. We will IMMEDIATELY send you a replacement printer at no charge via United Parcel Service, prepaid. This warranty, once again, proves that WE LOVE OUR CUSTOMERS!

15 DAY FREE TRIAL

OTHER OPTIONS

Extra Ribbons	$ 5.95
Roll Paper Holder	32.95
Roll Paper	4.95
5000 Labels	19.95
1100 Sheets Fan Fold Paper	13.95

Add $17.50 shipping, handling and insurance. Illinois residents please add 6% tax. Add $40.00 for CANADA, PUERTO RICO, HAWAII, ALASKA orders. WE DO NOT EXPORT TO OTHER COUNTRIES. Enclose cashiers check, money order or personal check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail available!! Canada orders must be in U.S. dollars.

PROTECTO
ENTERPRIZES (WE LOVE OUR CUSTOMERS)
BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

SUPER-10" ABCDEFGHIJKLMNOPQRSTUVWXYZ
ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890

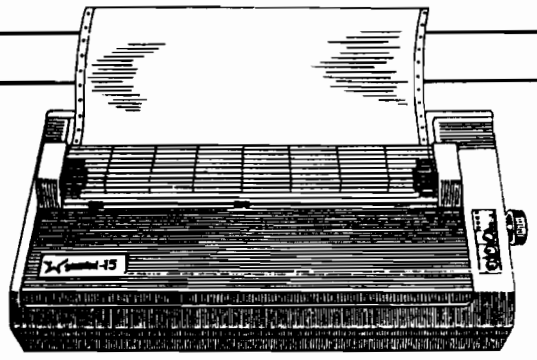
GRAFSTAR (Apple interface card - "Like Grappler Plus") SALE $69.00.

SUPER COM-STAR T/F 15″ PRINTER SALE $379.00

NOW YOU CAN BUY A TRACTOR-FRICTION 15″ CARRIAGE PRINTER FOR AN INCREDIBLE $379.00

	STAR COMSTAR 15″	EPSON* MX-100/Type III	OKIDATA* MICROLINE 83A
Warranty This refers to standard warranties. Coverage varies by manufacturer. Complete information is available at your place of purchase	180 day	90 day	90 day
Buffer	2.3 K	1 Line	1 Line
CPS	100	80	120
CPI	10.12.17	10.17	10.17
Over 80 Type Fonts	Yes	No	No
Block Graphics	Yes	No	No
Special Character Sets Available	Yes	No	No
Roll Paper Holder Standard	Yes	No	No
Suggested Retail Price Prices shown are based on data from each manufacturer. Actual retail prices may vary	$599.	$749.	$899.
Special Price	**$379.**		

*Epson and Okidata are registered trademarks of Epson America, Inc. and Okidata Corp., respectively.



Now you can save big on the Com-Star 15″ dot matrix printer from Protecto, while they last!

It's the 15½″ carriage printer that combines the most advanced features with a price that's better than ever. And component by component, Com-Star 15″ is so reliable, Protecto can warranty it for up to twice as long as its major competitors.

The popular Com-Star 15″. A great printer at a truly great price. When you add it up, it figures to be a super sale.

• LOWEST PRICES • 15 DAY FREE TRIAL • 90 DAY FREE REPLACEMENT WARRANTY
• BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL • FREE CATALOGS

Add $17.50 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $35.00 for CANADA, PUERTO RICO, HAWAII orders. WE DO NOT EXPORT TO OTHER COUNTRIES.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Canada orders must be in U.S. dollars. VISA — MASTER CARD — C.O.D.

PROTECTO
ENTERPRIZES (WE LOVE OUR CUSTOMERS)
BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

Spreadsheets

by Phil Daley

What does a Spreadsheet Do?

There are many software packages on the market today which have a multitude of uses for business and accounting applications, whether you own a multi-national conglomerate or are managing your own checkbook. Some are fill-in-the-blank accounting programs designed with a specific job or a specific set of jobs in mind. These are usually known as accounting packages: general ledger, accounts receivable/payable, payroll, and others. Many are designed to be general in nature, so that you can program your own particular functions into the software. These are spreadsheet packages that can do accounting functions, as a well as act as a mini data-base. Some are designed to be project oriented with specific abilities to organize and layout planning strategies. Others are designed to be multi-purpose with planning and spreadsheet capabilities combined.

This month we plan to concentrate on spreadsheet packages — how do they work, what do they offer and who can benefit from them. While each product has its own syntax and specifications, many of the features can be found on all of the spreadsheet packages and a look at the generic options will give you an overview of what they can do.

The standard display screen is a series of columns (normally designated alphabetically) and rows (normally desiginated numerically), blank at the beginning. Each intersection of row and column has a name (A1,C67,GG145...). These individual blocks are called "cells".

Each cell can contain one piece of information. You can define the size and type of the individual cells, or whole columns or rows. The size parameter can help save space on the screen by keeping the columns close together. The type parameters (such as Label, Integer, $, left or right justified...) help prevent input errors and neaten the appearance of the screen format. Each cell can be a number (value), name (label) or computation (formula). Values can be positive or negative, integer quantities or floating-point constants; labels can be names or numbers; formulae can contain any of the allowable computations grouped in any desired manner by use of parentheses. Cells can also reference other cells by name. If cell D8 contained B5, then the value of D8 would be the same as the value of B5. If it contained @SUM(A1...A124), then the value of D8 would be the sum of the values contained in all the cells from A1 through A124.

The real advantage to an electronic spreadsheet program is the instant feedback for each calculation entered. Upon entering a formula, the spreadsheet is immediately recalculated (assuming recalculation is turned on), and the value presented on the screen. This gives you a 'rough estimate' glance to see if the formula is at least in the ballpark. Normally, when writing a program to perform calculations, you don't get a chance to see the output of any particular formula until you run the whole program, or at the minimum, a compilable module.

Basic Functions

In addition to the standard + – * / < > and ^, most spreadsheet programs contain functions similar to the following:

@ABS	Return absolute value
@AND	Return **TRUE** if *all* **TRUE**
@AVERAGE	Calculate mean of list
@EXP	Raise *e* to a power
@FALSE	Return **FALSE**
@IF	Select value based on condition
@INT	Truncate value
@LN	Return natural log
@LOG	Return log base 10
@MAX	Return maximum value in list
@MIN	Return minimum value in list
@NOT	Return **TRUE** if **FALSE** else **FALSE**
@NPV	Calculate Net Present Value of list at discount rate
@OR	Return **TRUE** if *any* **TRUE**
@PI	Return value of Pi
@ROUND	Round a number to specified places
@SQRT	Return the square root
@SUM	Calculate the sum of a list
@TRUE	Return **TRUE**

Many of the newer spreadsheets also contain transcendental functions, standard deviation, internal rate of return and other specialized accounting functions.

@ACOS	Arc-cosine function
@ASIN	Arc-sine function
@ATAN	Arc-tangent function
@COS	Cosine function
@DIF	Calculate the difference of a list
@FRA	Return the fractional part of expression
@IRR	Return the internal rate of return
@PDIF	Return the percentage difference
@SIN	Sine function
@STDDEV	Return standard deviation of a list
@TAN	Tangent function

There are many different commands to operate on a worksheet, I will mention a few that should be considered when buying a spreadsheet software package. All have cursor movement from column to column, and row to row. Some allow movement to rows and columns by multiple movements. In addition to particular formats mentioned above, some work sheets allow "hidden" cells (the data is not displayed). You should be able to set the width of columns. The replicate command should have a "relative" copy, to copy cells with row and column orientation included. Most work sheets allow "windows", either horizontal or vertical screen splitting, with synchronous or unsynchronous scrolling. Some packages allow a "data save" to a textfile that can be operated on by a BASIC program, data base manager, mailing list or text processor.

Who Can Benefit?

There are many uses for a spreadsheet package, ranging from storing data in lists to complicated accounting procedures. The main limitation on all home computers is the amount of available RAM to store the input information. In addition to the memory consumed by the program, most of the spreadsheet programs require that *all* of the data be in memory *all* of the time. Even with the efficiency of storing data in a compact format, it doesn't take a very great number of columns and rows to deplete a 64K machine. (One new program, *Multiplan*, implements a "virtual memory" system which allows spreadsheet data to be as large as available disk space by swapping into and out of memory, the sections of the data currently being used.) This explains all of the interest in additional RAM cards to increase the available RAM to 128K or more. Some spreadsheets allow up to 512K of additional memory.

Very specific applications that can be expected to remain unchanged, such as maintaining a checkbook, are probably handled more easily with a dedicated program. A spreadsheet is very useful for applications that change often, being easily modifiable, and with instantaneous feedback as to the correctness of the calculations. Some applications, such as your income tax, change with each use. A spreadsheet set up to calculate your Form 1040 could be easily modified each year to account for changes in the form by the IRS. A dedicated program designed for a particular year would have to be rewritten each subsequent year.

Small database applications — for instance, lists of names, addresses and telephone numbers — can be easily maintained and sorted (only newer products have automatic sorting capability) by zip, last name, etc.

The ability to see and adjust the screen formatting is also very useful. Printed output for monthly finances, budgets and other reports is quickly generated and the templates can be used again each time a new report is needed.

Anyone connected with a statistically-oriented team (what sport isn't?) can keep records and all the associated stats easily and make updates quickly and effortlessly. Bowling league, Little League and local school teams can benefit from accurate reports generated on a timely basis.

Some Samples of Use

Spreadsheets can be used for tracking expeditures against a proposed budget. Many companies have to allocate an advertising budget amongst several different media and products. It is a simple matter to design an overall budget plan and then juggle figures on the spreadsheet to develop a good mix between emphasized products and target audiences. Recalculation of the budget totals is swift and feedback of the effects of various strategies is essentially, instantaneous.

Bid preparation can be handled well on a spreadsheet. Since pinning down all the expenses is very difficult, and profit margin depends a great deal on the accuracy of the bidding, a tool for juggling the numbers facily is a great

Program	Manufacturer		City, State	ZIP	Computer
A Financial Wizard 1.5	ON LINE Computer Centers	10944 North May	Oklahoma City, OK	73120	Atari
Accountant	Decision Support Software	1438 Ironwood Drive	McLean, VA	22101	Apple
BusiCalc	Skyles Electric Works	2316 South Whisman Road	Mountain View, CA	94041	Pet/C64/Vic
Business Planner	Duosoft Corporation	1803 Woodfield Drive	Savoy, IL	61874	Apple
Business Planning Tool	Sofstar	13935 Highway 1	Juno Beach, FL	33408	Apple
Calc Result	Computer Marketing Services	300 West Marlton Pike	Cherry Hill, NJ	08002	C-64
CalcStar	MicroPro International	33 San Pablo Avenue	San Rafael, CA	94903	Apple CP/M
Desktop PLAN	Visicorp	2895 Zanker Road	San Jose, CA	95134	Apple
DYNACALC	Computer Systems Center	13461 Olive Blvd.	Chesterfield, MO	63017	Flex
Elite‡Calc	Elite Software	Box 11224	Pittsburgh, PA	15238	TRS-80C
FCS-EPS	EPS	One Industrial Drive	Windham, NH	03087	Apple
Financial Analysis	Aeronca /Execuware	4530 Park Rd., Suite 340	Charlotte, NC	28209	Apple
Financial Modeling	Georgia Tech Research	225 North Avenue	Atlanta, GA	30332	Apple CP/M
Financial Planner	Ashton-Tate	10150 W. Jefferson Blvd.	Culver City, CA	90230	Apple CP/M
FORECAST	Northwest Analytical	1532 Southwest Morrison	Portland, OR	97205	Apple CP/M
LogiCalc	Software Products International	10343 Roselle Street	San Diego, CA	92121	Apple
MAGICALC	Artsci	5547 Satsuma Avenue	North Hollywood, CA	91601	Apple
microFINESSE	The P-E Consulting Group	Park House, Egham	Surrey, England		Apple
Micro-DSS/Finance	Ferox Microsystems	1701 N. Fort Meyer Dr.	Arlington, VA	22209	Apple
Multiplan	Microsoft Corporation	10700 Northrup Way	Bellvue, WA	98004	Apple/C64
Optimizer	Supersoft	P.O. Box 1628	Champaign, IL	61820	Apple
PeachCalc	Peachtree Software	3445 Peachtree Rd. NE	Atlanta, GA	30326	Apple CP/M
PLAN80	Digital Marketing Corporation	2363 Boulevard Circle	Walnut Creek, CA	94595	Apple CP/M
Senior Analyst II	Apple Computer Company	20525 Mariana Avenue	Cupertino, CA	95014	Apple
Spectaculator	Radio Shack	300 One Tandy Center	Fort Worth, TX	76102	TRS-80C
Super "Color" Calc	Nelson Software Systems	9072 Lyndale Avenue,So.	Minneapolis, MN	55420	TRS-80C
SuperCalc	Sorcim Corporation	2310 Lundy Avenue	San Jose, CA	95131	Apple CP/M
TABULA RASA	Computer Systems Consultants	1454 Latta Lane	Conyers, GA	30207	Flex
Visicalc	VisiCorp	2895 Zanker Road	San Jose, CA	95134	Apple/Pet
VI-CALC	United Microware Industries	3503-C Temple Avenue	Pomona, CA	91768	Vic

help in maximizing profit. All of the various factors — consulting, labor, equipment, materials and subcontracting — can be charted, with considerable "*what – if*ing" being done with the figures to arrive at an appropriate idea of expenses. The expected margin of profit can be added with some assurance that the final figures have taken a good deal of the risks involved into consideration.

Many small accounting type problems can be easily solved without resorting to large, unwieldy, fixed-format accounting packages. This is especially true of smaller businessmen who might not want to spend a lot of money for special accounting software that would need to be tailored to their own particular business. A spreadsheet can do multiple checkbooks, prepare invoices and purchase orders, track accounts payable and receiveable, and a multitude of other accounting functions. While none of the "just" spreadsheet programs can match a fully developed accounting system, some of the newer spreadsheets can do most, if not all, of the job.

At MICRO, we use a spreadsheet program for much of our work involving simple accounting procedures and data-base management. It keeps all of our paper work under control and we only have to enter names and addresses one time, with everyone sharing the files for additional uses. For instance, to help with the organization of the advertising department, we have a list of advertisers, such as figure 1, which can include such information, in addition to names and addresses, as account number, advertising pages, page size, page rate, commissions, and sales regions.

From such a master list, it is a simple matter to sort the list monthly by current page size, deleting the accounts that are currently inactive, alphabetizing the remaining accounts for a monthly advertising summary. The next step is to sort the list by region (figure 2) so that each sales representative can see the totals for his region and in comparison to the other regions. We can add magazine page numbers to the list (figure 3) and dump the list to a text file, instead of the printer, and transfer it to the typesetter to compose the advertisers' index without rekeying all the names. (See MICRO 59:54 for furthers details of our typesetting communications.)

A quick look at a work sheet to figure monthly income-expenses shows how simple a work sheet template can be, but still have a useful function. With a minimum of effort, each month, a quick summary can be prepared by entering the few necessary figures.

Figure 4 illustrates how formulae are stored in cells. The third column (C) contains the formula for the sum of cells Dn and En where n is the row number. This column is easily filled in by defining the formula in cell C5, and then, using the Replicate command with "relative" values, copying that cell into C6...C78. The formula in C80 is the sum of column C, and that is replicated into D and E. The screen display can show either the calculated values or the actual formulae. This is set from the command line.

This figure is a split example to demonstrate the relationship between formulae in the cells and the values in the cells. The chart was printed out once in formula dump mode and once regularly and then superimposed to give the illusion of the formulae being present with the values.

Record keeping for a team such as baseball or soccer is easily managed on a work sheet (figure 6). Such a list can easily be sorted by last name for a team roster, by birthdate for eligibilty and yearly updating, by phone numbers for a telephone tree, by zip code for a mailing list or by other factors such as individual game statistics or personal factors.

Other topics that lend themselves to worksheet solving include accounts receiveable ageing, invoicing from inventory, cost recovery, production scheduling, estimating, checkbook ledger, engineering formulae, accounts payable, payroll reporting, monthly sales reporting, daily inventory and financial forecasting.

Advanced Uses

There are additional features to be found on most spreadsheet packages. These are more complicated to use and require a deeper understanding of how a worksheet functions. These include, but are not limited to:

@CHOOSE	Returns the value of a particular cell
@COL	Returns the current column number
@COUNT	Returns the number of cells in a range
@ERROR	Returns error message
@INDEX	Returns value next to match
@ISERROR	Returns TRUE if ERROR, otherwise FALSE
@ISNA	Returns TRUE if NA, otherwise FALSE
@LOOKUP	Returns value less than or equal to match
@NA	Returns NOT AVAILABLE error
@ROW	Returns the current row number

The @CHOOSE function is useful for selecting a value from a pre-determined list. It is similar to @LOOKUP, except that the table does not have to be defined in the worksheet proper. For instance, if you knew that in Trial 1 you wanted to use an interest rate of 11.5%, in Trial 2 you would use 13.5%, and in Trial 3 you would use 17.875%, by defining cell C4 to contain the particular number of the trial you are running, the following formula can be used anywhere in the sheet to substitute for the appropriate interest rate:

@CHOOSE(C4,.115,.135,.17875)

When cell C4 contains a 1, the value returned is .115, if it has a 2, then the values is .135, and if it has a 3, the value is .17875.

The @COL function is useful for indexing items that ascend by increments of one, such as dates. The value for column A is 1, B is 2 and so on. If you replicate a formula such as

1982 + @COL

across the top of the worksheet, you will quickly generate a yearly sequence.

The @ROW function is useful for indexing items that ascend by increments of one, such as counters. The value for row 1 is 1, 2 is 2 and so on. If you replicate a formula such as

@ROW-6

down the side of the worksheet starting in row 7, you will quickly generate a numbered list.

The @COUNT function is useful for determining *n*, the number of items used in calculating a particular formula. Such a determination is necessary in many statistical analyses, such as NPV or STDDEV. @COUNT includes only values in the specified range, it does not

```
[          A                ][ B  ][ C   ][ D  ][ E  ][ F    ][ G  ]
1-[ADV-M02]                    Advertising Pages --- Advertiser by Month
2-                                        Year
3-   Advertiser                  A/C #    1983    O65    N66  Contract     Reg
4-
5-AB Computers                  2ABCOM      2      1      1     3/84       B2
6-Acorn Software Systems        4ACORN     .5    .25    .25                W4
7-Alternative Energy Products   4ALTER      1     .5     .5     6/84       W4
8-Amplify                       3AMPLI     .5    .25    .25     4/84       K3
9-Apogee                        4HOLLY    .33    .33     0                 W4
10-Apple Tree Electronics       4APPLE    .34    .17    .17     4/84       W4
11-Arbutus Total Soft           4ARBUT      1     .5     .5      6X        W4
12-Ark Computers                4ARK        1      1      0     4/84       W4
13-Artsci                       4SOFTE      1      0      1                W4
14-Atari Home Computers                     1      0      1    10/84       H2
15-Atari Program Exchange                   1      0      1                W4
16-Aurora Software              3AUROR    .17    .17      0      6X        K3
17-Avalon Hill                              1      1      0                D2
18-Check-Mate                   1CHECK      2      1      1     12X        H1
19-Commander Magazine           4COMMA    .25    .25      0      6X        W4
20-Communications Electronics   [illegible]EC      1      1      0                K3
21-Co[illegible]                          .66    [illegible]    .33    2/84       W4
22[illegible]sition                       [illegible]      0
```

Figure 1. A sample worksheet of advertisers

```
[          A                ][ B  ][ C   ][ D  ][ E  ][ F    ][ G  ]
1-[ADV-M02]                    Advertising Pages --- Advertiser by Month
2-                                        Year
3-   Advertiser                  A/C #    1983    O65    N66  Contract     Reg
4-
5-Check-Mate                    1CHECK      2      1      1     12X        H1
6-Computer Exposition           1C        .66    .66      0                D1
7-Computer Science Engineering  1CSCIE    .25    .25      0                H1
8-Granite Computer Sales        1GRANI    .25    .25      0
9-Inter-Action                  1INTER      1     .5     .5     6/84       H1
10-Leading Edge                 1LEADI      2      1      1                H1
11-Micro Sparc                  1M          2      1      1                D1
12-Microbits                    1M        .33    .33      0                H1
13-Monarch Data Systems         1MONAR    .25    .25      0     5/84       H1
14-Omega Sales International     1OMEGA      1      1      0     12X        H1
15-Performance Micro Products   1PERFO     .5    .25    .25     5/84       H1
16-Pion Inc                     1PION       1     .5     .5      3X        H1
17-Richvale Telecommunications  1RICHV      2      1      1                K1
18-----------------------------------------------------------------------
19-Avalon Hill                  2A          1      1      0                D2
20-AB Computers                 2ABCOM      2      1      1     3/84       B2
21-Computer Mail Order          2CMAIL      4      2      2    2p12X       B2
22-Computer Ma[illegible]       2CMKTG      2      1      1     3/84       H2
23-Comp[illegible]ciation 2C                  .33    .25                H2
                                              1
```

Figure 2. Sample worksheet sorted by account region

```
[          A                ][ B  ]
1-[ADV-M02]                    Page #
2-
3-   Advertiser
4-
5-AB Computers                  93
6-Acorn Software Systems        10
7-Alternative Energy Products   94
8-Amplify                       20
9-Apogee                        100
10-Apple Tree Electronics       92
11-Arbutus Total Soft           6
12-Ark Computers                100
13-Aurora Software              45
14-Avalon Hill                  57
15-Check-Mate                   10
16-Commander Magazine           19
17-Communications Electronics   23
18-CompuTech                    8
19-Computer Exposition          70
20-Computer Mail Order          84-85
21-Computer Marketing           1
22-Computer Sci[illegible]
23-Comp[illegible]
```

Figure 3. Sample worksheet with pages added for index

count labels or blanks. You can specify a list, range or list of ranges in the argument.

The @LOOKUP function is very useful to read elements of a table included in the worksheet. Suppose a software package had the following price based on quantity:

Quantity	Price/Package
100	22.95
300	17.95
500	14.95
800	11.95
000	9.95

This would be entered in the worksheet in two adjacent columns, say D and E. The price per document can be entered anywhere in the worksheet by the formula:

@LOOKUP(B2,D1...D5)

The @LOOKUP function would determine the quantity ordered from cell B2, say 650, and then skim through the D1...D5 column looking for a value larger than the current value. When it finds one, in this case at D4, it then backs up one entry and reads the value in the next adjacent column, here 14.95 (from E3), and returns with this value. It is also possible to specify the range to be searched as a row, and the value will be taken from the row below the searched row.

The @INDEX function is the same as the @LOOKUP function (cf.), except that an exact match is required.

The @ERROR function is used in several different ways. @ERROR displays the word "ERROR" in the current cell, and in any cells with formula references to that cell. It can be used in tables with CHOOSE, INDEX or LOOKUP formulae to screen out invalid table entries. It can also be used in combination with an @IF statement to exclude certain values from an acceptable range. For instance, if you wanted to sum a range of numbers only if A1 was in the range of 50-100, then the following formula could be used:

@IF(@AND(A1..=50,A1..=100),@SUM(B1...B20),
@ERROR)

This would check the value of A1 before evaluating the formula and would return "ERROR" if the value was outside the specified range.

The @NA function is used for template generating. All of the cells which require entered data are first flagged with @NA. Later, after the data has been entered, a simple test can be used to check to see if all the cells have been updated.

The @ISERROR function tests any type of argument and returns TRUE if the argument is an ERROR condition, false if it is not an ERROR. This is a good way to test whether one or more calculations has produced an error:

@IF(@ISERROR(A3),0,A3*B5)

The @ISNA function tests any type of argument and returns TRUE if the argument is a NOT AVAILABLE condition, false if it is available. This is a good way of making calculations conditional on the availability of data:

@IF(@ISNA(A3),@NA,A3*B5)

Some fairly complicated worksheets can be developed using these advanced features.

Memory Considerations

The particular spreadsheet that we use is a Flex-based system called *Dynacalc*. Since that is the system that I am most familiar with, I will describe some of the working techniques of that system, assuming that all work-sheet programs must use an overall somewhat similar system, while perhaps differing on some of the fine points.

This program allows 256 columns and 256 rows, not both at once, as it would require 128K bytes just to address all of the cells. A cell table is set up with a two-byte address for each cell in use (sometimes not in use, as we shall see). The cell table can hold 7680 entries, which means that you can address to cell AD256, for a tall worksheet with many rows, or to cell IV30, for a long worksheet with many columns, or any combination in-between, as long as the total figure *ROW*COL* doesn't exceed 7680.

Each entry in the cell table requires a two byte address. If you GOTO cell AD256 and enter a single character, you will have consumed 15360 bytes for cell addresses and one byte for the label. The program allocates all cells horizontally and vertically up to the largest address in use. However, it doesn't subtract cells from the table ever. If you have overflowed memory and deleted several rows, you won't get the full benefit of extra memory until you /S SAVE the file and /S LOAD the file back into the system after /Clearing the workspace. When the file is read back into memory, the unused rows and/or columns will not be allocated in the cell table with the resultant saving in memory.

Each value uses 10 bytes of memory, even "0". A cell reference in another cell also uses 10 bytes. Placing a B1 in cell A2 uses 10 bytes of memory. Labels use only one byte per character. Therefore, if you have a numerical sequence of labels "1", "2", "3" ..., it is much more memory efficient to enter them with a leading ' (single quote) to assure that the worksheet considers them to be a label.

A calculation (@SUM(A1...etc.)) starts at 10 bytes and consumes additional memory depending upon how long the calculation is. Therfore, if you are running out of room, it will save space to put an often used calculation in one cell, and reference it from other cells. If you are really desperate for a few extra bytes, replacing a formula by its value will also save space, at the expense of recalculation time, if any of the values in the formula change. You would have to re-enter the formula to recalculate the results (or do it by hand).

Visual display

Stop me if you've heard this one before — you really need 80 columns to be able to see a reasonable portion of the worksheet. Some of the new video boards (for Apple anyway) allow a display of 132 columns on-screen. The more you can see, the easier and faster it is to work with the program. The less scrolling that you have to do, the better. I use the GOTO *cell* command a lot because it is much faster than scrolling row by row or column by column to the desired site. I often add 20 to the desired cell number so that the cell is located in the upper portion of the screen when I get there.

When in the formula dump mode (formulae displayed on the screen instead of values), the formula is only printed to the width of the column. Often, this is not wide enough to see the whole formula on the screen, or printer. To enable printing of the whole formula, it is necessary to widen the columns containing the formulae somewhat. This is only necessary for dumping the formulae to a

printer. I often put the printer into compressed mode, to get the greatest number of columns on the paper at once. With a 15 inch carriage, you can print about 230 characters across.

I don't know how I functioned B.S. (that's before spreadsheets). I would recommend just about anyone who owns a computer to try out **MICROCalc** (elsewhere in this issue), and if you really need the power and memory of a full-size spreadsheet, buy one of the many software packages available. A list of the spreadsheets available for the computers we normally cover is listed at the end of this article. Happy calculating!

	A	B	C	D	E	F
3-Dealer		A/C#	Totals	WWA	M/A	BOM
4-						
5-Abacus North		99501A	@SUM(D5...F5)	75+150	225 30	30 12+9 21
6-A.P.P.L.E.		98032A	@SUM(D6...F6)	360+314+314+157	1145 187+187+187+2995	3556 54+54+5 113
7-Brodart Staceys		17705A	@SUM(D7...F7)	1990+4990	6980 86+135	221 18 18
8-Clinton Computer		20735A	@SUM(D8...F8)	4809	4809 141	141 77 77
9-Computer Shoppe		70002A	@SUM(D9...F9)	94	94 87+23+54	164 46+65 111
10-Computer Store		74105A	@SUM(D10...F10)	63+157	220 35+144	179 122+65 187
11-Comp. Market HI		96813A	@SUM(D11...F11)	1286	1286 31+31+31+76	169 67+54+90 211
12-Data Bank Fremont		94536A	@SUM(D12...F12)	360+210	570 520	520 79+52+9 140
13-Data Base		46805A	@SUM(D13...F13)	94+94+38	226 16+31	47 56+98 154
14-Data Domain		60195A	@SUM(D14...F14)	345+69	414 49+65+98+123	335 56 56
15-Esd Labortory		JAP-01	@SUM(D15...F15)	188+100+94	382 432	432 29+87 116
16-Farnsworth Comp.		60505A	@SUM(D16...F16)	361	361 86+54+67+32	239 87 87
17-Intergrated DP		CAN-02	@SUM(D17...F17)	144+144	288 235	235 234+76+123 433
18-Kroch's&Brentanos		60603D	@SUM(D18...F18)	157	157 72+72+79	223 76+78+98 252
19-Malibu Microcomp		90265A	@SUM(D19...F19)	144+58	202 144+187+100+125+115+300	15+65+90 170
20-Micro Chip		48084C	@SUM(D20...F20)	94+94+20	208 87+45+987	1119 21 21
21-Micro Computer		45459A	@SUM(D21...F21)	188+188+188+176	740 16+87	103 76+29+7 112
22-Micron Dist.		CAN-21	@SUM(D22...F22)	720	720 1180+100+100+100+240	72+32+80 184
23-Opamp Tech.Books		90038A	@SUM(D23...F23)	207+157	364 87+87+87	261 28+35 63
24-Pandasoft		GER-04	@SUM(D24...F24)	198	198 500+374+624+624+624	2746 78+54 132
25-Software Masters		60204D	@SUM(D25...F25)	157	157 249+249+250+250+250	1248 25 25
26-Software Store		33612B	@SUM(D26...F26)	62+157+62+157	438 129+40	169 76+36 112
27-Telecom Library		10011B	@SUM(D27...F27)	235	235 (115.2+186.75+72)*3	1122 45+65 110
28-Timecore		02114A	@SUM(D28...F28)	188+188+188	564 113	113 34+24+98+485
29-World Wide Media		10016D	@SUM(D29...F29)	235	235 4066+10+20+150+6000+500	70+6+3+875+105
30-						
31-TOTALS			@SUM(C4...C30)	@SUM(D4...D30)	21208 @SUM(E4...E30)	26809 @SUM(F4...F30)

Figure 4. **A sample worksheet with formula dump on. An overlay of some of the values is superimposed.**

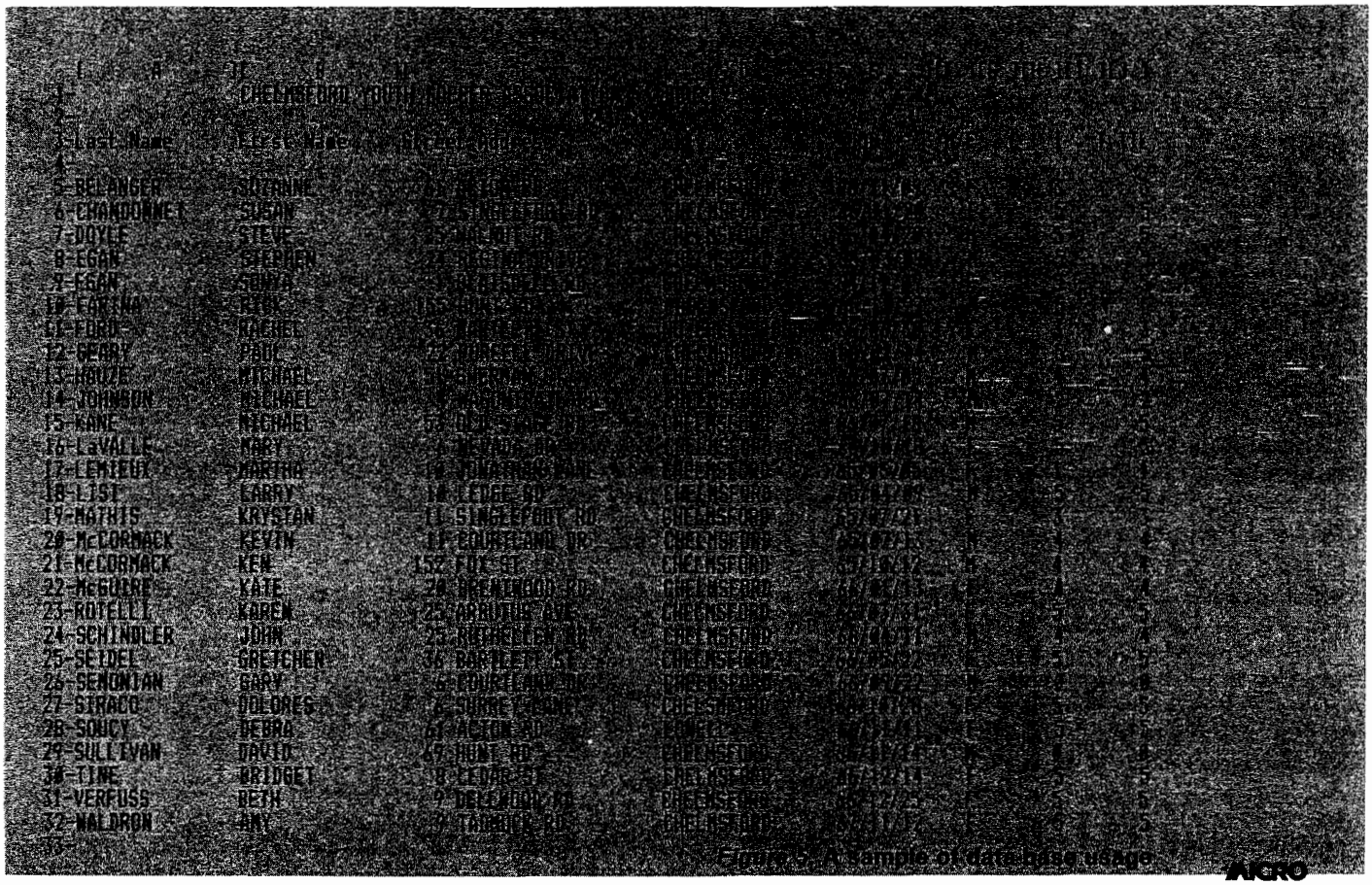


Figure 5. **A sample of data base usage**

SANYO MONITOR SALE!!

9" Data Monitor

- 80 Columns x 24 lines
- Green text display
- East to read - no eye strain
- Up front brightness control
- High resolution graphics
- Quick start - no preheating
- Regulated power supply
- Attractive metal cabinet
- UL and FCC approved

- *15 Day Free Trial - 90 Day Immediate Replacement Warranty*

9" Screen - Green Text Display	**$ 79.00**
12" Screen - Green Text Display (anti-reflective screen)	**$ 99.00**
12" Screen - Amber Text Display (anti-reflective screen)	**$119.00**
14" Screen - Color Monitor (national brand)	**$249.00**

Display Monitors From Sanyo

With the need for computing power growing every day, Sanyo has stepped in to meet the demand with a whole new line of low cost, high quality data monitors. Designed for commercial and personal computer use. All models come with an array of features, including up-front brightness and contrast controls. The capacity 5 x 7 dot characters as the input is 24 lines of characters with up to 80 characters per line.

Equally important, all are built with Sanyo's commitment to technological excellence. In the world of Audio/Video, Sanyo is synonymous with reliability and performance. And Sanyo quality is reflected in our reputation. Unlike some suppliers, Sanyo designs, manufactures and tests virtually all the parts that go into our products, from cameras to stereos. That's an assurance not everybody can give you!

SANYO

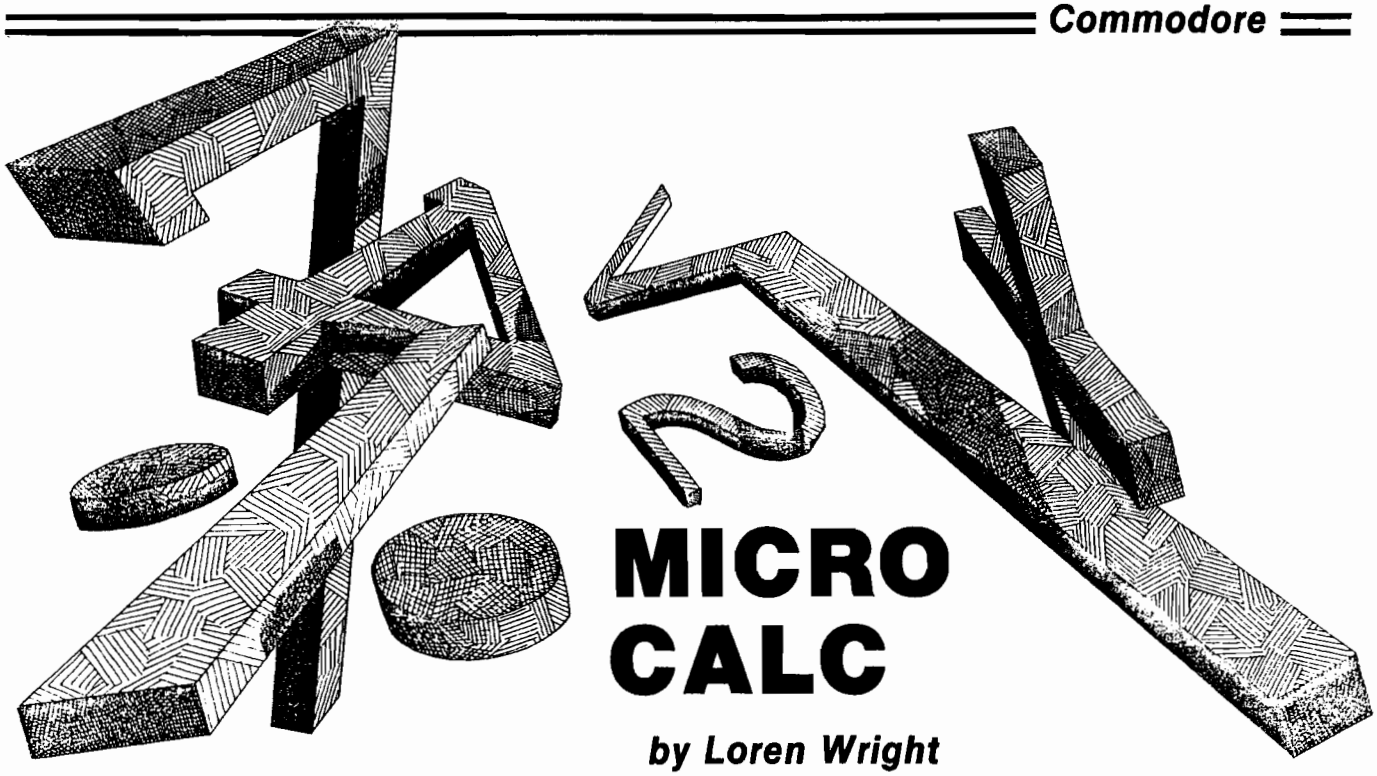
Official Video Products of the Los Angeles 1984 Olympics

• LOWEST PRICES • 15 DAY FREE TRIAL • 90 DAY FREE REPLACEMENT WARRANTY
• BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL • OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII orders. WE DO NOT EXPORT TO OTHER COUNTRIES.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Canada orders must be in U.S. dollars. Visa - MasterCard - C.O.D.

PROTECTO
ENTERPRIZES (WE LOVE OUR CUSTOMERS)
BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order



MICRO CALC

by Loren Wright

Typing in the Listing

For all Commodore computers you will be typing in all or part of listing 1, the version for the Commodore 64. If you are using a PET or a VIC-20, you should skip the lines marked. There are different versions of these lines in listing 2 (for the VIC-20) and listing 3 for the PET. The features for the PET and Commodore 64 versions are the same:

- ✓ 20 working lines
- ✓ 20 corresponding comment fields
- ✓ support of disk or tape files
- ✓ optional zeroing of user variables
- ✓ multiple statement support
- ✓ display of disk file name

The VIC-20 version has the following features

- ✓ 10 working lines
- ✓ support of tape files
- ✓ optional zeroing of user variables
- ✓ multiple statement support

Operating Instructions

Commodore 64

F7	performs calculation
F8	zeros user variables
left arrow	enters file mode
British pound	enters comment field

VIC-20

The VIC-20 version operates the same as the Commodore 64 version, except there is no comment field, so the British pound key has no function.

See the comments in the main article (page 11) for hints on how to get more onto the VIC's smaller screen. Predefined constants and functions will be particularly useful.

PET

Since the PET has no function keys, these have been replaced:

@	performs calculation
[	zeros user variables
left arrow	enters file mode
backslash	enters comment field

Using the Internal Timer

Commodore computers have a special variable TI, which increments once every 1/60 second. You can use this timer on a Micro Calc screen to compare the speed of BASIC functions. Following is a screen that demonstrates how to do this:

```
A=5.3507
T=TI
B=A↑2
U=TI-T
U?
T=TI
B=A*A
U=TI-T
U?
```

You may be surprised by the results of this comparison between using exponentiation and simple multiplication to square a number. Other comparisons you may wish to try are:

using a number *vs.* a variable in a calculation

the SQR( ) function *vs.* raising to the .5 power

SIN( ) *vs.* COS( )

How to Use the RND( ) Function

The RND( ) function on Commodore computers is actually a *pseudo*-random number generator. This is because each successive random number depends to some extent on the previous number. On VIC, C-64, and later PET models, the random number generator works as follows:

A *negative* argument reseeds the random number generator with a number calculated from the argument. If you use the same argument each time, you will generate the same sequence of random numbers. Use a negative argument only once to start a sequence. Then follow with positive arguments.

A *positive* argument will generate a new number in the sequence, without reseeding the generator.

A *zero* argument yields a random number that is not based on the seed.

To get the most closely random sequence, you should either use RND(0), or start by performing RND(-TI) and then follow with RND( ) using a positive argument.

To get random integers the following calculation should be used:

```
R=10
N=INT(R*RND(0)+1)
```

This gives random numbers N from 1 to R. If you leave the +1 out, you'll get numbers in the range 0 to R-1.

Comments on Commodore listings

Starting this month, our Commodore listings are being output on the EPSON FX-80 printer. This printer allows redefining some or all of the Epson ROM character set. After much testing, we arrived at a compromise set of characters. Since many of the reversed characters would be difficult to read at the size of these listings, we thought that it would be clearer for the reader typing these programs into his computer to underline the reversed characters. The Commodore programs that follow utilize this new style of listing. If anyone has any comments, pro or con, drop us a line with your viewpoint.

Listing 1 Commodore 64

```
 10 PRINT"■":POKE53281,0:POKE53280,0:
    GOSUB8000
 20 Q$=CHR$(34):CR$=CHR$(13):
    DL$=CHR$(20):RB$="R_■"                    A
 25 BL$="                              ":
    DI$="■♦■■":CC$="●■"
 30 NL=20:DIMC$(NL),S$(NL),S(NL)
 40 LL=1:GOSUB8490:PRINT"SQ";:GOTO110
100 LL=1:GOSUB8500:PRINT"SQ";:
    GOSUB9000
110 S$=S$(LL):
    IFRIGHT$(S$,1)
    ="?"THENPRINT"R"BL$CR$"■"TAB(10);
115 PRINTTAB(10)"■"S$DI$;
120 GOSUB2000
130 IFT$="■"THEN300
135 IFT$="■"THENGOSUB9000:LL=1:
    PRINT"SQ";:GOTO110
140 IFT$="■"THEN100
150 IFT$=CR$ORT$="Q"THEN210
160 IFT$="■"THEN270                           B
170 IFT$=DL$THEN240
180 IFT$="_"THENS$(LL)=S$:GOSUB5000:
    GOSUB9000:LL=1:GOTO110
185 IFT$="ε"THENPRINT"R_■":S$(LL)=S$:
    GOTO1000
200 IFLEN(S$)<27THENS$=S$+T$:
    PRINTT$DI$;:GOTO120
210 S$(LL)=S$
220 LL=LL+1:IFLL=NL+1THENLL=1:
    PRINTRB$;:PRINT"SQ"TAB(10);:
    GOTO110
230 PRINTRB$CR$TAB(10);:GOTO110
240 IFS$=""THEN120
250 PRINTRB$"■■■"DI$;
260 S$=LEFT$(S$,LEN(S$)-1):GOTO120
270 S$(LL)=S$:LL=LL-1
275 IFLL=0THENLL=NL:
    PRINTRB$"S]]]]]]]]]]
    QQQQQQQQQQQQQQQQQQQQ";:GOTO110
280 PRINTRB$CR$"■■"TAB(10) ;:GOTO110
300 PRINTRB$:S$(LL)=S$:GOSUB3000:
    GOSUB7000:PRINT"SQ"TAB(10);:LL=1:
    GOTO110
1000 PRINT"SQ■";:LL=1
1010 C$=C$(LL):PRINTC$CC$;
1020 GOSUB4500
1030 IFT$="ε"THENPRINT"_SQ";:
     C$(LL)=C$:LL=1:GOTO110
1040 IFT$=CR$ORT$="Q"THEN1100
1050 IFT$="■"THEN1200
1060 IFT$=DL$THEN1300
1065 IFASC(T$)<32ORASC(T$)
     >127THEN1020                             C
1080 IFLEN(C$)<9THENC$=C$+T$:
     PRINTT$CC$;:GOTO1020
1100 C$(LL)=C$
1110 LL=LL+1:IFLL=NL+1THENLL=1:
     PRINT"_■SQ";:GOTO1010
1120 PRINT"_■"CR$;:GOTO1010
1200 C$(LL)=C$
1210 LL=LL-1:IFLL=0THENLL=NL:PRINT
     "_SQQQQQQQQQQQQQQQQQQQQ";:
     GOTO1010
1220 PRINT"_■"CR$"■■";:GOTO1010
1300 IFC$=""THEN1020
1310 PRINT"_■■■"CC$;
1320 C$=LEFT$(C$,LEN(C$)-1):GOTO1020
2000 GOSUB4500
2010 IFT$="■"ORT$=CR$ORT$="Q"ORT$="■"O
     RT$=" "OR
     T$=DL$ORT$="■"ORT$="ε"THENRETURN
2015 IFT$="■"THENRETURN
2020 IFT$>","ANDT$<":"THEN2070
2030 IFT$>":"ANDT$<"["THEN2070                D
2040 IFT$>"'"ANDT$<","THEN2070
2050 IFT$="^"THEN2070
2060 GOTO2000
2070 RETURN
3000 PRINT"S■CALCULATING"
3005 FORJJ=1TONL:
     IFRIGHT$(S$(JJ),1)                       E
     ="?"THENGOSUB6500:GOTO3030
3010 IFLEN(S$(JJ))<3THEN3030
3020 A$=S$(JJ):GOSUB4000
3030 NEXT:PRINT"S              ■":RETURN
4000 II=0:KK=II
4010 II=II+1:KK=KK+1:                         F
     IFII>LEN(A$)THENGOSUB4100:RETURN
4020 XX=ASC(MID$(A$,II,1)):
     IFXX=59THENGOSUB4100:GOTO4010
4030 IFXX=33THENGOSUB4100:RETURN
4040 POKE511+KK,XX:GOTO4010
4100 POKE511+KK,0:KK=0:SYS828:RETURN
4500 GETT$:IFT$=""THEN4500
4510 RETURN                                   G
5000 PRINT"■RL■OAD OR RS■AVE"
5010 GOSUB4500
5020 IFT$="L"THENSA=0:FD$=",S,R":             H
```

```
     GOTO5045
5030 IFT$="S"THENSA=1:FD$=",S,W":
     GOTO5045
5040 GOTO5010
5045 PRINT"QRD■ISK OR RT■APE":
     GOSUB4500
5046 IFNOT((T$="D")OR(T$="T"))
     THEN5045
5048 INPUT"QQNAME";NA$
5050 IFT$="D"THENSA=SA+8:DV=8:NA$="@0:
     "+NA$+FD$:OPEN15,8,15:GOTO5060
5055 DV=1:NA$=""
5060 OPEN1,DV,SA,NA$:
     IFSAAND1THENGOSUB5090:GOSUB5200:
     GOTO5080
5070 GOSUB5110:GOSUB5300
5080 CLOSE1:CLOSE15:GOSUB8000:
     GOSUB8510:PRINT"SQ";:RETURN
5090 A$="":FORII=1TONL:S$=S$(II):
     IFS$=""THENS$="␣"
5100 A$=A$+S$+CR$:NEXT:PRINT#1,A$:
     DE=0:GOSUB5900:RETURN
5110 FORII=1TONL:INPUT#1,A$:DE=0:
     GOSUB5900:IFDETHENII=NL:NEXT:
     RETURN
5115 IFA$="␣"THENA$=""
5120 S$(II)=A$:NEXT:RETURN
5200 IFDETHENRETURN
5205 A$="":FORII=1TONL:S$=C$(II):
     IFS$=""THENS$="␣"
5210 A$=A$+S$+CR$:NEXT:PRINT#1,A$:
     DE=0:GOSUB5900:RETURN
5300 IFDETHENRETURN
5310 FORII=1TONL:DE=0:INPUT#1,A$:
     GOSUB5900:IFDETHENII=NL:NEXT:
     GOTO5340
5320 IFA$="␣"THENA$=""
5330 C$(II)=A$:NEXT
5340 RETURN
5900 IFDV=1THENRETURN
5910 INPUT#15,D1$,D2$,D3$,D4$:
     IFVAL(D1$)=0THENRETURN
5920 PRINT"■"D1$" "D2$" "D3$" "D4$
5930 FORJJ=1TO2000:NEXT
5940 DE=-1:RETURN
6500 BB=ASC(LEFT$(S$(JJ),1))-64:
     IFBB>13THENBB=BB-13:GOTO6530
6510 ONBBGOSUB6560,6570,6580,6590,
     6600,6610,6620,6630,6640,6650,
     6660,6670,6680
6520 GOTO6540
6530 ONBBGOSUB6690,6700,6710,6720,
     6730,6740,6750,6760,6770,6780,
     6790,6800,6810
6540 S(JJ)=XX
6550 RETURN
6560 XX=A:RETURN
6570 XX=B:RETURN
6580 XX=C:RETURN
6590 XX=D:RETURN
6600 XX=E:RETURN
6610 XX=F:RETURN
```

I

```
6620 XX=G:RETURN
6630 XX=H:RETURN
6640 XX=I:RETURN
6650 XX=J:RETURN
6660 XX=K:RETURN
6670 XX=L:RETURN
6680 XX=M:RETURN
6690 XX=N:RETURN
6700 XX=O:RETURN
6710 XX=P:RETURN
6720 XX=Q:RETURN
6730 XX=R:RETURN
6740 XX=S:RETURN
6750 XX=T:RETURN
6760 XX=U:RETURN
6770 XX=V:RETURN
6780 XX=W:RETURN
6790 XX=X:RETURN
6800 XX=Y:RETURN
6810 XX=Z:RETURN
7000 PRINT"SQ";:FORII=1TONL:S$=S$(II):
     SS=S(II)
7010 X$="":
     IFRIGHT$(S$,1)="?"THENX$=STR$(SS)
     +"R"+LEFT$(BL$,24-LEN(STR$(SS)))
7020 PRINTTAB(10)S$X$:NEXT:RETURN
8000 RESTORE:FORII=0TO42:READAA:
     POKE828+II,AA:NEXT:RETURN
8490 FORII=1TONL:READS$(II):S(II)=0:
     NEXT
8495 FORII=1TONL:READC$(II):NEXT:
     GOTO8510
8500 FORII=1TONL:C$(II)="":S$(II)="":
     S(II)=0:NEXT
8510 PRINT"■Q";:FORII=1TONL:S$=S$(II):
     C$=C$(II)
8520 PRINT"■"C$LEFT$(BL$,10-LEN(C$))
     "■"S$"R"LEFT$(BL$,28-LEN(S$))
8530 NEXT:
     PRINT"Q■"MID$(NA$,4)"■■■■■■    ":
     RETURN
9000 PRINT"S■CLEAR"
9010 A=0:B=A:C=A:D=A:E=A:F=A:G=A:H=A:
     I=A:J=A:K=A:L=A:M=A
9020 N=A:O=A:P=A:Q=A:R=A:S=A:T=A:U=A:
     V=A:W=A:X=A:Y=A:Z=A
9030 PRINT"S     ■S":RETURN
9828 DATA165,122,141,112,3,165,123,
     141,113,3,169,0,133,122,169,2,
     133,123,32,121
9848 DATA165,169,0,133,122,169,2,133,
     123,32,165,169,173,112,3,133,122,
     173,113,3
9868 DATA133,123,96
9900 DATAA=8000,M=48,I=11.9,I=I/1200,
     D=(1-(1+I)^-M)/I
9910 DATAP=A/D,P=INT(P*100+.5)
     /100,P?,,
9915 DATA,,,,,,,,,
9920 DATAPRINCIPAL,MONTHS,INTEREST,,
     DIVISOR,,,PAYMENT,,
9925 DATA,,,,,,,,,
```

J

K

L

M

Comments on VIC and Pet listings

The C-64 listing is the complete MICROCalc listing. If you have a VIC, Expanded VIC or PET then the listings are not complete. For VIC and PET, you must use the C-64 listing from lines 4000-7999, an Expanded VIC has additional changes to the standard VIC program.

Listing 2 VIC-20

```
 10 PRINT"■":POKE36879,8:GOSUB8000
 20 CR$=CHR$(13):DL$=CHR$(20):
    RB$="R ■":
    BL$="                    ":
    DI$="♦■■"
 30 NL=10:DIMS$(NL),S(NL)
 40 LL=1:GOSUB8490:GOTO110                    A
100 LL=1:GOSUB8500:GOSUB9000
110 S$=S$(LL):
    IFRIGHT$(S$,1)
    ="?"THENPRINT"R"BL$CR$"■";
115 PRINTS$DI$;
120 GOSUB2000
125 IFT$="■"THENPRINTRB$:GOSUB9000:
    LL=1:PRINT"SQ";:GOTO110
130 IFT$="■"THENPRINTRB$:S$(LL)=S$:
    GOSUB3000:GOSUB7000:PRINT"SQ";:
    LL=1:GOTO110
140 IFT$="■"THEN100
150 IFT$=CR$ORT$="Q"THEN210                  B
160 IFT$="■"THEN270
170 IFT$=DL$THEN240
180 IFT$="_"THENS$(LL)=S$:GOSUB5000:
    GOSUB9000:LL=1:GOTO110
190 S$=S$+T$
200 IFLEN(S$)<19THEN120
210 S$(LL)=S$
220 LL=LL+1:IFLL=NL+1THENLL=1:
    PRINTRB$"SQ";:GOTO110
230 PRINTRB$CR$CR$;:GOTO110
240 IFS$=""THEN120
250 PRINTRB$"■■"DI$;
260 S$=LEFT$(S$,LEN(S$)-1):GOTO120
270 S$(LL)=S$:LL=LL-1:IFLL=0THENLL=NL:
    PRINTRB$"SQQQQQQQQQQQQQQQQQQQ";:
    GOTO110
280 PRINTRB$CR$"■■■";:GOTO110
2000 GOSUB4500
2005 IFT$="■"THENRETURN
2010 IFT$="■"ORT$=CR$ORT$="Q"ORT$="■"O
     RT$="_"ORT$=DL$ORT$="■"THENRETURN
2020 IFT$>"","ANDT$<":"THEN2070
2030 IFT$>":"ANDT$<"["THEN2070               D
2040 IFT$>"'"ANDT$<","THEN2070
2050 IFT$="^"THEN2070
2060 GOTO2000
2070 PRINTT$DI$;:RETURN
3000 FORJJ=1TONL:
     IFRIGHT$(S$(JJ),1)
     ="?"THENGOSUB6500:GOTO3030
3010 IFLEN(S$(JJ))<3THEN3030
3020 A$=S$(JJ):GOSUB4000                     E
3030 NEXT:RETURN
```

```
5100 XX=FRE(0):A$=A$+S$+CR$:NEXT:
     PRINT#1,A$:RETURN
```

```
8000 RESTORE:FORII=0TO42:READAA:              K
     POKE828+II,AA:NEXT:RETURN
8490 FORII=1TONL:READS$(II):S(II)=0:
     NEXT:GOTO8510
8500 FORII=1TONL:S$(II)="":S(II)=0:           L
     NEXT
8510 PRINT"■";:FORII=1TONL:S$=S$(II)
8520 PRINT"Q■"S$"R"LEFT$(BL$,
     20-LEN(S$)):NEXT:PRINT"SQ";:
     RETURN
9000 PRINT"S■CLEAR■":A=0:B=A:C=A:D=A:
     E=A:F=A:G=A:H=A:I=A:J=A:K=A:L=A:
     M=A
9010 N=A:O=A:P=A:Q=A:R=A:S=A:T=A:U=A:
     V=A:W=A:X=A:Y=A:Z=A:
     PRINT"S      ":RETURN                 M
9828 DATA165,122,141,112,3,165,123,
     141,113,3,169,0,133,122,169,2,
     133,123,32,121
9848 DATA197,169,0,133,122,169,2,133,
     123,32,165,201,173,112,3,133,122,
     173,113,3
9868 DATA133,123,96
9900 DATAA=8000,M=48,I=11.9,I=I/1200,
     D=(1-(1+I)^-M)/I
9910 DATAP=A/D,P=INT(P*100+.5)
     /100,P?,,
```

Listing 3 Expanded VIC-20

Change These Lines to get the Improved Expanded VIC Version

```
130 IFT$="■"THENPRINTRB$:S$(LL)=S$:
    GOSUB3000:GOSUB7000:PRINT"SQ";:
    LL=1:GOTO110
135 IFT$="■"THENGOSUB9000:PRINT"SQ";:
    LL=1:GOTO110
180 IFT$="_"THENS$(LL)=S$:GOSUB5000:
    LL=1:GOTO110
230 PRINTRB$CR$;:GOTO110
280 PRINTRB$CR$"■■■";:GOTO110
2005 IFT$="@"THEN2005
2015 IFT$="■"THENRETURN
3000 PRINT"S■CALCULATING":FORJJ=1TONL:
     IFRIGHT$(S$(JJ),1)="?"THEN
     GOSUB6500:GOTO3030
3030 NEXT:PRINT"S          @":RETURN
8510 PRINT"■Q";:FORII=1TONL:S$=S$(II)
8520 PRINT"■"S$"R"LEFT$(BL$,
     20 -LEN(S$)):NEXT
8530 PRINT"Q■"MID$(NA$,4)"■■■
```

Listing 4 PET

```
 10 PRINT"■":GOSUB8000
 20 Q$=CHR$(34):CR$=CHR$(13):
    DL$=CHR$(20):RB$="R_■"                    A
 25 BL$="_____________________________":
    DI$="♦■■":CC$="●■"
 30 NL=20:DIMC$(NL),S$(NL),S(NL)
 35 GOTO100
 40 LL=1:GOSUB8490:PRINT"SQ";:GOTO110
100 LL=1:GOSUB8500:PRINT"SQ";:
    GOSUB9000
110 S$=S$(LL):
    IFRIGHT$(S$,1)
    ="?"THENPRINT"R"BL$CR$"■"TAB(10);
115 PRINTTAB(10)S$DI$;
120 GOSUB2000
130 IFT$="@"THEN300                           B
135 IFT$="["THENGOSUB9000:LL=1:
    PRINT"SQ";:GOTO110
140 IFT$="■"THEN100
150 IFT$=CR$ORT$="Q"THEN210
160 IFT$="■"THEN270
170 IFT$=DL$THEN240
180 IFT$="_"THENS$(LL)=S$:GOSUB5000:
    GOSUB9000:LL=1:GOTO110
185 IFT$="ε"THENPRINT"R_■":S$(LL)=S$:
    GOTO1000
200 IFLEN(S$)<27THENS$=S$+T$:
    PRINTT$DI$;:GOTO120
210 S$(LL)=S$
220 LL=LL+1:IFLL=NL+1THENLL=1:
    PRINTRB$;:PRINT"SQ"TAB(10);:
    GOTO110
230 PRINTRB$CR$TAB(10);:GOTO110
240 IFS$=""THEN120
250 PRINTRB$"■■■"DI$;
260 S$=LEFT$(S$,LEN(S$)-1):GOTO120
270 S$(LL)=S$:LL=LL-1
275 PRINTRB$"S]]]]]]]]]QQQQQQQQQQ
    QQQQQQQQQQ";:GOTO110
280 PRINTRB$CR$"■"TAB(10);:GOTO110
300 PRINTRB$:S$(LL)=S$:GOSUB3000:
    GOSUB7000:PRINT"SQ"TAB(10);:LL=1:
    GOTO110
1000 PRINT"SQ■";:LL=1
1010 C$=C$(LL):PRINTC$CC$;
1020 GOSUB4500
1030 IFT$="ε"THENPRINT"_SQ";:
     C$(LL)=C$:LL=1:GOTO110
1040 IFT$=CR$ORT$="Q"THEN1100
1050 IFT$="■"THEN1200
1060 IFT$=DL$THEN1300                         C
1065 IFASC(T$)<32ORASC(T$)
     >127THEN1020
1080 IFLEN(C$)<9THENC$=C$+T$:
     PRINTT$CC$;:GOTO1020
1100 C$(LL)=C$
1110 LL=LL+1:IFLL=NL+1THENLL=1:
     PRINT"_■SQ";:GOTO1010
1120 PRINT"_■"CR$;:GOTO1010
1200 C$(LL)=C$
1210 LL=LL-1:IFLL=0THENLL=NL:
     PRINT"_SQQQQQQQQQQQQQQQQQQQ";:
     GOTO1010
1220 PRINT"_■"CR$"■";:GOTO1010
1300 IFC$=""THEN1020
1310 PRINT"_■■■"CC$;
1320 C$=LEFT$(C$,LEN(C$)-1):GOTO1020
2000 GOSUB4500
2010 IFT$="@"ORT$=CR$ORT$="Q"ORT$="■"OR
     T$="_"ORT$=DL$ORT$="■"ORT$="ε"THE
     NRETURN
2015 IFT$="["THENRETURN
2020 IFT$>","ANDT$<":"THEN2070                D
2030 IFT$>":"ANDT$<"["THEN2070
2040 IFT$>"'"ANDT$<","THEN2070
2050 IFT$="^"THEN2070
2060 GOTO2000
2070 RETURN
3000 PRINT"SRCALCULATING"
3005 FORJJ=1TONL:
     IFRIGHT$(S$(JJ),1)
     ="?"THENGOSUB6500:GOTO3030
3010 IFLEN(S$(JJ))<3THEN3030                  E
3020 A$=S$(JJ):GOSUB4000
3030 NEXT:PRINT"S              ":RETURN
```

```
8000 RESTORE:FORII=0TO42:READAA:              K
     POKE828+II,AA:NEXT:RETURN
8490 FORII=1TONL:READS$(II):S(II)=0:
     NEXT
8495 FORII=1TONL:READC$(II):NEXT:             L
     GOTO8510
8500 FORII=1TONL:C$(II)="":S$(II)="":
     S(II)=0:NEXT
8510 PRINT"■Q";:FORII=1TONL:S$=S$(II):
     C$=C$(II)
8520 PRINTC$LEFT$(BL$,10-LEN(C$))
     S$"R"LEFT$(BL$,28-LEN(S$))               M
8530 NEXT:
     PRINT"Q"MID$(NA$,4)"■■■■■■■     ":
     RETURN
9000 PRINT"SRCLEAR"
9010 A=0:B=A:C=A:D=A:E=A:F=A:G=A:H=A:
     I=A:J=A:K=A:L=A:M=A
9020 N=A:O=A:P=A:Q=A:R=A:S=A:T=A:U=A:
     V=A:W=A:X=A:Y=A:Z=A
9030 PRINT"S      S":RETURN
9828 DATA165,119,141,112,3,165,120,
     141,113,3,169,0,133,119,169,2,
     133,120,32,251
9848 DATA180,169,0,133,119,169,2,133,
     120,32,48,185,173,112,3,133,119,
     173,113,3
9868 DATA133,120,96
9900 DATAA=8000,M=48,I=11.9,I=I/1200,
     D=(1-(1+I)^-M)/I
9910 DATAP=A/D,P=INT(P*100+.5)
     /100,P?,,
9915 DATA,,,,,,,,,
9920 DATAPRINCIPAL,MONTHS,INTEREST,,
     DIVISOR,,,PAYMENT,,
9925 DATA,,,,,,,,,
```

MICRO

PUT PRICES IN CHECK

CARTRIDGE RIBBONS FOR

APPLE PRINTERS

NEC 8023A

C. ITOH PROWRITER

$9.95 EA. $107.46 DOZ.

CARTRIDGE RIBBONS FOR

EPSON

MX-80 MX-100

$6.99 EA $11.95 EA

$86.29 EA $129.06 DOZ

MAXELL DISKETTES

5¼" SINGLE SIDE
DUAL DENSITY
MD-1

$29.90 10 PACK

INNOVATIVE CONCEPTS

FLIP'N'FILE

DISC STORAGE BOX
HOLDS UP TO 60 DISKETTES

5¼" 8"

$24.95 EA $29.95 EA

RIBBONS FOR

IDS PRINTERS

	EA	DOZ
440	$2.77	$29.92
PAPER TIGER	$6.95	$75.06
MICROPRISM	$7.99	$86.29
PRISM	10.95	118.25

DISKETTE STORAGE BOXES

5¼" — BLUE OR BEIGE

$2.49 EA.

DUAL SPOOL RIBBONS FOR

OKIDATA PRINTERS

	EA	DOZ
80, 82, 83 92, 93	$2.77	$29.92
84	$5.99	$64.69

MEMOREX DISKETTES

5¼ SINGLE SIDE - DUAL DENSITY

$24.99 10 PACK

COLOR-CODER

LIBRARY CASE SET

CONTAINS 5 BRIGHT COLORS

5¼" 8"

$19.95 SET OF 5 $23.95 SET OF 5

ANTI-STATIC SPRAY

FULL QUART SIZE
WITH DISPENSER

$6.95 QT.

1 GALLON REFILL $19.95

CARTRIDGE RIBBONS FOR

COMREX

DAISYWRITER 2000

$2.49 EA $26.89 DOZ

LABEL SPECIAL

$2.99 / K

(5K MIN)

1 ACROSS 3 x 15/16 CONTINUOUS LABELS

MOST RIBBONS AVAILABLE IN COLORS TOO!

CALL OR WRITE FOR OUR SUPPLIES CATALOGUE

ON ORDERS UNDER $14.00 PLEASE ADD $3.00 FOR SHIPPING

MINIMUM RIBBON ORDER $30.00 OR 1 DOZEN

MasterCard VISA

Check-Mate™

TOLL FREE 800-343-7706
IN MASS 617-963-7694
PHONES OPEN 9AM-7PM EASTERN TIME

51 DIAUTO DR. P.O. BOX 103
RANDOLPH, MA 02368

MASS RESIDENTS
ADD 5% SALES TAX

C-64 CIA ALARM CLOCK

8:30:58.9

By Ian Adam

This article shows how to use the extremely accurate time-of-day clock built into the Commodore 64's CIA chip. The demonstration includes an alarm clock that runs independently of most other programs and I/O function.

The Commodore 64 does an acceptable job of keeping time with its TI$ clock. All you have to do is set TI$ to the current time, and it will run as long as the computer remains on. You can use the C-64 for just about anything else and just type ?TI$ when you want the time.

However, there are several limitations on use of the built-in time function. First of all, it is not very convenient to use while a program is running. You have to STOP the program, carefully ask the time (no syntax error, or else...), then CONTinue with the program. Second, the TI$ function is not very accurate. A recent sample of a half dozen computers gave a typical error of 2.2%. That's over one minute per hour, or 32 minutes per day! Good enough to keep track of whether it's light or dark outside, perhaps, but not adequate to launch the space shuttle. The TI$ clock also stops, running when a file or program is being loaded or saved. And third, of course, it's only there when you ask, and we all know how easy that is to forget!

But despair not; a ready solution is at hand. In fact, your 64 contains not one, but three clocks (count 'em!). The additional timepieces are contained in the two 6526 Complex Interface Adapter chips (CIA's), and they offer some considerable advantages. These two chips are provided by Commodore for the purpose of carrying out a multitude of housekeeping functions, such as generating interrupts, reading the keyboard and joysticks, external communications, etc. The two clocks are a bonus. And very accurate, too ... they include tenths of seconds, and appear to be that accurate over the course of a day. They also keep running during input/output operations, in fact anytime the computer is turned on. With two clocks, you could keep track of the time in Mandalay, if you want, and there are even programmable alarms so you won't miss dinner there either!

Using the Program

Type in the listing as shown; type the DATA statements carefully, since they contain the machine language program. Any error in that portion can crash the computer.

When you've finished typing, SAVE a copy of the program before you run it. This will avoid having to retype the whole thing in the event of a fatal typographical error. Then go ahead and RUN it. First, the program will READ the machine language DATA and store it in memory. The variable CH is a checksum to guard against errors in the data. If the program stops and indicates a data error, then double-check everything. Assuming that is ok, the program will then ask you to enter the correct time. Give the AM/PM and the hour; when asked for the minute, check an accurate time source, and type in the number of the next minute (e.g. if it's 8:30, type in 31), then wait until that minute arrives to press "RETURN". Pressing return starts the clock.

After the instructions, you will be prompted for the time you want the alarm to be set to. When the alarm time matches the clock time, it will be announced by:

-the border of the screen flashing;
-a buzzing sound; and,
-the word "ALARM" flashing above the time.

If that isn't enough to attract your attention, then perhaps you're in a time warp! In any event, you can turn off this display simple by pressing the "F1" key.

When properly loaded, the program will run just like clockwork. It will supply you with the instructions, but for your reference I'll repeat them here:

	(equals) 0 to 15;
SYS 832:	recall time display to screen;
SYS 994:	turn off time display (still runs internally);
POKE 982,n:	change color, where n (equals) 0 to 15;
GOSUB 9140:	set or reset alarm time;
F1:	stop alarm display.

As before, the operating part of the program is in subroutine form, so you can include it in other programs as you wish. The commands listed above can also be used within a program, once the time is correctly set. To stop the alarm display under program control, just use POKE 197,4.

If the F1 key is not convenient for stopping the alarm, for example if your program uses it for some other purpose, then it can be changed: it's the value that appears in memory location 197

when the key is pressed.

The main program is stored in the cassette buffer, while the alarm portion occupies an unused area of page 2 memory as well. Thus, they may conflict with other programs that use these areas for machine language or sprites. The program would also be destroyed by any tape load or save activities. If these cautions create a problem, then the routine should be relocated to a different area of memory. This is a straightforward task, but does require a basic knowledge of machine language. For those who want to study the machine code, an assembly listing is provided (listing 2).

Technical Details

The CIA clocks are a little more difficult to access than TI$; with this program, however, it's as easy as apple pie. The time is contained in four registers for hours, minutes, seconds, and tenths (locations 56331 to 56328 for chip A, and 56587 to 56584 for chip B). To avoid having the time flip over while reading the clock, all four registers are frozen whenever the hours register is read or written to. The clocks continue to keep good time while this is in progress. The last step in any access should be a read or write to the tenths-of-seconds register, to "unfreeze" the clock.

Each register stores its data in binary-coded decimal, or BCD, format. The hours register also contains an AM/PM indicator in bit 7. This would be an ideal format if the chip were running a display such as a digital alarm clock directly. For use in the computer, however, it does require some coding and decoding. This is a little awkward but not too difficult in BASIC, and even simpler in machine language. Once set, the clocks will automatically keep track of the time and AM/PM for as long as the machine is on, or until they are reset.

So how do we set the alarm? By exactly the same process as setting the time... by poking values into the same four registers! Only difference is, we first have to set bit 7 of the control register to a one. This signals to the chip that we want to set the alarm time instead of the clock time. The multiple use of these registers does keep things simple-honest! Lines 9160 and 9170 of the program take care of the BCD conversion. Line 9175 sets the control register to its normal value.

Once the alarm has been set, it cannot be read. When the prescribed time is reached, this is signalled by setting bit 2 of the interrupt register. The program must recognize this, and proceed to alarm the operator in whatever way is specified. Don't worry, it won't bite. All of this may sound complicated, but the program takes care of the details.

These are the main registers involved:

Function	Chip A	Chip B
Hours & AM/PM	56331 ($DC0B)	56587 ($DD0B)
Minutes	56330 ($DC0A)	56586 ($DD0A)
Seconds	56329 ($DC09)	56585 ($DD09)
Tenths	56328 ($DC08)	56584 ($DD08)
Interrupt	56333 ($DC0D)	56589 ($DD0D)
Alarm Control	56336 ($DC0F)	56591 ($DD0F)

A Quirk In The Chip

Help; there's a quirk on the loose!

In programming the alarm, I came upon a most unusual feature. I got the alarm all set up, and (after a lot of hit-and-miss changes) finally got it to work well. Eureka. The only trouble was, the alarm would mysteriously sound for a second time, exactly one minute later. Really had me stumped for a while. After a sleepless night, however, it came to me what the problem was. Say the alarm is set for 8:30:00.0, and sounds at that time. One minute later, at precisely 8:30:59.9, the tenths-of-a-second register rolls over, giving an instantaneous time reading of 8:31:00.0, but it is apparently sufficient to trigger a second alarm (sometimes).

There are three or four ways to program around this quirk, once you know it exists. I decided the easiest way to solve the problem would be to POKE a 1 into the tenths register (i.e. setting the alarm time in our example to 8:30:00.1). This removes the alarm from the vicinity of the rollover, and seems to have banished the quirk to another world. If it should ever return to haunt you, please let me know; maybe we'll try an exorcist. MICRO

You may contact Ian Adam at 3706 West 20th Ave., Vancouver BC, V6S1E8 Canada.

Listing 1

```
  0 REM ** TIME IS OF THE ESSENCE   **
  1 REM
  2 REM ** CLOCK DISPLAY WITH ALARM **
  3 REM     FOR COMMODORE 64
  4 REM
  5 REM     BY IAN ADAM
  6 REM     VANCOUVER, B. C.
  7 REM
 10 GOSUB9000
 20 PRINT"□QQ** TIME WAITS FOR NO MAN
    **Q
 30 PRINT"CLOCK INSTRUCTIONS:Q"
 40 PRINT"SYS 832:     TURN ON DISPLAY
 50 PRINT"SYS 994:     TURN OFF DISPLAY
 60 PRINT"POKE 982,N: CHANGE COLOUR
 70 PRINT"GOSUB 9140: RESET ALARM
 80 PRINT"F1:         TURN OFF ALARM
 90 GOSUB9140:END:
    REM SET OR RESET ALARM
8990 REM BALANCE OF PROGRAM IS
    SUBROUTINES THAT CAN BE USED
    INDEPENDENTLY
9000 CH=0:FORI=832TO1008
9010 READA:POKEI,A:CH=CH+A:NEXT
```

(continued)

Listing 1 *(continued)*

```
9015 FORI=679TO744:READA:POKEI,A:
     CH=CH+A:NEXT
9020 IFCH-23614THENPRINT"QWHOA...DATA
     ERROR":STOP:NOTE CHECKSUM
9030 INPUT"QQQQQQ IS IT AM OR PM";A$:
     INPUT"Q AND THE HOUR";H
9040 PRINT"QQ ENTER THE MINUTE WHEN
     YOU WISH TO START
9050 PRINT" PRESS 'RETURN' TO START
     THE CLOCK:Q
9060 IFH>12THENA$="P":H=H-12:GOTO9060
9070 IFH>9THENH=H+6  :
     REM CONVERSION TO BCD
9080 IFLEFT$(A$,1)="P"THENH=H+128
9090 C=56328:POKEC+3,H:POKEC+1,0
9100 INPUTM:M=M+INT(M/10)*6
9110 POKEC+2,M:POKEC,0:SYS832:
     PRINT"QQ IF NOT OK,
     PRESS ANY KEY
9120 FORI=1TO1000:
     IFPEEK(198)THENPOKE198,0:SYS994:
     GOTO9030
9130 NEXT:RETURN
9140 PRINT"QWHAT TIME WOULD YOU LIKE
     THE ALARM?Q"
9145 INPUT"AM OR PM";A$:
     A$=LEFT$(A$,1)
9150 INPUT"THE HOUR";H
9155 IFH>12THENA$="P":H=H-12:GOTO9155
9160 H=H-6*(H>9)-128*(A$="P"):
     REM CONVERT TO BCD AND ADD AM/PM
     INDICATOR
9165 INPUT"THE MINUTE";M
9170 M=M+INT(M/10)*6
9175 C=56328:POKEC+7,136:POKEC+3,H:
     POKEC+2,M:POKEC,1:POKEC+7,8:
     REM ALARM
9180 POKE54273,99:POKE54278,240:

     POKE54276,21
9185 POKE54287,2:POKE54290,17:
     REM  SOUND
9190 RETURN
9200 DATA 120,173,20,3,162,89,234,234,
     234,142,20,3,173,21,3
9210 DATA 162,3,234,234,234,142,21,3,
     88,96,173,11,220,170,41
9220 DATA 15,24,105,48,141,67,4,138,
     16,4,162,16,16,2,162,1,142
9230 DATA 77,4,162,32,41,16,240,2,162,
     49,142,66,4,173,10,220
9240 DATA 170,41,15,105,48,141,70,4,
     138,74,74,74,74,24,105,48
9250 DATA 141,69,4,173,9,220,170,41,
     15,105,48,141,73,4,138,74
9260 DATA 74,74,74,24,105,48,141,72,4,
     173,8,220,105,48,141,75
9270 DATA 4,169,32,141,65,4,141,76,4,
     141,79,4,162,14,157,24
9280 DATA 4,202,208,250,169,58,141,68,
     4,141,71,4,169,46,141,74
9290 DATA 4,169,13,141,78,4,169,1,162,
     13,157,65,216,202,208,250,76,167,
     2
9300 DATA 120,169,49,234,141,20,3,169,
     234,234,141,21,3,88,96
9310 DATA 173,13,220,41,4,240,3,141,
     227,2,173,227,2,240,42,173,162,0
9320 DATA 106,106,106,41,12,141,32,
     208,41,4,141,24,212,240,11,162,5,
     189
9330 DATA 227,2,157,33,4,202,208,247,
     173,197,0,201,4,208,6,142,227,2
9340 DATA 142,24,212,76,49,234,0,1,12,
     1,18,13
```

Listing 2

```
                0010 ; CIA ALARM CLOCK
                0020 ;  BY IAN ADAM
                0030 ;
                0035 ALARM    .DE $02A7
                0040 CINV     .DE $0314
;HARDWARE INTERRUPT
                0050 MESSGE   .DE $0418
;'ALARM' DISPLAYED HERE
                0055 SCRMSG   .DE MESSGE+9
                0060 DISP     .DE $0441
;BEGINNING OF TIME DISPLAY
                0065 BORDER   .DE $D020
                0070 SIDVOL   .DE $D418
                0080 DSPCLR   .DE $D841
;COLOR MEMORY
                0090 ;CIA #1 REGISTERS
; FOR TIME DISPLAY
                0100 TENTHS   .DE $DC08
                0110 SECS     .DE TENTHS+1
                0115 MINS     .DE TENTHS+2
                0120 HOURS    .DE TENTHS+3
                0125 CIAINT   .DE TENTHS+5
;CIA INTERRUPT
                0130 ALCTRL   .DE TENTHS+7
                0135 INTPTR   .DE $EA31
;NORMAL CONTENTS
                0140 ;
                0150          .BA $0340
                0160 ;
0340-78         0170          SEI
0341-AD 14 03   0180          LDA CINV
0344-A2 59      0190          LDX #L,START
0346-EA         0200          NOP
0347-EA         0210          NOP
0348-EA         0220          NOP
0349-8E 14 03   0230          STX CINV
034C-AD 15 03   0240          LDA CINV+1
034F-A2 03      0250          LDX #H,START
0351-EA         0260          NOP
0352-EA         0270          NOP
0353-EA         0280          NOP
0354-8E 15 03   0290          STX CINV+1
0357-58         0300          CLI
0358-60         0310          RTS
                0320 ;
0359-AD 0B DC   0330 START    LDA HOURS
035C-AA         0340          TAX
035D-29 0F      0350          AND #$0F
035F-18         0360          CLC
0360-69 30      0370          ADC #$30
0362-8D 43 04   0380          STA DISP+2
0365-8A         0390          TXA
0366-10 04      0400          BPL LBLA
0368-A2 10      0410          LDX #$10
036A-10 02      0420          BPL LBLB
036C-A2 01      0430 LBLA     LDX #$01
036E-8E 4D 04   0440 LBLB     STX DISP+12
0371-A2 20      0450          LDX #$20
0373-29 10      0460          AND #$10
0375-F0 02      0470          BEQ LBLC
0377-A2 31      0480          LDX #$31
0379-8E 42 04   0490 LBLC     STX DISP+1
037C-AD 0A DC   0500          LDA MINS
037F-AA         0510          TAX
0380-29 0F      0520          AND #$0F
0382-69 30      0530          ADC #$30
0384-8D 46 04   0540          STA DISP+5
0387-8A         0550          TXA
0388-4A         0560          LSR A
0389-4A         0570          LSR A
038A-4A         0580          LSR A
038B-4A         0590          LSR A
038C-18         0600          CLC
038D-69 30      0610          ADC #$30
038F-8D 45 04   0620          STA DISP+4
0392-AD 09 DC   0630          LDA SECS
0395-AA         0640          TAX
0396-29 0F      0650          AND #$0F
0398-69 30      0660          ADC #$30
```

(Continued on next page)

Commodore 64
and
VIC-20

SuperTerm

$149.95

Telecommunications
with a difference!

Unexcelled communications power and compatibility, especially for professionals and serious computer users. Look us over; **SuperTerm** isn't just "another" terminal program. Like our famous Terminal-40, **it's the one others will be judged by.**

- **EMULATION**—Most popular terminal protocols: cursor addressing, clear, home, etc.
- **EDITING**—Full-screen editing of Receive Buffer
- **UP/DOWNLOAD FORMATS**—CBM, Xon-Xoff, ACK-NAK, CompuServe, etc.
- **FLEXIBILITY**—Select baud, duplex, parity, stopbits, etc. Even work off-line, then upload to system!
- **DISPLAY MODES**—40 column; 80/132 with side-scrolling
- **FUNCTION KEYS**—8 standard, 52 user-defined
- **BUFFERS**—Receive, Transmit, Program, and Screen
- **PRINTING**—Continuous printing with Smart ASCII interface and parallel printer; buffered printing otherwise
- **DISK SUPPORT**—Directory, Copy, Rename, Scratch

Program options are selected by menus and function keys. For maximum convenience, an EXEC file sets all options on start-up. SuperTerm may be backed-up for safety. Software on disk with special cartridge module.

Write for the full story on SuperTerm; or, if you already want that difference, order today!

Requires: Commodore 64 or VIC-20, disk drive or Datasette, and compatible modem. VIC version requires 16K memory expansion. Please specify VIC or 64 when ordering.

Smart ASCII Plus . . . $59.95

The only interface which supports streaming —sending characters simultaneously to the screen and printer — with SuperTerm.

Also great for use with your own programs or most application programs, i.e., word processors. **Print modes:** CBM Graphics, TRANSLATE, DaisyTRANSLATE, CBM/True ASCII, and PIPELINE.

Complete with printer cable and manual. On disk or cassette.

VIC 20 and Commodore 64 are trademarks of Commodore Electronics, Ltd.

(816) 333-7200

Send for a free brochure

MAIL ORDER: Add $1.50 shipping and handling ($3.50 for C.O.D.). Visa/Mastercard accepted (card # and exp. date). MO residents add 5.625% sales tax. Foreign orders payable U.S.$, U.S. Bank ONLY, add $2 shipping.

MIDWEST MICRO inc.

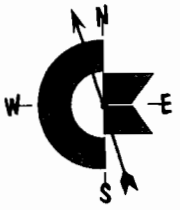
311 WEST 72nd ST. • KANSAS CITY • MO • 64114

Commodore

Listing 2 *(continued)*

```
039A-8D 49 04 0670            STA DISP+8
039D-8A       0680            TXA
039E-4A       0690            LSR A
039F-4A       0700            LSR A
03A0-4A       0710            LSR A
03A1-4A       0720            LSR A
03A2-18       0730            CLC
03A3-69 30    0740            ADC #$30
03A5-8D 48 04 0750            STA DISP+7
03A8-AD 08 DC 0760            LDA TENTHS
03AB-69 30    0770            ADC #$30
03AD-8D 4B 04 0780            STA DISP+10
03B0-A9 20    0790            LDA #$20
03B2-8D 41 04 0800            STA DISP
03B5-8D 4C 04 0810            STA DISP+11
03B8-8D 4F 04 0820            STA DISP+14
03BB-A2 0E    0830            LDX #$0E
03BD-9D 18 04 0840 LBLD       STA MESSGE,X
03C0-CA       0850            DEX
03C1-D0 FA    0860            BNE LBLD
03C3-A9 3A    0870            LDA #$3A
03C5-8D 44 04 0880            STA DISP+3
03C8-8D 47 04 0890            STA DISP+6
03CB-A9 2E    0900            LDA #$2E
03CD-8D 4A 04 0910            STA DISP+9
03D0-A9 0D    0920            LDA #$0D
03D2-8D 4E 04 0930            STA DISP+13
03D5-A9 01    0940            LDA #$01
03D7-A2 0D    0950            LDX #$0D
03D9-9D 41 D8 0960 LBLE       STA DSPCLR,X
03DC-CA       0970            DEX
03DD-D0 FA    0980            BNE LBLE
03DF-4C A7 02 0990            JMP ALARM
03E2-78       1000            SEI
03E3-A9 31    1010            LDA #L,INTPTR
03E5-EA       1020            NOP
03E6-8D 14 03 1030            STA CINV
03E9-A9 EA    1040            LDA #H,INTPTR
03EB-EA       1050            NOP
03EC-8D 15 03 1060            STA CINV+1
03EF-58       1070            CLI
03F0-60       1080            RTS
              1090 ;
              1100 ;UNUSED P. 2 AREA
              1110 ;
              1120            .BA $02A7
              1130 ;
02A7-AD 0D DC 1140            LDA CIAINT
02AA-29 04    1150            AND #$04
02AC-F0 03    1160            BEQ BEGIN
02AE-8D E3 02 1170            STA FLAG
02B1-AD E3 02 1180 BEGIN      LDA FLAG
02B4-F0 2A    1190            BEQ RETURN
02B6-AD A2 00 1200            LDA $00A2
02B9-6A       1210            ROR A
02BA-6A       1220            ROR A
02BB-6A       1230            ROR A
02BC-29 0C    1240            AND #$0C
02BE-8D 20 D0 1250            STA BORDER
02C1-29 04    1260            AND #$04
02C3-8D 18 D4 1270            STA SIDVOL
02C6-F0 0B    1280            BEQ LBL
02C8-A2 05    1290            LDX #$05
02CA-BD E3 02 1300 LOOP       LDA FLAG,X
02CD-9D 21 04 1310            STA SCRMSG,X
02D0-CA       1320            DEX
02D1-D0 F7    1330            BNE LOOP
02D3-AD C5 00 1340 LBL        LDA $00C5
02D6-C9 04    1350            CMP #$04
02D8-D0 06    1360            BNE RETURN
02DA-8E E3 02 1370            STX FLAG
02DD-8E 18 D4 1380            STX SIDVOL
02E0-4C 31 EA 1390 RETURN     JMP INTPTR
02E3-00       1400 FLAG       .BYO
02E4-41 4C 41 1410            .BY 'ALARM'
02E7-52 4D                                MICRO
              1420            .EN
```

MICRO™ Commodore Compass



by Loren Wright

Low-cost Word Processing for C-64

Commodore has been bringing out a great deal of software for the Commodore 64 lately. Most of it is very good and most of it is priced less than competing products. *Easy Script* is no exception.

It is very much like Steve Punter's *Word Pro 3 Plus/64* (sold by Professional Software and Pro-Line and reviewed earlier in this column). In fact, the overall design and command syntax are nearly identical. There are several differences, and most of them work in favor of *Easy Script*.

Like *Word Pro 3 Plus/64* (which I hereafter refer to as simply *Word Pro*), *Easy Script* uses a wordstream format, which results in words being split across the end of a screen line. Screens of the two word processors look very similar. *Easy Script*'s is a bit easier to follow because the cursor flashes and because line endings and format chracters appear in reverse field. *Easy Script* allows you to set a working screen width of up to 80 characters. This makes working with tabular and indented material much easier, but typing on this wider screen is not very convenient due to the necessary panning across the 40-column screen. There is also an output-to-video function (lacking in the C-64 version of *Word Pro*), which allows you to see what your document looks like before you print it out. While viewing the video output you can select any page or pages for printing out. With *Word Pro* you get all or nothing.

Easy Script can be used with either cassette or disk, but not both at the same time. Editing is more convenient, particularly since there are true block-delete, -transfer, and -copy commands. *Word Pro* only allows these operations on whole screen lines. There is also a major difference in the files produced by the two word processors. *Word Pro* produces *program* files, while *Easy Script* produces *sequential* files. Sequential files are more accessible from other programs, including your own BASIC programs. *Easy Script* allows considerably more text in memory at one time — 764 lines *vs.* 329. It is also possible to save only part of the text in memory to a disk file.

Easy Script lacks the "extra text" feature of *Word Pro*, but at least one use of it is taken care of: *Easy Script* makes it possible to get a disk directory without wiping out text in memory. Another use of extra text is not duplicated. *Easy Script* has no "append characters" or "append text" features. With *Word Pro* it is possible to label a number of frequently used phrases or text segments in extra text and call them into main text with a few keystrokes.

Word Pro was once the best word processor available for Commodore machines. It can no longer claim that honor. As each new Commodore machine has come out, a new version of *Word Pro* has become available, but instead of taking advantage of the features of the machine, only enough changes to get it running have been made. *Easy Script* is a better word processor, and, according to Jim Strasma and a number of others, *Paper Clip* from Batteries Included is also better. *Easy Script* is especially attractive because of its price. Commodore won't quote a suggested retail price, but $50 is a good guess.

The Complete Personal Accountant

Since I am now completely self-employed, I suddenly need to keep much better financial records. I was intrigued by Jim Strasma's number one rating for *Complete Personal Accountant* in last month's Commodore Buyer's Guide, so I obtained a copy with the idea of reviewing it here. I now have a good start on getting my finances in order!

Formerly called *The Color Accountant*, *Complete Personal Accountant* is actually a set of programs that work together in various ways. The heart of the package is the Chart of Accounts, which operates with the Checkbook Maintenance, Financial Statements, and Budget Analysis programs. Setting up is a little involved, but extra time spent in set-up is rewarded in time saved in maintaining your records. The first order of business is to set up your chart of accounts. There is a standard chart provided, but you will surely want to make changes, additions, and deletions. Accounts in the chart are divided into assets, income, liabilities, equity, and expense accounts. There is room for 99 different accounts, with up to 9 subcategories in each, up to a total of 300 subcategories. Next, you go through your checkbook, check by check and deposit by deposit. As you enter each item, you decide what account to credit or debit. Each check and deposit is automatically entered on the disk file. When you're done, not only have you balanced your checkbook, but you have also recorded your expenditures in the different accounts. There is provision for more than one checking account, although these files must be stored on separate disks, and you may indicate some payments to take place automatically. There is a great deal of support for error checking and for making backup copies of your disks.

When you have your checkbook balanced, you can then proceed to generating financial statements or to budget analysis. Other capabilities of these programs include generating checks from your computer, graphing results in color on the screen or on a VIC printer, and checkbook search. The other programs, which don't work with the ones mentioned above, include a Payments Calendar, Appointments Calendar, and Mailing List.

I had a little trouble figuring out when a debit subtracts from and when it adds to an account. An appendix in the excellent manual explains these terms — I wish I had read it before I started entering checks! *Complete Personal Accountant* is available from Programmer's Institute for $79.95. The cassette version is $74.95, and the package has been divided into three parts for about $30 each.

Getting Started in Machine Language

There are several things you can do with your VIC or C-64 in machine

language that you can't do in BASIC, and there are many things that can be done faster. For instance, using BASIC to clear the high resolution screen is a very slow process that takes a fraction of a second in machine language, and raster interrupt programming is virtually impossible in BASIC. As the computer comes, though, there is little that you can do beyond simple programs that you POKE in from DATA statements. Larger and more expensive computers have built-in machine language monitors, while less expensive, smaller computers, including Atari, Color Computer, VIC, and Commodore 64 do not. A monitor is a program that lets you look at and modify the contents of memory locations and processor registers, and load and save ranges of memory. An *extended* monitor is one that adds extra functions, such as a disassembler, a mini-assembler, and trace and break-point capability. Monitors are available on cartridge, disk, or cassette. Some of the better known monitors for Commodore machines are VICMON, 64MON, HESMON, SUPERMON, MICROMON, and TINYMON. There is also a minimal monitor included with Richvale Telecommunication's V-Link and 64-Link cartridges. VICMON and 64MON are cartridges available from Commodore; HESMON is available on cartridge for both machines from Human Engineered Software; and the others are disk or cassette-based monitors available as listings in magazines or from user's groups. One of the commercial cartridges will cost $40-$50. The others are free or nearly free.

You should also have a copy of the *Programmer's Reference Guide* for your computer and a good general 6502 programming book, such as Lance Laventhal's *6502 Assembly Language Programming*.

The next step is to get a full-fledged assembler. This will cost about $100. I have been using *PAL* (by Brad Templeton, sold by Pro-Line Software) and find it especially convenient because it's designed to work with *POWER*, which I reviewed here earlier. *PAL* does not have macros or conditional assembly, but it has several advantages, such as using the BASIC editor for source files. It is quite a bit smaller than others and is relatively position-independent. MAE, from Eastern House Software, is a full-featured assembler that is well supported. It is still the only assembler available for all the major 6502 machines. I have used versions for the PET since the beginning. Commodore's assembler is also an excellent package that includes macros and conditional assembly. There are other programs beyond the assembler, such as Pterodactyl's *PTD 6502/6510 Debugger*, but they are for pretty serious programming.

If you are at all serious about learning about computers, you owe it to yourself to explore machine language. Many things will suddenly become much clearer. You may not end up doing a lot of assembly language programming, but just the exercise will be rewarding.

MICRO™

You may contact Loren Wright at P.O. Box 6502, Amherst, NH 03031.

FOR APPLE II PLUS, FRANKLIN, APPLE IIe

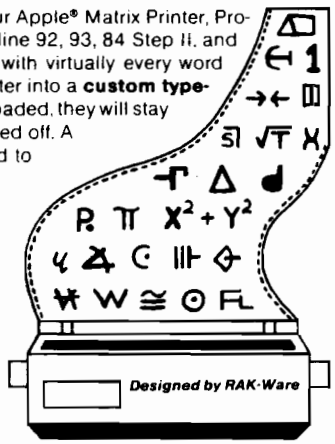
Font DownLoader*

Expand the capacity of your printer hundreds of times

Load custom fonts into your Apple® Matrix Printer, Prowriter™ 8510A, OKI® Microline 92, 93, 84 Step II, and Epson® FX and use them with virtually every word processor to turn your printer into a **custom typesetter.** After the fonts are loaded, they will stay in your printer until it's turned off. A font editor is also provided to allow you to create your own graphics, text, foreign language letters, math and electronics symbols to load into your printer.

On-Disk *(Specify Printer)* $39.00

*Requires printer interface that adheres to Apple protocall (Tymac, Apple, Epson, etc.) (Will not work with Pkaso, Wizard and some others)



$100 REWARD

Submit the best or most unique font using the above software and we will make you $100 richer. Other prizes for the first 25 runners up.

Dealer and Distributor Inquiries Invited

MasterCard VISA

MICRO WARE D

1342 B Rt. 23, Butler, NJ 07405
201-838-9027

C64-FORTH
for the Commodore 64

FORTH SOFTWARE FOR THE COMMODORE 64

C64-FORTH (TM) for the Commodore 64 - $99.95
- Fig Forth-79 implementation with extensions
- Full feature screen editor and macro assembler
- Trace feature for easy debugging
- 320x200, 2 color bit mapped graphics
- 16 color sprite and character graphics
- Compatible with VIC peripherals including disks, data set, modem, printer and cartridges
- Extensive 144 page manual with examples and application screens
- "SAVETURNKEY" normally allows application program distribution without licensing or royalties

C64-XTEND (TM) FORTH Extension for **C64-FORTH** - $59.95
(Requires original **C64-FORTH** copy)
- Fully compatible floating point package including arithmetic, relational, logical and transcendental functions
- Floating point range of 1E+38 to 2E-39
- String extensions including LEFT$, RIGHT$, and MID$
- BCD functions for 10 digit numbers including multiply, divide, and percentage. BCD numbers may by used for DOLLAR.CENTS calculations without the round-off error inherent in BASIC real numbers.
- Special words are provided for inputting and outputting DOLLAR.CENTS values
- Detailed manual with examples and applications screens

(Commodore 64 is a trademark of Commodore)

TO ORDER - Specify disk or cassette version
- Check, money order, bank card, COD's add $1.50
- Add $4.00 postage and handling in USA and Canada
- Mass. orders add 5% sales tax
- Foreign orders add 20% shipping and handling
- Dealer inquiries welcome

PERFORMANCE MICRO PRODUCTS

MasterCard

770 Dedham Street. S-2
Canton, MA 02021
(617) 828-1209

VISA

MICRO™ Commodore Reviews

Product Name: **SYSRES**
Equip. req'd: Commodore 64 and 1541 disk drive
Price: $95.00
Manufacturer: Solidus International Corp.
1060 Roosevelt Crescent
North Vancouver, BC
Canada, V7P 1M3
(604)984-0477

Description: *Sysres* is supplied on a single 5¼" disk. It extends and enhances the C-64 operating system. Originally developed for the PET, Sysres adds 33 new commands to BASIC and includes 11 DIS-support commands. The added commands rovide such features as renumbering a program, search and replace, auto-line numbering, forward and backward scrolling through a listing, and many more. Some of these new commands function in different ways, depending upon the options selected, so that altogether over one thousand new functions are added. Sysres will function with an IEEE-488 adaptor, gaining access to larger, faster CBM dual disk drives and printers. It also supports non-CBM, ASCII printers.

Pluses: Although the number of features available is large, the syntax is clear and logical — quite easily mastered. Also notable is the fact that Sysres code is "hidden" — using almost none of the available program space. It can be booted without disturbing the resident BASIC program. From end-to-end Sysres appears to be well thought out and professionally implemented. In the "programmers aid" category Sysres is definitely a Cadillac. Note that programs written using Sysres do not need Sysres to run later.

Minuses: The system is supplied on a Master diskette that is copy protected. It cannot be backed up. However, Solidus guarantees replacement free of charge within 90 days of purchase, and replacement for $10.00 thereafter.

Skill level required: This product is targeted for the serious programmer. However, it's logical, easily learned syntax should make it useful for anyone with even modest proficiency in BASIC.

Documentation: The Sysres master disk is accompanied by a 112 page user's guide in the form of a 3-ring hard-

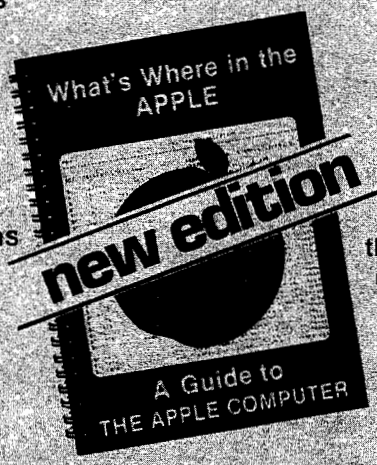
(Continued on next page)

What's Where in the Apple
A Complete Guide to the Apple Computer

This **Revised Edition** of the famous Apple Atlas provides Apple computerists with a framework for understanding both the overall structure of the Apple system and programming techniques that exploit that knowledge.

What's Where in the Apple contains the most complete memory map ever published as well as detailed information needed for actual programming.

All for only $24.95
(plus $5.00 s/h)



For owners of the original edition, MICRO is offering a companion book, ***THE GUIDE to What's Where in the Apple,*** for only **$9.95** (plus $2.00 s/h).

THE GUIDE contains all new material that explains and demonstrates how to use the Atlas and Gazetteer published in the original volume of *What's Where in the Apple?*

For Fast Service
Call our toll free number:
1-800-354-8112
(In PA 1-800-662-2444)

MICRO makes it easy to order.
Send check (payable to MICRO) to:

MICRO
P.O. Box 6502, Chelmsford, MA 01824

VISA and MasterCard accepted.

MA residents add 5% sales tax.

backed, loose-leaf notebook. However, it is very good in the description of the syntax and use of Sysres. Each command is fully explained and examples are given.

Reviewer: Roger Crites

Product Name: **Smart Ascii**
Equip. req'd: Commodore VIC-20 or '64 any memory configuration
Price: $59.95
Manufacturer: Midwest Micro Associates
P.O. Box 6148
Kansas City, MO 64110

Description: *Smart Ascii* is a software/hardware package that interfaces the Commodore VIC-20 or '64 to any parallel printer using the "Centronics standard" protocol. The package consists of a cassette tape containing the software interface (there are separate versions for the VIC and '64, both on the same tape), and a three foot cable for connecting the computer to the printer via the user port.

Pluses: *Smart Ascii* is very easy to install and responds to the same type of commands as the VIC printer (OPEN, CMD, PRINT#). It has three very useful printing options: TRANSLATE translates selected control characters into a character string (reverse on becomes "(RVS)", etc.). "CBM" ASCII prints all uppercase, for program listings. "TRUE" ASCII prints upper and lowercase for word processing applications. The software is not protected and may be backed up to cassette or disk.

Minuses: The supplied cable is only three feet long. The software disables the RESTORE key, which makes life a little difficult if a program ends or is stopped with any of the screen or sound registers not reset.

Documentation: The documentation is very complete and clear. It is well organized, and includes a table of contents. A minor complaint is that the information concerning linefeed conventions and printer control codes, both of which are essential to proper operation of the printer, are hidden in the section labelled "Advanced Programmer Tips".

Skill level required: Minimal. Some knowledge of the printer being used may be required to set the linefeed convention correctly.

Reviewer: Michael Morris

Product Name: **Fundamentals of Mathematics**
Equip. req'd: Commodore 64 with 1541 disk drive
Price: 6-disk set-$249.95
3rd grade level only-$69.95 (2 disks)
5th grade level only-$69.95 (2 disks)
9th grade level only-$99.95 (3 disks)
Worksheets for Lessons/Programs 1-89-$29.95
"Hands-on Preview" disk-$9.95
Manufacturer: Sterling Swift Publishing Co.
7901 South IH-35
Austin, TX 78744
(512)282-6840

Description: An educational set of 89 lessons and programs that may be used with children from the third grade on. The lessons cover mathematics from reading and writing two to seven digit whole numbers through equation solving and hit almost everything in between. The package is formatted for use by teachers in the classroom. Each lesson is backed up with worksheets which may be reproduced for classroom use by the students. The worksheets are broken up into pre-test, sample problem, problem, and post-test to allow use as needed to reinforce the learning process.

Pluses: The lessons are well done. In fact this is one of the best teaching packages I have worked with. When the problems re answered correctly the success is congratulated by terms such as: very good, fabulous, etc.. When a wrong answer is given, it is simply stated without any chastisement. At the end of each lesson, if more than 40% of the answers were given wrong, the program suggests that a review might help.

Minuses: The program were evidently translated from PET versions and do not made good use of the color, graphics and sound available from the Commodore 64.

Documentation: As the programs with their worksheets are self explanatory, not much addition documentation is needed or supplied. It tells how to use the programs and suggests methods for obtaining the best learning results from children whose needs vary.

Skill level required: The program set is made to be used in a learning environment. This does not limit them to teacher use only as a parent could make good use of them at home to help the children develop their skills in mathematics. Almost no specific computer knowledge is required.

Reviewer: Richard E. DeVore

Product Name: **C64-FORTH**
Equip. req'd: Commodore 64 Computer; 1541 Disk drive & printer optional
Price: $49.95
Manufacturer: Computer Marketing Services
300 W. Marlton Pike
Cherry Hill, NJ 08002
(609)795-9480

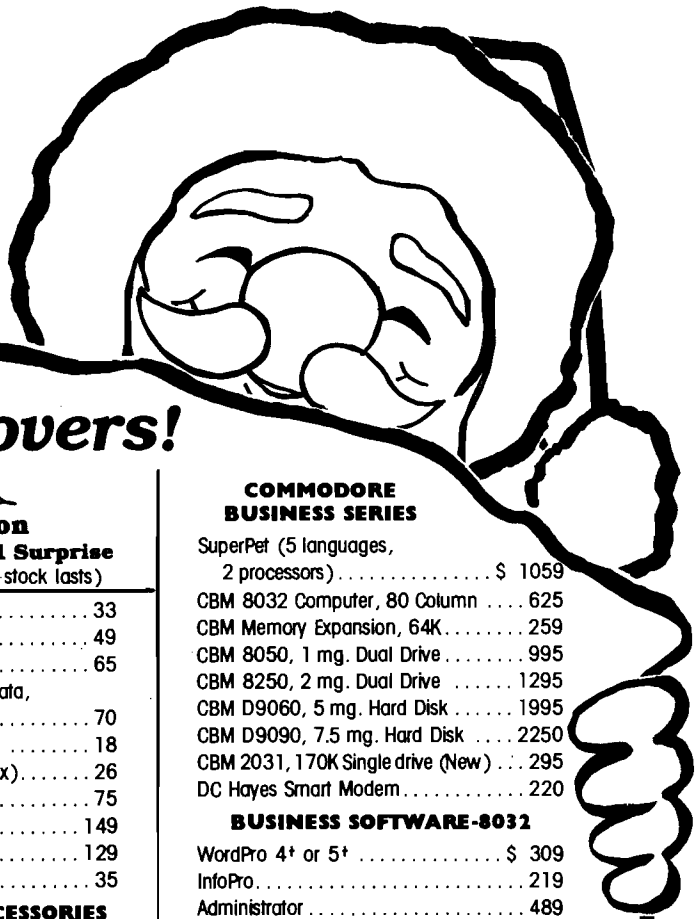
Description: *C – 64 FORTH* is a cartridge based implementation of the language. It allows programming on the Commodore 64 with a language that is transportable between systems.

Pluses: It is extremely close to a standard version of fig-FORTH and can be adapted to accept FORTH-79 standard

(Continued on page 38)

SJB DISTRIBUTORS

One Stop Shopping for COMMODORE Systems



Gift Ideas for Computer Lovers!

Holiday Specials!

Word Processing $1995.00
CBM 8032 CBM 8050 PET/IEEE CABLE
4022 PRINTER WP4 + VERBATIM DISKS

Recreation
Buy 2 Games, Get 1 Surprise Game FREE! (while stock lasts)

NEW COMMODORE PRODUCTS

The Executive 64	Call
CBM B128-80	$ 825
CBM B256-80	1095
CBM BX700	2990
B Series Software	Call
CBM 1520 Plotter	169
CBM 1526 Printer	349

SOFTWARE FOR CBM 64 BUSINESS

WordPro 3+/64w/Spell Right Plus	$ 95
Spell Right Plus	55
Calc Result (Advanced)	125
Calc Result (Easy)	75
Busicalc II	95
Mirage Concepts (Powerful Data Base)	95
M File (merge with WordPro)	89
Home Utilities	49
64 Mailing List (Galactic)	28
The Manager	50
Home Accountant (continental)	75
Code Writer (Writes Basic Programs)	95
Stock (investment analysis)	80
Agricultural Management	Call
General Ledger, A/R, A/P, P/R, Inv	Call

RECREATION

Assembler Package (cassette or disk, compiled, includes editor, loader, disassembler)	39
Sprite Master (access)	30
Neutral Zone (access)	35
Space Belt	19
Pet Emulator	30
Coco II (build your own games)	40
Vic Tree (programmers utilities)	75
Micro-Term (save to printer disk)	39
Hesmon	35
Synthesound	45
Gothmogs Lair	30
Road Toad	15
Commodore Games	Call

INTERFACES & ACCESSORIES

80 Column Expander	$ 159
VIC 1600 Modem	95
VIC 1650 (auto answer, auto dial)	150
VIC 1525 Graphic Printer	225
VIC 1530 Datasette Recorder	65
VIC 1541 Disk Drive	249
VIC Switch (connect 8 64's or Vics to printer, dd)	149
PET-IEEE cable	33
IEEE-IEEE cable (2m)	49
5 Slot Expander for 64	65
Parallel Interface (Epson, Okidata, IDS, NEC)	70
Programmers Reference Guide	18
Verbatim Diskettes (10 per box)	26
Hes Modem	75
ADA 1450	149
ADA 1800 (new)	129
Numeric Keypad	35

VIC PRODUCTS & ACCESSORIES

8K RAM Memory Expansion Cartridge	$ 40
16K RAM	70
24K RAM	105
VIC 3 Slot Expander	27
VIC 6 Slot Expander	70
Gorf (64 also)	30
Omega Race	30
Arcade Joystick - Heavy duty w/2 firing buttons! Great for the VIC or 64	25
Auto Clock	125

MONITORS - GREAT RESOLUTION (64 OR VIC)

CBM 1701 Color Monitor	$ 249
Amdek Color Plus	299
Panasonic TR-120 (w/speaker)	155
BMC (green screen)	95
Video/Audio Cable	15

PRINTERS - LETTER QUALITY

CBM 6400, 40 cps	$ 1450
Diablo 620, 25 cps	949
Transtar 140 (serial)	1395
Transtar 130, 16 cps (auto load, wp features!)	769
NEC 3500 Series	1600
NEC 7700 Series	2350
TL 20	500

PRINTERS - DOT MATRIX

CBM 8023, 150 cps/graphics	$ 545
CBM 4023 Printer	395
Epson FX Printer, 160 cps	549
Epson MX-80 FT w/graftrax	Call
Epson FX-100	859
Okidata 82A, 120 cps (serial and parallel)	429
NEC 8023A (parallel)	429
Okidata 92	559
Star Gemini, 10X	329
Star Gemini, 15	499
Transtar 315 (hi-res, color)	575

COMMODORE BUSINESS SERIES

SuperPet (5 languages, 2 processors)	$ 1059
CBM 8032 Computer, 80 Column	625
CBM Memory Expansion, 64K	259
CBM 8050, 1 mg. Dual Drive	995
CBM 8250, 2 mg. Dual Drive	1295
CBM D9060, 5 mg. Hard Disk	1995
CBM D9090, 7.5 mg. Hard Disk	2250
CBM 2031, 170K Single drive (New)	295
DC Hayes Smart Modem	220

BUSINESS SOFTWARE-8032

WordPro 4+ or 5+	$ 309
InfoPro	219
Administrator	489
VisiCalc (expanded)	199
BPI A/R, G/L, Job Cost, Inventory, Payroll	ea. 325

NOTE

SJB has a full line of computer media in stock, call or write for more information.

Product Selection Advice
Customer Service
214 - 343-1328

SJB DISTRIBUTORS INC.
10520 Plano Road, Suite 206
Dallas, Texas 75238

To Order—Call Toll Free:
800-527-4893
800-442-1048
(Within Texas)

Business Hours:
Mon. - Fri. 8:30 - 5:30
Saturday 10 - 2

POLICY

VISA/MASTERCARD add 3%.
C.O.D. for Cash or Bank Check.
Exact Freight Calculated.
Products shipped with manufacturer's warranty.
Prices and stock subject to change without notice.
F.O.B. DALLAS, TEXAS.
*Customers must call for return authorization before returning any product.
*Minimum order of $50.00.

WRITE for FREE Catalog!

code. The language itself is extremely fast. It almost equals assembly language in its speed of operation. It is an excellent medium to write games in due to the speed of operation. Once learned, FORTH is much easier to write than assembly or machine language.

Minuses: The program does not come with sufficient information to start using it if you are not already familiar with the language. A disadvantage to someone who is used to using FORTH on another computer is the fact that it is supplied in a cartridge rather than on disk. Other implementations which I am familiar with are disk based. The manual does say that it is not a textbook on FORTH and supplies a list of reference material to help get you started.

Documentation: The 34 page User's guide & Reference Manual is broken into three parts. The first portion explains FORTH differences. The second and largest section contains a glossary of the words in C-64 FORTH while the third section explains how to get C-64 FORTH to work with FORTH-79 standard code. If you know something about FORTH or are willing to learn outside of the provided information, you will find that it is a good implementation. The manual, within the above limitations, presents the information clearly.

Skill level required: User who has progressed beyond the beginner stage.

Reviewer: Richard E. DeVore

Product Name: **Passive Solar Design Program for Home Owners**
Equip. req'd: Commodore 64 with 1541 disk drive or Dattasette
Price: $99.95
Manufacturer: Don Danvlyk
1538 Ohio Ave.
Virginia Beach, VA 23454
(804)425-7792

Description: A solar design program for the Commodore 64 that helps determine the effectiveness of your design. The program will help design add-ion greenhouses or direct solar-gain passive structures. The choices are: a solar addition without heat storage; an addition with uninsulated heat storage; an addition with well insulated heat storage; and an addition using direct solar gain.

Pluses: The program gives a full financial breakout for each of your designs if desired. It also states whether or not the design is economically feasible. It does this through comparing heat savings to project cost and heat savings to interest that could have gotten from the same investment.

Minuses: The computations are approximate rather than actual. If you want to change a dimension while inputting your design, the program takes you back to the menu.

Documentation: Almost non-existent. The saving grace is that after trying the program several times, you won't need documentation.

Skill level required: Could be used by a beginning computerist.

Reviewer: Richard E. DeVore MICRO™

HOW MUCH LONGER WILL YOU LAST?

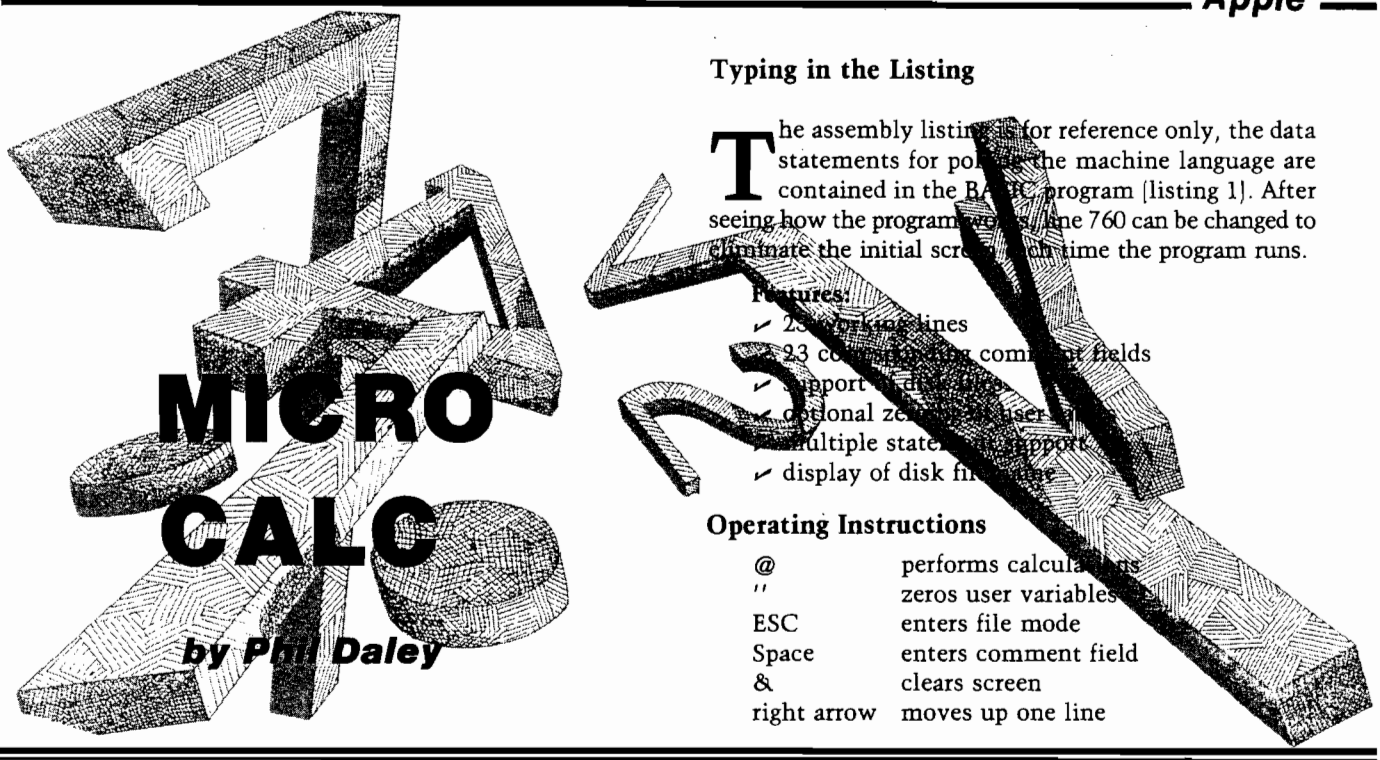
How long can you endure? When will it end?... We're not talking about a new shoot 'em up game for the Commodore 64, but Commodore's own disk operating system! Commodore made a great computer in the 64 but left its disk operating system out in the cold. If you've been waiting for a true disk operating system, here it is!... If you've been waiting for a great BASIC language enhancement that will let you utilize the Commodore's many special features, here it is! What is it? It's grafDOS, the great new utility from Xylex Software that allows the user to actually become friendly with the Commodore 64! grafDOS includes commands like DELETE, RENAME, CATALOG, RUN, etc. The BASIC allows you to do high resolution and low resolution graphics, sound, sprite program, plus much, much more for a total of 40 commands! Plus included in every package is MINIMON, a powerful machine language monitor that includes another 20 commands for use in machine language. The disk also comes with sample programs and demos including a great music generator! And all this together is only $49.95! How could you have lasted this long without it?

DON'T WAIT ANY LONGER!

Make your programming easier! grafDOS is available now at your local dealer or:

INTERESTING SOFTWARE
21101 S. Harvard Blvd.
Torrance, CA 90501
(213) 328-9422

Visa/MC/Check/Money Order
Add $2.00 shipping
CA residents add 6½# sales tax.
Dealer inquiries invited.



MICRO CALC

by Phil Daley

Typing in the Listing

The assembly listing is for reference only, the data statements for poking the machine language are contained in the BASIC program (listing 1). After seeing how the program works, line 760 can be changed to eliminate the initial screen each time the program runs.

Features:

- ✓ 23 working lines
- ✓ 23 corresponding comment fields
- ✓ support of disk files
- ✓ optional zeroing of user [illegible]
- ✓ multiple statement support
- ✓ display of disk filename

Operating Instructions

@	performs calculations
"	zeros user variables
ESC	enters file mode
Space	enters comment field
&	clears screen
right arrow	moves up one line

Listing 1

```
10  REM  *    MICROCALC
20  REM  *   BY P. DALEY
30  REM  * COPYRIGHT (C) 1983
40  REM  *   BY MICRO INK
50  FOR II = 1 TO 29:C$ = C$ + " ": NEXT :C$ = C$ +
    "*"
60 D$ =  CHR$ (4): HOME : GOTO 550
70  VTAB XX: PRINT B$(XX);
80  IF  MID$ (B$(XX),2,1) = "?" THEN  PRINT BB$(XX
    );
90  IF XX = 24 THEN  CALL  - 868: GOTO 130
100  INVERSE
110  PRINT  CHR$ (BB); RIGHT$ (C$,29 -  LEN (B$(XX
    )) -  LEN (BB$(XX)));
120  NORMAL : PRINT A$(XX);: CALL  - 868: PRINT
130  RETURN
140 XX = 1: GOSUB 70
150 AA =  PEEK ( - 16384): IF AA < 127 THEN 150
160  IF FLAG = 1 THEN FLAG = 0: FOR II = 1 TO 24:B
    B$(II) = "": NEXT
170  POKE  - 16368,0
180 AA = AA - 128
190  IF AA = 64 THEN FLAG = 1:BB = 32: GOSUB 70: GOTO
    400
200  IF AA = 38 THEN  GOSUB 640:XX = 1: GOSUB 70: GOTO
    150
210  IF AA = 34 THEN  GOSUB 1410: GOTO 150
220  IF AA = 44 OR AA = 59 OR AA = 93 THEN 350
230  IF AA > 39 AND AA < 95 THEN 320
240 BB = 32: GOSUB 70
250  IF AA = 32 THEN 360
260  IF AA = 13 THEN XX = XX + 1: IF XX > 23 THEN
    XX = 1
270  IF AA = 8 AND YY > 1 THEN B$(XX) =  LEFT$ (B$
    (XX), LEN (B$(XX)) - 1):YY = YY - 1: GOTO 290

280  IF AA = 8 AND YY = 1 THEN B$(XX) = "":YY = YY
     - 1
290  IF AA = 21 THEN XX = XX - 1: IF XX < 1 THEN X
    X = 23
300  IF AA = 27 THEN  GOTO 1100
310  GOTO 340
320 YY = YY + 1: IF YY > 28 THEN BB = 32: GOSUB 70
    :XX = XX + 1:YY = 0: GOTO 340
330 B$(XX) = B$(XX) +  CHR$ (AA)
340 YY =  LEN (B$(XX)):BB = 95: GOSUB 70
350  GOTO 150
360  VTAB XX: HTAB 31
370  INPUT A$(XX)
380  IF  LEN (A$(XX)) > 9 THEN A$(XX) =  LEFT$ (A$
    (XX),9)
390  HTAB 1: GOSUB 70:XX = XX + 1: GOTO 340
400  FOR II = 1 TO 24:CT = 0:BUF = 511: POKE 216,0

410  IF  LEN (B$(II)) < 2 THEN 530
420  IF  MID$ (B$(II),2,1) <  > "=" THEN 490
430  FOR JJ = 1 TO  LEN (B$(II))
440  IF  MID$ (B$(II),JJ,1) = ":" THEN FG = 1: GOSUB
    470: GOTO 460
450  POKE BUF + JJ, ASC ( MID$ (B$(II),JJ,1)):CT =
    CT + 1
460  NEXT JJ
470  POKE BUF + JJ,13:BUF = BUF - (CT + 1): ONERR
     GOTO 1360
480 CT = 0: CALL 768: IF FG = 1 THEN FG = 0: RETURN

490  IF  MID$ (B$(II),2,1) <  > "?" THEN 530
500  GOSUB 810
510 BB$(II) = " " +  STR$ (X1)
520 XX = II:BB = 32: GOSUB 70
530  NEXT II
540 XX = 1: GOTO 340
550  FOR II = 1 TO 29:S$ = S$ + "*": NEXT
560 SS$ = "*                          *"
570  VTAB 5: PRINT S$: FOR II = 1 TO 10
580  PRINT SS$: NEXT
590  PRINT S$: VTAB 8: HTAB 5: PRINT "MICRO CALC F
    OR APPLE"
600  VTAB 10: HTAB 5: PRINT "BY P. DALEY"
610  VTAB 12: HTAB 5: PRINT "COPYRIGHT (C) 1983"
620  DIM B$(25),A$(25),BB$(25)
630  GOSUB 690: GOTO 1190
640  INVERSE : VTAB 1
650  HOME
660  FOR II = 1 TO 23: PRINT C$
670 B$(II) = "":A$(II) = ""
680  NEXT : GOSUB 1410: NORMAL : RETURN
690  FOR II = 768 TO 805
700  READ AA: POKE II,AA: NEXT
710  DATA  165,184,72,165,185,72,169,0,133,184
720  DATA  169,2,133,185,32,89,213,169,0,133
730  DATA  184,169,2,133,185,32,70,218,104,133
```

(Continued on next page)

Listing 1 *(continued)*

```
740  DATA  185,104,133,184,96
750  DATA  104,104,96
760  REM  RETURN :REM TAKE OUT FIRST REM TO REMOVE
      STARTUP VARIABLES
770  FOR II = 1 TO 15: READ A$(II),B$(II): NEXT
780  RETURN
790  DATA PRINCIPAL,A=8000,,,NUM MNTHS,M=48,,,INT
     RTE,I=11.9,,,MNTHLY IR,I=I/1200,,
800  DATA DIVISOR,D=(1-(1+I)^-M)/I,,,MONTH RTE,P=A
     /D,,,ROUND,P=INT(P*100+.5)/100,,,PAYMENT,P?
810 JJ =  ASC ( LEFT$ (B$(II),1)) - 64
820  ON JJ GOTO 840,850,860,870,880,890,900,910,92
     0,930,940,950,960,970,980,990,1000,1010,1020,
     1030,1040,1050,1060,1070,1080,1090
830  RETURN
840 X1 = A: RETURN
850 X1 = B: RETURN
860 X1 = C: RETURN
870 X1 = D: RETURN
880 X1 = E: RETURN
890 X1 = F: RETURN
900 X1 = G: RETURN
910 X1 = H: RETURN
920 X1 = I: RETURN
930 X1 = J: RETURN
940 X1 = K: RETURN
950 X1 = L: RETURN
960 X1 = M: RETURN
970 X1 = N: RETURN
980 X1 = O: RETURN
990 X1 = P: RETURN
1000 X1 = Q: RETURN
1010 X1 = R: RETURN
1020 X1 = S: RETURN
1030 X1 = T: RETURN
1040 X1 = U: RETURN
1050 X1 = V: RETURN
1060 X1 = W: RETURN
1070 X1 = X: RETURN
1080 X1 = Y: RETURN
1090 X1 = Z: RETURN
1100  HOME : ONERR  GOTO 1190
1110  VTAB 10: INVERSE : PRINT "S";
1120  NORMAL : PRINT "AVE OR ";
1130  INVERSE : PRINT "L";
1140  NORMAL : PRINT "OAD?"
1150  PRINT : PRINT "(RETURN) FOR CATALOG."
1160  GET A$: PRINT : IF  ASC (A$) = 13 THEN  PRINT
     D$"CATALOG": GET A$: GOTO 1100
1170  IF A$ = "S" THEN  GOSUB 1200
1180  IF A$ = "L" THEN  GOSUB 1270
1190  POKE 216,0: HOME :BB = 32: FOR XX = 1 TO 24:
      GOSUB 70: NEXT :BB = 95: HTAB 1: GOTO 140
1200  PRINT : PRINT "FILENAME?": INPUT F$:F$ = F$ +
     ".CAL":B$(24) = F$
1210  PRINT D$"OPEN"F$
1220  PRINT D$"WRITE"F$
1230  FOR II = 1 TO 25
1240  PRINT A$(II): PRINT B$(II)
1250  NEXT
1260  PRINT D$"CLOSE": RETURN
1270  PRINT : PRINT "FILENAME?": INPUT F$:F$ = F$ +
     ".CAL"
1280  PRINT D$"OPEN"F$
1290  PRINT D$"READ "F$
1300  FOR II = 1 TO 25
1310  INPUT A$(II)
1320  GET A$: IF A$ =  CHR$ (13) THEN 1340
1330 B$(II) = B$(II) + A$: GOTO 1320
1340  NEXT
1350  PRINT D$"CLOSE": RETURN
1360  POKE 216,0
1370  VTAB 11: HTAB 23: FLASH
1380  PRINT "<-SYNTAX ERROR";
1390  NORMAL : CALL  - 868: HTAB 1
1400 XX = 1: GOTO 340
1410 A = 0:B = 0:C = 0:D = 0:E = 0:F = 0:G = 0:H =
     0:I = 0:J = 0:K = 0:L = 0:M = 0:N = 0:O = 0:P
      = 0:Q = 0:R = 0:S = 0:T = 0:U = 0:V = 0:W =
     0:X = 0:Y = 0:Z = 0: RETURN
```


SCIENTIFIC SOFTWARE ASSOCIATES, LTD.

Q-card

Questionnaire Analysis Software

- Microcomputer based
  Avoid the expense of contract services -- do everything in-house on your own Apple II+ microcomputer.
- Easy data entry
  Avoid time consuming keypunching. Uses respondent-marked cards entered with an Optical Mark Reader (keyboard entry also possible).
- Comprehensive data analysis
  Sort on any variable(s), tally all responses, conduct cross tabs, correlations, linear regression, frequency distributions, and more.
- Complete editing capabilities
  Weight items, derive composites, add or delete items, and more.
- Easy-to-use
  Programs are user friendly, menu driven, and interactive. No special computer expertise is required.

Call or send for more information today.

SCIENTIFIC SOFTWARE ASSOCIATES, LTD.

BOX 208 · WAUSAU, WI. 54401

TELEPHONE: (715) 845-2066

Apple II+ is a registered trademark of Apple Computer, Inc.

Listing 2

```
                        1          TTL "INPUTTING STATEMENTS"
                        2  ****************************
                        3  *
                        4  *    INPUTTING STATEMENTS
                        5  *      ON THE APPLE II
                        6  *
                        7  *     PHIL DALEY
                        8  *
                        9  TXTPTR   EPZ $B8
                       10  INPUT    EQU $200
                       11  TOKEN    EQU $D559
                       12  LET      EQU $DA46
                       13  *
                       14          ORG $300
0300 A5 B8             15  START    LDA TXTPTR
0302 48                16          PHA
0303 A5 B9             17          LDA TXTPTR+$1
0305 48                18          PHA
0306 A9 00             19          LDA #INPUT
0308 85 B8             20          STA TXTPTR
030A A9 02             21          LDA /INPUT
030C 85 B9             22          STA TXTPTR+$1
030E 20 59 D5          23          JSR TOKEN
0311 A9 00             24          LDA #INPUT
0313 85 B8             25          STA TXTPTR
0315 A9 02             26          LDA /INPUT
0317 85 B9             27          STA TXTPTR+$1
0319 20 46 DA          28          JSR LET
031C 68                29          PLA
031D 85 B9             30          STA TXTPTR+$1
031F 68                31          PLA
0320 85 B8             32          STA TXTPTR
0322 60                33  END      RTS
0323                   34          END
```

MICRO™

APPLE CAT SORT

A Catalog Sorter for the Apple II requires:

Apple II with at least 32K RAM, disk drive with DOS 3.3

by Mark Harris

When I purchased my Apple II + a few years ago, my first disk was a model of organization. With only a few programs on the disk, it was easy to find any one of them. Now, with dozens of disks and hundreds of files littered around my basement, I have become a victim of creeping overhead; an ever-increasing fraction of my time is devoted to locating files rather than using them. (Think about me with hundreds of disks-Ed.) I decided that one modest step towards putting things in order would be to alphabetize the catalog on each of my disks.

Of the 35 tracks on a standard DOS 3.3 disk, one (number 17) is set aside for keeping track of usage in the others. Most of the track is taken up by directory entries, each consisting of a file name and type, and a pointer to a track/sector list elsewhere on the disk. While these entries cannot be loaded or stored as a standard DOS file, they can be read and modified in a straightforward manner by the RWTS (Read or Write a Track and Sector) routine described on pages 94-98 of the DOS Manual. I set out to write a program which would read all current directory entries, sort them, and re-write them so that subsequent CATALOGs would list them in alphabetical order.

I first had to decide on what kind of sorting procedure to use, and whether to use BASIC or machine language. I started with the easiest combination to program: a simple bubble sort in BASIC. (For a description of all the sorts mentioned in this article, see [1].) After a few false starts (and destroyed directory tracks), I had the program functioning properly, but it took about two minutes to sort the catalog of a typical disk. I didn't know how the blame should be split between the slow speed of BASIC and my choice of sorting algorithm, so I replaced the bubble sort with the generally-faster Quick sort and tried again. The sorting time was reduced to one minute, but it was clear that BASIC was the primary culprit. I decided to throw in the towel and re-write the program in machine language. I also decided to use an insertion sort, which performs well for a short list (less than 50) that is already partially sorted. I thought this would be appropriate since (1) I anticipated re-sorting my catalogs occasionally as new programs would be added, and (2) a disk cannot catalog more than 105 files, and typically has no more than 30 or 40.

The finished product listed in this article does the sort in under one second. The user is prompted by the program to insert the disk to be alphabetized into the drive (drive 1, slot 6) and to press the RETURN key. Then the program reads the directory entries, sorts them, re-writes them to disk, and calls the CATALOG routine in DOS. From the user's point of view, upon pressing RETURN he sees the alphabetized catalog in about the same length of time required for a standard CATALOG command. It is surprising to find that the disk has been updated in this short interval.

Using the Program.

After you have keyed in and saved the program, a simple "BRUN CAT SORT" will get you under way. The program will ask for the disk to be alphabetized to be inserted into the drive. I *strongly* suggest trying the program first on disks that you have backed up, just in case you made a mistake in entering the program. Since the program tampers with track 17, which is critical to accessing the other tracks, any scrambling of data could result in the effective loss of all files on the disk. However, you can take some comfort from the fact that even if track 17 is completely clobbered, standard utilities such as "FIND T/S LISTS UTILITY" in [2] can reconstruct the disk.

How the Program Works.

All of the secrets of direct access to directory entries are given in the DOS Manual. The pertinent information is given in the description of the RWTS routine (pp.94-98) and of the diskette directory (pp.129-131). Each file on a disk has a 35 byte entry in the diskette directory on track 17. The first two bytes give the track and sector number of the track/sector list associated with the file, which in turn lists the locations of the actual data sectors. Following the track and sector numbers is a one-byte code for the file type (text, binary, etc.), then 30 bytes for the file name. Finally, the last two bytes give the number of sectors used by the file.

The first byte of the entry actually doubles as a flag. If the associated file has been deleted, an "FF" is entered in this position. If the entry has never been opened, a "00" is used. Since neither value represents a legitimate track number for file storage (track 0 is used for DOS), there is no conflict involved. The idea behind CAT SORT is to keep reading entries into a table in RAM until a "00" is encountered as the first byte. As the entries are read, if the lead byte is not "FF", the RAM address of the entry is put into a separate table. When all entries have been read, the entry table and the address table are duplicated in memory. A sort is done by swapping addresses rather than entries (this greatly speeds up the process) in one of the tables. When this is accomplished, entries in one entry table pointed to by the sorted addresses are transferred to the other entry table in the correct order (at the positions pointed to by the remaining address table). The altered directory is written back to disk, and the CATALOG routine is called to show the fruits of these labors.

The sort used is very straightforward. An insertion sort

uses pretty much the same algorithm that most people would use for a manual sort of a few items. Suppose I want to alphabetize a stack of index cards, each of which has a single name on it. I start by taking the first two cards and swapping them if they are out of order. I take the third card and put it in the correct position in the first two. The fourth card is then inserted into the first three, and so on.

Bibliography

1. H.S. Gentry, Sorting Techniques Explained, Kilobaud Microcomputing, Nov 81, pp.156-160.
2. Worth & Lechner, Beneath Apple DOS, Quality Software, Reseda, CA 1981.

MICRO™

You may contact Mark Harris at Dept. of Mathematical Sciences, Appalachian State University, Boone, NC 28608.

Listing

```
                        * CAT SORT

                        * BY MARK HARRIS

                        * SORTS AND REWRITES THE
                        * CATALOG OF THE DISK IN
                        * DRIVE 1. USE ONLY WITH
                        * STANDARD DOS 3.3 DISKS

                                START

                        DIR         GEQU  $FE
                        TBL1        GEQU  $FC
                        A1          GEQU  $3C
                        A2          GEQU  $3E
                        A4          GEQU  $42
                        CH          GEQU  $24
                        DOSWARM     EQU   $3D0
                        CATALOG     EQU   $A56E
                        MOVE        EQU   $FE2C
                        COUT        EQU   $FDED
                        RDKEY       EQU   $FD0C
                        HOME        EQU   $FC58
                        TABV        EQU   $FB5B

                                ORG   $4200

0031  4200 A200      AGAIN     LDX   #$00
0032  4202 8E0002               STX   $200
0033  4205 86FC                 STX   TBL1
0034  4207 8E6D44               STX   NUMADR
0035  420A 2058FC               JSR   HOME
0036  420D   * PRINT TITLE AND WAIT FOR (RET):
0037  420D A90C                 LDA   #$0C
0038  420F 8524                 STA   CH
0039  4211 A902                 LDA   #$02
0040  4213 205BFB               JSR   TABV
0041  4216 A971                 LDA   #MSGTTL
0042  4218 85FE                 STA   DIR
0043  421A A944                 LDA   /MSGTTL
0044  421C 85FF                 STA   DIR+1
0045  421E 205243               JSR   MOUT
0046  4221 A90C                 LDA   #$0C
0047  4223 8524                 STA   CH
0048  4225 A904                 LDA   #$04
0049  4227 205BFB               JSR   TABV
0050  422A A980                 LDA   #MSGNM
0051  422C 85FE                 STA   DIR
0052  422E A944                 LDA   /MSGNM
0053  4230 85FF                 STA   DIR+1
0054  4232 205243               JSR   MOUT
0055  4235 A900                 LDA   #$00
0056  4237 8524                 STA   CH
0057  4239 A90A                 LDA   #$0A
0058  423B 205BFB               JSR   TABV
0059  423E A98F                 LDA   #MSGINS
0060  4240 85FE                 STA   DIR
0061  4242 A944                 LDA   /MSGINS
0062  4244 85FF                 STA   DIR+1
0063  4246 205243               JSR   MOUT
0064  4249 200CFD   GETCR       JSR   RDKEY
0065  424C C98D                 CMP   #$8D
0066  424E D0F9                 BNE   GETCR
0067  4250                    * SET UP IOB FOR READING
                                DIRECTORY ENTRIES
0068  4250                    * INTO $2000-2FFF:
0069  4250 A901                 LDA   #1
0070  4252 8D6444               STA   IBCMD
0071  4255 A940                 LDA   #$40
0072  4257 85FD                 STA   TBL1+1
0073  4259 A90F                 LDA   #$0F
0074  425B 8D5D44               STA   IBSECT
0075  425E A92F                 LDA   #$2F
0076  4260 8D6144               STA   IBBUFP+1
0077  4263                    * KEEP READING SECTORS UNTIL A '00'
0078  4263                    * IS FOUND IN THE TRACK POSITION OF
0079  4263                    * A DIRECTORY ENTRY:
0080  4263 85FF     LP1         STA   DIR+1
0081  4265 204843               JSR   RWTS
0082  4268 A90B                 LDA   #$B
0083  426A 85FE                 STA   DIR
0084  426C A000                 LDY   #$00
0085  426E B1FE     NXTENT      LDA   (DIR),Y
0086  4270 F02F                 BEQ   DONERD
0087  4272 C9FF                 CMP   #DIR+1
0088  4274 F00F                 BEQ   ADD23
0089  4276 A5FE                 LDA   DIR
0090  4278 91FC                 STA   (TBL1),Y
0091  427A E6FC                 INC   TBL1
0092  427C A5FF                 LDA   DIR+1
0093  427E 91FC                 STA   (TBL1),Y
0094  4280 E6FC                 INC   TBL1
0095  4282 EE6D44               INC   NUMADR
0096  4285 18       ADD23       CLC
0097  4286 A923                 LDA   #$23
0098  4288 65FE                 ADC   DIR
0099  428A 85FE                 STA   DIR
0100  428C D0E0                 BNE   NXTENT
0101  428E AD5D44     NXTSEC      LDA   IBSECT
0102  4291 C901                 CMP   #1
0103  4293 F00C                 BEQ   DONERD
0104  4295 CE5D44               DEC   IBSECT
0105  4298 CE6144               DEC   IBBUFP+1
0106  429B AD6144               LDA   IBBUFP+1
0107  429E 4C6342               JMP   LP1
0108  42A1                    * COPY TRACK DUMP TO $3000-3FFF:
0109  42A1 AD6044     DONERD      LDA   IBBUFP
0110  42A4 853C                 STA   A1
0111  42A6 8542                 STA   A4
0112  42A8 AD6144               LDA   IBBUFP+1
0113  42AB 853D                 STA   A1+1
0114  42AD 18                   CLC
0115  42AE 6910                 ADC   #$10
0116  42B0 8543                 STA   A4+1
0117  42B2 A9FF                 LDA   #DIR+1
0118  42B4 853E                 STA   A2
0119  42B6 A92F                 LDA   #$2F
0120  42B8 853F                 STA   A2+1
0121  42BA 202CFE               JSR   MOVE
0122  42BD                    * COPY $4000-40FF TO $4100-41FF:
0123  42BD A900                 LDA   #$0
0124  42BF 853C                 STA   A1
0125  42C1 8542                 STA   A4
0126  42C3 A9FF                 LDA   #DIR+1
0127  42C5 853E                 STA   A2
0128  42C7 A940                 LDA   #$40
0129  42C9 853D                 STA   A1+1
0130  42CB 853F                 STA   A2+1
```

```
0131  42CD A941            LDA   #$41
0132  42CF 8543            STA   A4+1
0133  42D1 202CFE          JSR   MOVE
0134  42D4                 * DO INDEX SORT ON ADDRESS
                             LIST AT $4100:
0135  42D4 20CE43          JSR   SORT
0136  42D7 AD6D44          LDA   NUMADR
0137  42DA 0A              ASL   A
0138  42DB 8D6E44          STA   LSTBYT
0139  42DE                 * PUT ENTRIES IN ALPHABETICAL
                             ORDER:
0140  42DE                 *  MOVE ENTRIES INDEXED BY
                             ADDRESS LIST AT
0141  42DE                 *  $4100 (POINTING TO
                             $3000-3FFF) TO
0142  42DE                 *  TO POSITIONS ($2000-2FFF)
                             POINTED TO
0143  42DE                 *  BY LIST AT $4000.
0144  42DE A200            NXTMV LDX   #$00
0145  42E0 BD0080          LDA   TABL1,X
0146  42E3 85FC            STA   TBL1
0147  42E5 BD0041          LDA   TABL2,X
0148  42E8 85FE            STA   DIR
0149  42EA E8              INX
0150  42EB BD0080          LDA   TABL1,X
0151  42EE 85FD            STA   TBL1+1
0152  42F0 BD0041          LDA   TABL2,X
0153  42F3 18              CLC
0154  42F4 6910            ADC   #$10
0155  42F6 85FF            STA   DIR+1
0156  42F8                 * MOVE ENTRY POINTED TO BY
                             (DIR) TO
0157  42F8                 * POSITION POINTED TO BY (TBL1).
0158  42F8 204C44          JSR   MOVENT
0159  42FB E8              INX
0160  42FC EC6E44          CPX   LSTBYT
0161  42FF D0DF            BNE   NXTMV
0162  4301                 * WRITE SORTED LIST TO DISK:
0163  4301 A902            LDA   #$02
0164  4303 8D6444          STA   IBCMD
0165  4306 AD5D44          LDA   IBSECT
0166  4309 8D6F44          STA   J
0167  430C A90F            LDA   #$0F
0168  430E 8D5D44          STA   IBSECT
0169  4311 A92F            LDA   #$2F
0170  4313 8D6144          STA   IBBUFP+1
0171  4316 204843          NXTWRT JSR  RWTS
0172  4319 AD5D44          LDA   IBSECT
0173  431C CD6F44          CMP   J
0174  431F F009            BEQ   FINISH
0175  4321 CE5D44          DEC   IBSECT
0176  4324 CE6144          DEC   IBBUFP+1
0177  4327 4C1643          JMP   NXTWRT
0178  432A 206EA5          FINISH JSR  CATALOG
0179  432D A914            ASKAGN LDA  #$14
0180  432F 85FE            STA   DIR
0181  4331 A945            LDA   #$45
0182  4333 85FF            STA   DIR+1
0183  4335 205243          JSR   MOUT
0184  4338 200CFD          JSR   RDKEY
0185  433B C9D9            CMP   #'Y'
0186  433D D003            BNE   EXIT
0187  433F 4C0042          JMP   AGAIN
0188  4342 2058FC          EXIT  JSR   HOME
0189  4345 4CD003          JMP   DOSWARM
0190  4348
0191  4348                 * SUBROUTINES:
0192  4348
0193  4348 A944            RWTS  LDA   /IOB
0194  434A A058            LDY   #IOB
0195  434C 20D903          JSR   $3D9
0196  434F B00E            BCS   ERROR
0197  4351 60              RTS
0198  4352
0199  4352 A000            MOUT  LDY   #$00
0200  4354 B1FE            NXTOUT LDA  (DIR),Y
0201  4356 F006            BEQ   RET
0202  4358 20EDFD          JSR   COUT
0203  435B C8              INY
0204  435C D0F6            BNE   NXTOUT
0205  435E 60              RET   RTS
0206  435F
0207  435F AD6544          ERROR LDA   IBSTAT
0208  4362 C910            CMP   #$10
0209  4364 D031            BNE   NOTWP
0210  4366 2058FC          JSR   HOME
0211  4369 A900            LDA   #$00
0212  436B 8524            STA   CH
0213  436D A908            LDA   #$08
0214  436F 205BFB          JSR   TABV
0215  4372 A9CD            LDA   #MSGWP
0216  4374 85FE            STA   DIR
0217  4376 A944            LDA   /MSGWP
0218  4378 85FF            STA   DIR+1
0219  437A 205243          JSR   MOUT
0220  437D A900            LDA   #$00
0221  437F 8524            STA   CH
0222  4381 A90A            LDA   #$0A
0223  4383 205BFB          JSR   TABV
0224  4386 A9EC            LDA   #MSGPRESS
0225  4388 85FE            STA   DIR
0226  438A A944            LDA   /MSGPRESS
0227  438C 85FF            STA   DIR+1
0228  438E 205243          JSR   MOUT
0229  4391 200CFD          JSR   RDKEY
0230  4394 4C0042          JMP   AGAIN
0231  4397 2058FC          NOTWP JSR   HOME
0232  439A A90F            LDA   #$0F
0233  439C 8524            STA   CH
0234  439E A908            LDA   #$08
0235  43A0 205BFB          JSR   TABV
0236  43A3 A909            LDA   #MSGIOER
0237  43A5 85FE            STA   DIR
0238  43A7 A945            LDA   /MSGIOER
0239  43A9 85FF            STA   DIR+1
0240  43AB 205243          JSR   MOUT
0241  43AE A900            LDA   #$00
0242  43B0 8524            STA   CH
0243  43B2 A916            LDA   #$16
0244  43B4 205BFB          JSR   TABV
0245  43B7 A9EC            LDA   #MSGPRESS
0246  43B9 85FE            STA   DIR
0247  43BB A944            LDA   /MSGPRESS
0248  43BD 85FF            STA   DIR+1
0249  43BF 205243          JSR   MOUT
0250  43C2 200CFD          JSR   RDKEY
0251  43C5 2058FC          JSR   HOME
0252  43C8 4C2D43          JMP   ASKAGN
0253  43CB 200CFD          JSR   RDKEY
0254  43CE
0255  43CE                 * INSERTION SORT
0256  43CE                 *
0257  43CE                 * TABLE OF ADDRESSES MUST START
0258  43CE                 * AT $4100
0259  43CE                 * NUMADR MUST CONTAIN #
                             ADDRESSES
0260  43CE                 *
0261  43CE                 KADR  GEQU  $EC
0262  43CE                 LADR  GEQU  $EE
0263  43CE                 TABL1 EQU   TABL1
0264  43CE                 TABL2 EQU   $4100
0265  43CE
0266  43CE A901            SORT  LDA   #1
0267  43D0 8D6F44          STA   J
0268  43D3 AD6F44          LDA   J
0269  43D6 0A              JLP   ASL   A
0270  43D7 A8              TAY
0271  43D8 B90041          LDA   TABL2,Y
0272  43DB 85EC            STA   KADR
0273  43DD C8              INY
0274  43DE B90041          LDA   TABL2,Y
0275  43E1 85ED            STA   KADR+1
0276  43E3 AC6F44          LDY   J
0277  43E6 88              DEY
0278  43E7 8C7044          STY   L
```

(Continued on next page)

```
0279  43EA AD7044   LLP       LDA   L
0280  43ED 0A                 ASL   A
0281  43EE A8                 TAY
0282  43EF B90041             LDA   TABL2,Y
0283  43F2 85EE               STA   LADR
0284  43F4 C8                 INY
0285  43F5 B90041             LDA   TABL2,Y
0286  43F8 85EF               STA   LADR+1
0287  43FA A003               LDY   #3
0288  43FC B1EC     NXTCHR    LDA   (KADR),Y
0289  43FE D1EE               CMP   (LADR),Y
0290  4400 D007               BNE   DNCMP
0291  4402 C021               CPY   #33
0292  4404 F003               BEQ   DNCMP
0293  4406 C8                 INY
0294  4407 D0F3               BNE   NXTCHR
0295  4409 900A     DNCMP     BCC   NXTITM
0296  440B AC7044             LDY   L
0297  440E C8                 INY
0298  440F 203D44             JSR   RPLC
0299  4412 4C3144             JMP   NXTJ
0300  4415 AD7044   NXTITM    LDA   L
0301  4418 0A                 ASL   A
0302  4419 A8                 TAY
0303  441A C8                 INY
0304  441B C8                 INY
0305  441C A5EE               LDA   LADR
0306  441E 990041             STA   TABL2,Y
0307  4421 C8                 INY
0308  4422 A5EF               LDA   LADR+1
0309  4424 990041             STA   TABL2,Y
0310  4427 CE7044             DEC   L
0311  442A 10BE               BPL   LLP
0312  442C A000               LDY   #$00
0313  442E 203D44             JSR   RPLC
0314  4431 EE6F44   NXTJ      INC   J
0315  4434 AD6F44             LDA   J
0316  4437 CD6D44             CMP   NUMADR
0317  443A D09A               BNE   JLP
0318  443C 60                 RTS
0319  443D 98       RPLC      TYA
0320  443E 0A                 ASL   A
0321  443F A8                 TAY
0322  4440 A5EC               LDA   KADR
0323  4442 990041             STA   TABL2,Y
0324  4445 A5ED               LDA   KADR+1
0325  4447 C8                 INY
0326  4448 990041             STA   TABL2,Y
0327  444B 60                 RTS
0328  444C
0329  444C A000     MOVENT    LDY   #$00
0330  444E B1FE     NXTBT     LDA   (DIR),Y
0331  4450 91FC               STA   (TBL1),Y
0332  4452 C8                 INY
0333  4453 C023               CPY   #$23
0334  4455 D0F7               BNE   NXTBT
0335  4457 60                 RTS
0336  4458
0337  4458          IOB       EQU   *
0338  4458 01       IBTYPE    DC    H'01'
0339  4459 60       IBSLOT    DC    H'60'
0340  445A 01       IBDRVN    DC    H'01'
0341  445B 00       IBVOL     DC    H'00'
0342  445C 11       IBTRK     DC    H'11'
0343  445D 00       IBSECT    DC    H'00'
0344  445E 6944     IBDCTP    DC    A'DEVTBL'
0345  4460 002F00   IBBUFP    DC    H'002F0000'
      4463 00
0346  4464 00       IBCMD     DC    H'00'
0347  4465 00       IBSTAT    DC    H'00'
0348  4466 00       IBSMOD    DC    H'00'
0349  4467 60       IOBFSN    DC    H'60'
0350  4468 01       IOBFDN    DC    H'01'
0351  4469 0001EF   DEVTBL    DC    H'0001EFD8'
      446C D8
0352  446D 00       NUMADR    DC    H'00'
0353  446E 00       LSTBYT    DC    H'00'
0354  446F 00       J         DC    H'00'
0355  4470 00       L         DC    H'00'
0356  4471
0357  4471          * MESSAGE LIST:
0358  4471
0359  4471 C3C1D4   MSGTTL    DC    C'CATALOG
      4474 C1CFC7                   SORTER'
      4477 C7A0D3
      447A CFD2D4
      447D C5D2
0360  447F 00                 DC    H'00'
0361  4480 C2D9A0   MSGNM     DC    C'BY MARK
      4483 CDC1D2                   HARRIS'
      4486 CBA0C8
      4489 C1D2D2
      448C C9D3
0362  448E 00                 DC    H'00'
0363  448F C9CED3   MSGINS    DC    C'INSERT DISK
      4492 C5D2D4                   TO BE
      4495 A0C4C9                   ALPHABETIZED'
      4498 D3CBA0
      449B D4CFA0
      449E C2C5A0
      44A1 C1CCD0
      44A4 C8C1C2
      44A7 C5D4C9
      44AA DAC5C4
0364  44AD 8D                 DC    H'8D'
0365  44AE C9CEA0             DC    C'IN DRIVE 1
      44B1 C4D2C9                   AND PRESS
      44B4 D6C5A0                   RETURN...'
      44B7 B1A0C1
      44BA CEC4A0
      44BD D0D2C5
      44C0 D3D3A0
      44C3 D2C5D4
      44C6 D5D2CE
      44C9 AEAEAE
0366  44CC 00                 DC    H'00'
0367  44CD D2C5CD   MSGWP     DC    C'REMOVE
      44D0 CFD6C5                   WRITE-PROTECT
      44D3 A0D7D2                   TAB, THEN'
      44D6 C9D4C5
      44D9 ADB0D2
      44DC CFD4C5
      44DF C3D4A0
      44E2 D4C1C2
      44E5 ACA0D4
      44E8 C8C5CE
0368  44EB 00                 DC    H'00'
0369  44EC D0D2C5   MSGPRESS   DC    C'PRESS ANY
      44EF D3D3A0                   KEY TO
      44F2 C1CED9                   CONTINUE...'
      44F5 A0CBC5
      44F8 D9A0D4
      44FB CFA0C3
      44FE CFCED4
      4501 C9CED5
      4504 C5AEAE
      4507 AE
0370  4508 00                 DC    H'00'
0371  4509 C9AFCF   MSGIOER   DC    C'I/O ERROR.'
      450C A0C5D2
      450F D2CFD2
      4512 AE
0372  4513 00                 DC    H'00'
03/3  4514 8D8D     MSGAGN    DC    H'8D8D'
03/4  4516 D3CFD2             DC    C'SORT
      4519 D4A0C1                   ANOTHER DISK?
      451C CECFD4                   (Y,N) '
      451F C8C5D2
      4522 A0C4C9
      4525 D3CBBF
      4528 A0A8D9
      452B ACCEA9
      452E A0
0375  452F 00                 DC    H'00'
0376  4530
0377  4530                   END
```

Fast and restless.

The new Delta-15 printer kicks out AnyCalc, easy as 1,2,3!

In a 9 to 5 world full of changing spreadsheet data, you need a business printer that moves fast. A printer that constantly fires out printed information. A printer called Delta-15.

Delta has the ability to print multi-copy spreadsheets at an intense 160 cps. Its throughput never rests. In fact , it never even wavers in speed. That means that Delta constantly fits the most work into every single second.

Delta comes in a 10″ or 15½″ carriage size. It offers you the flexibility of standard 8K parallel and serial interface. And has the ability to underline, accept macro instruction, and print characters that range from full graphics to everyday printing to scientific notations.

Plus, as always, you get our unique 180 day warranty (90 days on print head).

So for everyone who needs their spreadsheet data "yesterday," this is as close as you can come! The new fast and restless Delta-15 business printer from Star.

star™
micronics·inc

THE POWER BEHIND THE PRINTED WORD.

Computer Peripherals Division
P.O. Box 612186, Dallas/Ft. Worth Airport, TX 75261 (214) 456-0052

Richvale Telecommunications

10610 BAYVIEW (Bayview Plaza)
RICHMOND HILL, ONTARIO, CANADA L4C 3N8
(416) 884-4165

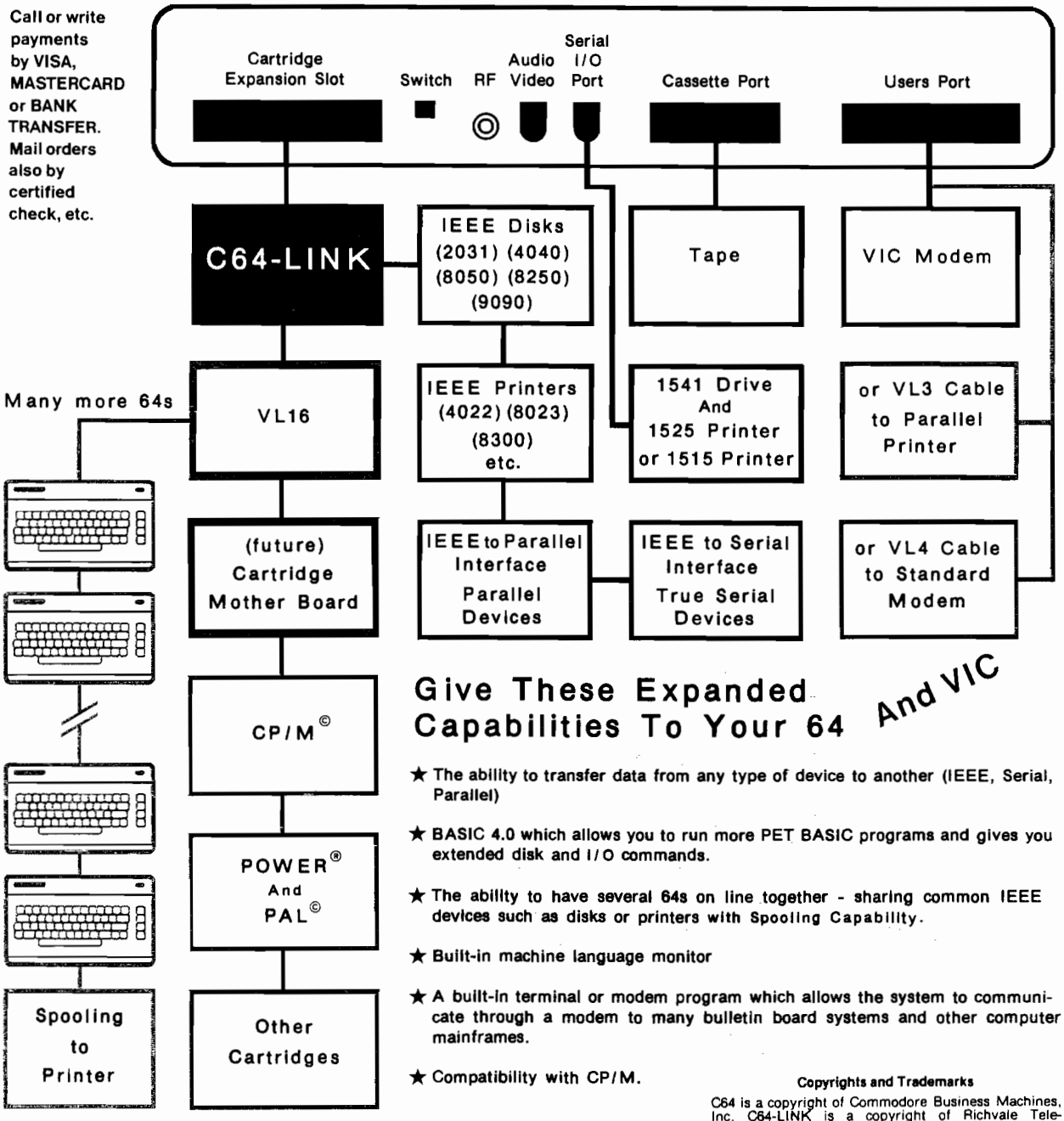
$185.00 Canadian
$149.00 U.S.
PLUS CUSTOMS BROKERAGE, HANDLING AND MAILING CHARGE.

Also available for VIC 20

C64-LINK©
The Smart 64

RTC RTC

Call or write payments by VISA, MASTERCARD or BANK TRANSFER. Mail orders also by certified check, etc.



Give These Expanded Capabilities To Your 64 And VIC

★ The ability to transfer data from any type of device to another (IEEE, Serial, Parallel)

★ BASIC 4.0 which allows you to run more PET BASIC programs and gives you extended disk and I/O commands.

★ The ability to have several 64s on line together - sharing common IEEE devices such as disks or printers with Spooling Capability.

★ Built-in machine language monitor

★ A built-in terminal or modem program which allows the system to communicate through a modem to many bulletin board systems and other computer mainframes.

★ Compatibility with CP/M.

Contact your local Commodore dealer or RTC.

Copyrights and Trademarks

C64 is a copyright of Commodore Business Machines, Inc. C64-LINK is a copyright of Richvale Telecommunications. CP/M is a registered trademark of Digital Research. POWER is a trademark of Professional Software. PAL is a copyright of Brad Templeton.

Master Directory For The Apple

By Charles Hill

Keep all your disk directories on a master file. Sort and Print the file for a quick reference to all your programs.

(Editor's note: This program is much longer than we normally publish. Since we think this is such an outstanding program (similar commercial ventures sell for over $100), we are publishing it in two pieces. This month contains all the main routines for the menu and reading/writing the library file. Next month we will conclude the program with the print and sort routines.)

It never fails. No matter how hard you try to keep your disk library in some semblance of rational order, it never seems to remain that way for long. Files that you know are on a particular disk have disappeared, and others have mysteriously moved from one disk to another. If this sounds familiar to you, then here is the solution. MASTER DIRECTORY allows you to create one large file directory containing all the CATALOGS of your disks. There is room for 64 disk IDs and 1100 file names. This directory can be sorted and printed in a variety of ways and saved to disk for use later or by other programs.

Operating Instructions

To get started, simply type "BRUN MASTER DIRECTORY". The main menu will then be displayed — six options are available. Press the key corresponding to the number of your choice. If you make a mistake, press return as the next input and you will be returned to the menu.

The first option is to read the individual disk directories. The program can only read DOS 3.3 disks. Pascal and CP/M disks use a different directory format and protected disks can't be read at all. To read the disk, insert it in drive one, enter the disk ID (1 to 8 characters) and press return. When you have CATALOGed all your disks, press return to re-enter the main menu.

The next two options allow the master directory to be saved to or read from the disk. Insert the disk into drive 1 and enter the filename. Any DOS errors that occur will be trapped and the appropriate error message printed.

Option 4 is for sorting the directory. A Shell-Metzner sort is used — it can sort 360 entries on two fields in nine seconds. To select the sort fields, enter the number next to the field name on the sort menu. One to three fields can be entered in any order. The first field entered is the most important descending to the last entered being the least important. The sort returns to the main menu when finished.

Printing the directory is the fifth option. Similar to the sort, up to three fields can be entered for printing in any desired order. To select the field, enter the number of the field from the sort mini-menu. A page eject is issued after each 65 lines. Be sure that top-of-form is set to the top of the page before printing. If this is not done, page breaks will occurr during the middle of a page. This routine also returns to the main menu.

The last option restores normal DOS and does a BASIC cold-start.

The Program

The program doesn't have a search function because in the time taken to load the directory and find the desired file name will take longer than to look it up in an alphabetized list kept next to your Apple. I keep one of these lists handy at all times. It has proven an invaluable time saver.

The first step is to set MAXFILES equal to 1. Modifications are made to DOS to allow direct access through use of machine language. This technique was described by William Reynolds III in his article *Using Text Files From Machine Language* in NIBBLE (2:2). Another modification allows the interception of DOS errors after the error message has been printed. The menu box is then set up and protected by lowering the top of the text screen. HIMEM is then lowered by 32 bytes to prevent overwriting DOS. The main loop is entered and a keypress is checked for to choose the correct subroutine. After completion of the main program, DOS is restored to it's original condition and the program jumps to BASIC.

(Continued on next page)

The routine to read the CATALOG first zeroes the disk ID buffer. Each disk ID is stored in this buffer with an index to this name stored with each filename. The reading of individual entries is simple — consecutive directory sectors are read and processed. Each entry is checked to see if it was deleted or the end of the directory. When a good entry is found, the disk index, file type and file name are copied into the name buffer. The buffer pointer is incremented and a memory check is done with appropriate error handling. Then the next entry is read.

The routines to read and save the name file on disk enter the values normally set by BSAVE and BLOAD. Drive 1 is defaulted in the program, however, this may be changed. You may wonder why I use DOS directly rather than printing the commands (preceded by CTRL-D) to execute them. The reason is that DOS stores the letters of a command being printed in the input buffer at $200. Since this is the buffer where I was reading the filename from while it was being printed, some very strange conflicts occurred. This problem took some time to find, but the new arrangement works perfectly. One item not mentioned in Reynold's article is that the KEY WORDS FOUND byte at $AA65 must be set appropriately for some routines to work correctly. The following is a list of these values (Table 1):

KEYWORD	VALUE
C	$C0
I	$A0
O	$90
V	$40
D	$20
S	$10
L	$08
R	$04
B	$02
A	$01

TABLE 1.

For example, with the BSAVE command, both the Address and Length parameters must be specified, so $08 + $01 = $09. On the other hand, no parameters need be specified with a BLOAD, so the value is $00.

The Shell-Metzner sort has appeared innumerable times with full explanations in MICRO and other magazines, so I won't go into any detail here. The only part I will mention is the comparison of Disk IDs. For the file type and name, a direct comparison is made in the entries in the file names buffer. The disk ID is a single number of no alphabetic significance. It must first be converted to an address in the disk ID buffer. The IDs then pointed at are compared and a swap of the entries in the buffer made, if needed. The disk IDs are never reordered, they stay in the buffer in their original entry order and are accessed by pointer only.

There is a ROM multiply routine that I recently discovered that may be of use in your own programming. Put the numbers to be multiplied in $64,65 and $AD,AE. After calling $E2B8, the result returns in the X register (low) and Y register (high).

After getting the numbers corresponding for the fields to be printed, the print routine converts these numbers to a range of 0-2. This value is used to test which field to print. The printer is initialized with a "PR#1". If your printer requires additional initialization, you will have to insert this code into the routine or it may be possible to initialize the printer before running the MASTER DIRECTORY program. The needed titles are printed and the fields are printed centered under the titles. The disk ID and file name are copied directly from the buffer. The file type is printed by borrowing some code from DOS at $ADDB with some changes so that LISA files are indicated with an "L". When the printing is finished, a "PR#0" disconnects the printer.

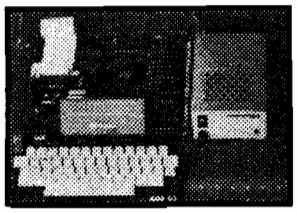
Modifications

One useful modification would be to write additional CATALOG read subroutines that could read Pascal, CP/M, Flex or OS9 directories. The code is not that complex so that changing the tracks/sectors and bytes read should be straightforward. If you need more information — the file size, disk volume number, free space remaining or other parameters — it can be read from the disk, stored and printed. Those of you with a 16K (or larger) RAM card can increase the buffer size by expanding into the additional RAM.

You may contact Charles Hill at 226 Park St., Brandon, Manitoba Canada R7A 5M3.

AIM + POWER
from COMPUTECH

All prices **Postpaid** (Continental U.S. — otherwise $2 credit)



Check the **outstanding documentation** supplied with AIM65

Top quality power supply designed to Rockwell's specs for fully populated AIM 65 — includes overvoltage protection, transient suppression, metal case and power cable:
PSSBC-A (5V 2A Reg; 24V .5A Avg, 2.5A Peak, Unreg) ...**$64.95**
Same but an extra AMP at 5 volts to drive your extra boards:
PSSBC-3 (5V 3A Reg; 24V .5A Avg, 2.5A Peak, unreg) ...**$74.95**

The **professional's** choice in microcomputers:
AIM65/1K RAM.....**$429.95** BASIC (2 ROMS) **$59.95**
AIM65/4K RAM.....**$464.95** ASSEMBLER (1 ROM) ..**$32.95**
FORTH (2 ROMS) .....**$59.95.**

SAVE EVEN MORE ON COMBINATIONS
AIM65/1K + PSSBC-A .**$479.95** AIM65/4K + PSSBC-3 .**$524.95**
We gladly quote on all AIM65/40 and RM65 items as well.

ORDERS: **(714) 369-1084**
P.O. Box 20054 • Riverside, CA 92516
California residents add 6% sales tax

Master Directory
requires:
Apple II with DOS

```
1 * MASTER DIRECTORY
* BY CHARLES HILL
* MICRO INK
* AMHERST, NH 03031
*
*   ZERO PAGE DEFINITIONS
BUFFER   EPZ $EE              ;DIRECTORY BUFFER
                               POINTER

WINTOP   EPZ $22              ;TOP OF TEXT WINDOWW
PROMPT   EPZ $33              ;INPUT PROMPT
MAX      EPZ $F0              ;SORT VARIABLES
J        EPZ $F2              ;      !
K        EPZ $F4              ;      V
HPTR     EPZ $F6
VPTR     EPZ $F8
H        EPZ $FA
V        EPZ $FC
FLDPOS   EPZ $EA              :FIELD POSITION TABLE
                               (1,2,3) FOR SORT

IDBUFFR  EPZ $FE              ;DISK ID BUFFER
                               POINTER

CV       EPZ $25              ;VERTICAL CURSOR POS.
CH       EPZ $24              ;HORIZONTAL CURSOR
                               POS.

NUMIDS   EPZ $F0              ;NUMBER OF DISK ID'S
FLDPTR   EPZ $E9              ;POINTER TO FLDPOS
                               TABLE

MAXMY    EPZ $E8              ;MAXIMUM Y VALUE
COUNT    EPZ $F0              ;NUMBER OF NAMES
                               USED BY PRINT

NUMLINES EPZ $F2              ;# OF LINES PRINTED
PRINTFLD EPZ $EA              ;TABLE OF PRINT
                               FIELDS

PRNTBUF  EPZ $F4              ;TEXT POINTER
HIMEM    EPZ $E6
*
*   ROM AND DOS DEFINITIONS
*
CROUT    EQU $FD8E            ;CARRIAGE RETURN
RDKEY    EQU $FD0C            ;GET A KEYPRESS
HOME     EQU $FC58            ;GUESS WHAT?
GETLNZ   EQU $FD67            ;GET INPUT LINE
COUT     EQU $FDED            ;PRINT CHAR. INA REG.
RDSCTR   EQU $B011            ;READ A DISK SECTOR
DIRIDX   EQU $B39C            ;INDEX INTO DIRECTORY
DIRBGN   EQU $B4C6            ;START OF DIRECTORY
                               ENTRIES

RDVTOC   EQU $AFF7            ;READ VTOC ROUTINE
NXTONE   EQU $B230            ;ROUTINE TO ADVANCE
                               DIRECTORY INDEX

DOSNAME  EQU $AA75            ;NAME BUFFER FOR DOS
VOL      EQU $AA66            ;PARAMETERS FOR DOS
DRIVE    EQU $AA68
SLOT     EQU $AA6A
LEN      EQU $AA6C
ADDR     EQU $AA72
KYWRDFND EQU $AA65            ;KEYWORDS FOUND BYTE

DOCOMND  EQU $A186            ;ROUTINE TO DO DOS
                               COMMAND

COMND    EQU $AA5F            ;DOS COMMAND
BLANKNAM EQU $A095            ;ROUTINE TO BLANK
                               NAME BUFFER

VTAB     EQU $FC24            ;VTAB ROUTINE
ROMULT   EQU $E2B8            ;ROM MULTIPLY ROUTINE
MAXFILES EQU $A258            ;MAXFILES ROUTINE
SETIN0   EQU $FE89            ;DO IN#0
SETOUT0  EQU $FE93            ;DO PR#0
SETOUT   EQU $FE95            ;DO PR#SLOT
*
*   WORKING DEFINITIONS
*
IDBUFR   EQU $0EF0            ;DISK ID BUFFER
NUMBER   EQU IDBUFR-$02       ;NUMBER OF ENTRIES
                               IN BUFFERS

NAMEBUFR EQU IDBUFR+$0200     ;FILE NAMES BUFFER
INBUFF   EQU $0200            ;INPUT BUFFER
MAXLINE  EQU 65               ;NUMBER OF LINES PER
                               PRINTED PAGE

*
* MAIN PROGRAM
*
         JSR MENU             ;SET UP THE SCREEN
         SEC                  ;SET HIMEM TO POINT
                               BELOW ACTUAL HIMEM

         LDA $73
         SBC #$20
         STA HIMEM
         LDA $74
         SBC #0
         STA HIMEM+1
         LDA #")"
         STA PROMPT
COMMAND  JSR HOME             ;DETERMINE DESIRED
                               ROUTINE AND JUMP TO
         LDA #")"              IT
         JSR COUT
         JSR RDKEY
         CMP #"1"
         BNE )1
         JMP SCANDISK
^1       CMP #"2"
         BNE )2
         JMP SAVECAT
^2       CMP #"3"
         BNE )3
         JMP READCAT
^3       CMP #"4"
         BNE )4
         JMP SORTCAT
^4       CMP #"5"
         BNE )5
         JMP PRINTCAT
^5       CMP #"6"
         BNE COMMAND
         LDA #$00             ;SET FULL WINDOW
         STA WINTOP
         JSR HOME
         LDA #$A5
         STA $A851            ;RESTORE NORMAL DOS
         LDA #$A2
         STA $9EE0
         LDA #$20
         STA $A6EF
         LDA #$51
         STA $A6F0
         LDA #$A8
         STA $A6F1
         JMP $03D3            ;COLD-START DOS
*
* ROUTINE TO READ CATALOG
*
SCANDISK LDA #$31             ;INV "1"
         STA $502
```

```
         LDA #0
         STA NUMBER          ;ZERO NUMBER OF
         STA NUMBER+1        ;FILE NAMES
         LDA #NAMEBUFR
         STA BUFFER          ;SET BUFFER POINTERS
         LDA /NAMEBUFR
         STA BUFFER+1
         LDA #$FF
         STA NUMIDS          ;ZERO NUMBER OF DISK IDS
         LDA #$A0            ;BLANK DISK ID BUFFER
         LDY #0
^1       STA IDBUFR,Y
         STA IDBUFR+$0100,Y
         INY
         BNE <1
NEXTDISK JSR HOME
         JSR GETID       ;GET ID FOR DISK
         BCS >1          ;BRANCH IF ID ENTERED
^4       LDA #$B1        ;NORM "1"
         STA $502
         JMP COMMAND
^1       JSR RDVTOC      ;READ VTOC
         CLC             ;SET TO READ FIRST SECTOR
RDSECT   JSR RDSCTR      ;AND READ IT
         BCS NEXTDISK    ;CHECK FOR END OF DIRECTORY
                          SECTORS
         LDX #$00
NXTNAM   STX DIRIDX      ;RESET DIR. INDEX
         LDA DIRBGN,X    ;GET FIRST BYTE OF THIS ENTRY
         BEQ NEXTDISK    ;CHECK FOR END OF DIRECTORY
         BMI NXTENT      ;CHECK FOR DELETED ENTRY
         LDY #$00
         LDA NUMIDS      ;GET DISK ID NUMBER
         STA (BUFFER),Y  ;STORE IT WITH NAME
         INX             ;ADVANCE POINTER TO FILE
                          TYPE
^1       INX             ;MOVE FILE TYPE AND NAME
         INY
         LDA DIRBGN,X
         STA (BUFFER),Y
         CPY #$1F
         BNE <1
         INC NUMBER          ;INCREMENT NUMBER OF
                              ENTRIES
         BNE >2
         INC NUMBER+1
^2       CLC                 ;INCREMENT BUFFER
                              POINTERS
         LDA BUFFER
         ADC #$20
         STA BUFFER
         BCC >1
         INC BUFFER+1
^1       LDA BUFFER          :CHECK FOR OUT OF
                              MEMORY
         CMP HIMEM
         LDA BUFFER+1
         SBC HIMEM+1
         BLT NXTENT
         LDX #OUTBL          ;YES WE ARE
         LDY /OUTBL
         JSR PRINTMES
         JSR RDKEY
         JMP <4
OUTBL    HEX 8787
         ASC "OUT OF MEMORY!"
         HEX 878D00
NXTENT   JSR NXTOME          ;POINT TO NEXT FILE
                              ENTRY
         BCS RDSECT          ;NO MORE SO READ NEXT
                              SECTOR
         BCC NXTNAM          ;MORE IN THIS SECTOR
*
* ROUTINE TO GET DISK ID
*
GETID    LDX #IDTBL          ;REQUEST ID
         LDY /IDTBL
         JSR PRINTMES
         JSR GETLNZ          ;GET ID
         INC NUMIDS          ;INC NUMBER OF IDS
         LDA NUMIDS          ;POINT TO FREE AREA
         JSR POINTID
         LDY #0              ;MOVE DISK ID TO
                              IT'S BUFFER
^2       LDA INBUFF,Y
         CMP #$8D            ;TEST FOR END OF LINE
         BEQ >1
         STA (IDBUFFR),Y
         INY
         CPY #$08            ;TEST FOR END OF ID
         BNE <2
^1       CPY #$01            ;TEST FOR NO ID (RTN
                              FIRST CHAR.)
         RTS
IDTBL    ASC "INSERT DISK, AND ENTER ID"
         HEX 8D
         ASC "JUST PRESS RETURN TO EXIT"
         HEX 8D00
*
* ROUTINE TO SAVE CATALOG TO DISK
*
SAVECAT  LDA #$32            ;INV "2"
         STA $582
         JSR HOME
         JSR GETNAME         ;GET FILENAME
         TXA
         BEQ >2
         LDA #48             ;BSAVE COMMAND
         STA COMND
         LDA #$09            ;SET KEYWORDS FOUND
                              BITS TO
         STA KYWRDFND        ;SHOW 'A' AND 'L'
                              PARAMETERS
         LDX NUMBER          ;SET LENGTH PARAMETER
         LDY NUMBER+1        ;GET LENGTH OF NAME
                              AREA
         JSR MULT
         CLC
         TXA
         ADC #$03            ;ADD LENGTH OF
                              NUMBER OF ENTRIES
         STA LEN             ;AND DISK ID AREA
         TYA
         ADC #$02
         STA LEN+1
         JSR DOCOMND         ;AND DO IT!
^2       LDA #$B2            ;NORM "2"
         STA $582
         JMP COMMAND
*
* ROUTINE TO READ CATALOG FILE
*
READCAT  LDA #$33            ;INV "3"
         STA $602
         JSR HOME
         JSR GETNAME         ;GET FILENAME
         TXA
         BEQ >1
         LDA #0
         STA KYWRDFND
         LDA #50             ; BLOAD COMMAND
         STA COMND
         JSR DOCOMND         ;DO IT
^1       LDA #$B3            ;NORM "3"
         STA $602
         JMP COMMAND
*
* ROUTINE TO HANDLE DISK ERRORS
*
DISKERR  LDX #ERRMESS        ;PRINT MESSAGE
         LDY /ERRMESS
         JSR PRINTMES
         JSR RDKEY           ;WAIT FOR KEYPRESS
```

(Continued on page 55)

PRO-MODEM 1200

It's about time.

Time for your computer to make the telephone connection – with an intelligent, full 212A 300/1200 baud modem – with a real time clock/calendar – and with the capability to expand into a complete telecommunications system. It's time for PRO-MODEM 1200. Much more than just a phone modem.

When you're on-line, time is money. PRO-MODEM telecommunication systems help you save. By monitoring the duration and cost of your phone calls. And by sending and receiving messages, unattended, at preset times when the rates are lower. . . with or without your computer.

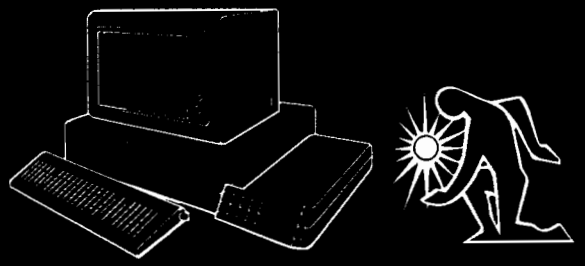
Compare the $495 PRO-MODEM 1200 with any other modem on the market. For example, you'd have to buy both the Hayes Smartmodem 1200 plus their Chronograph for about $950 to get a modem with time base.

PRO-MODEM 1200 is easy to use. A convenient "Help" command displays the Menu of operating command choices for quick reference whenever there's a question about what to do next. Extensive internal and remote self-diagnostics assure that the system is operating properly. Some of the other standard features include Auto Answer, Touch Tone and Pulse Dialing, and Programmable Intelligent Dialing.

PRO-MODEM does more. It lets you build a full telecommunications system with features like Auto Dialer, Incoming and Outgoing Message Buffering, Business/Personal Phone Directory, Programmable Operating Instructions, a 12-Character Alpha-Numeric Time and Message Display, and versatile PRO-COM Software. PRO-MODEM commands are Hayes compatible so you can use most existing telecommunications software without modification.

There's much more to the PRO-MODEM story. See your local dealer for complete details. He'll show you how to save time. And money.

Prometheus Products, Inc., 45277 Fremont Blvd., Fremont CA 94538, (415) 490-2370



PROMETHEUS

COMPUTER MAIL ORDER

TeleVideo

TERMINALS

910	$559.00
912	$689.00
920	$739.00
925	$719.00
950	$929.00
970	$1069.00

COMPUTERS

800A	$1099.00
802	$2699.00
803	$1949.00
802H	$4695.00
806/20	$4999.00
816/40	$9199.00
1602	$3399.00
1603	CALL

MODEMS

HAYES

Smart	$219.00
Smart 1200 (1200 Baud)	$509.00
Chronograph	$199.00
Micromodem 100	$309.00
Micromodem II	$279.00
Micromodem II (with term)	$299.00
Smart Com II	$99.00
Smart 1200B	$469.00

NOVATION

J-Cat	$119.00
Cat	$144.00
D-Cat	$159.00
103 Smart Cat	$189.00
Apple Cat II	$279.00
103/212 Smart Cat	$439.00
212 Apple Cat II	$609.00
Apple Cat II 212 Upgrade	$309.00

ANCHOR

Mark I (RS-232)	$79.00
Mark II (Atari)	$79.00
Mark III (T.I. 99)	$109.00
Mark IV (CBM-PET)	$125.00
Mark V (Osborne)	$95.00
Mark VI (IBM-PC)	$179.00
Mark VII (Auto Ans/Auto Dial)	$119.00
Mark VIII	$269.00
TRS-80 Color Computer	$99.00
9 Volt Power Supply	$9.00

ZENITH

ZT1 Terminal	$369.00

EAGLE

IIE-1	$1369.00
IIE-2	$1649.00
IIE-3	$2399.00
IIE-4	$3199.00
PC-E	$1579.00
PC-1	$2399.00
PC-2	$2799.00
PC-XL	$3599.00
1620	$3599.00
1630	$5499.00
1640	$6499.00
Cyma Software	CALL

MONITORS

AMDEK

300G	$149.00
300A	$159.00
310A	$169.00
Color I	$279.00
Color II	$399.00
Color III	$349.00
Color IV	$999.00

USI

Pi 1, 9" G	$99.00
Pi 2, 12" G	$119.00
Pi 3, 12" A	$149.00
Pi 4, 9" A	$139.00
1400 Color	$299.00

ZENITH

ZVM 122A	$109.00
ZVM 123G	$99.00

BMC

12" Green	$85.00
9191 AU 13" Color	$249.00

TAXAN

12 N Green	$129.00
12 A Amber	$139.00
12 RGB I	$299.00

NEC

JB 1260	$119.00
JB 1201	$149.00
JB 1205	$169.00
JC 1215	$299.00
JC 1216	$429.00
JC 1203	$469.00

GORILLA

12" Amber	$95.00
12" Green	$89.00

IBM

NEC 3550 Printer....$1799

PERCOM/TANDON

DRIVES

5¼" 320K Floppy	$249.00
5 Meg Hard w/Controller	$1399.00
10 Meg Hard w/Controller	$1699.00
15 Meg Hard w/Controller	$2095.00
20 Meg Hard w/Controller	$2399.00

AMDEK

310A Amber Monitor	$169.00
DXY 100 Plotter	$599.00
Color II	$399.00

AST RESEARCH, INC.

Six Pak Plus...from	$279.00
Combo Plus II...from	$279.00
Mega Plus...from	$309.00
I/O Plus II...from	$139.00

QUADRAM

Quadlink	$549.00
Quadboard...as low as	$309.00
Quad 512 Plus...as low as	$259.00
Quadcolor...as low as	$219.00
Chronograph	$89.00
Parallel Interface Board	$89.00
64K RAM Chips Kit	$69.00

MICRO PRO

Word Star/Mail Merge	$369.00
InfoStar	$299.00
Spell Star	$159.00
CalIStar	$159.00

MICROSTUF

Crosstalk	$129.00

MICROSOFT

Multiplan	$179.00

ASHTON-TATE

0-Base II	$419.00

IUS

EasyWriter II	$219.00
EasySpeller	$119.00
EasyFiler	$239.00

CONTINENTAL SOFTWARE

1st Class Mail/Form Letter	$79.00
The Home Accountant Plus	$99.00

SYNAPSE

File Manager	$89.00

LOTUS

123	$329.00

PFS

	APPLE	IBM
File	$79.00	$89.00
Report	$79.00	$79.00
Graph	$79.00	$89.00
Write	n/a	$89.00

KRAFT

IBM Joystick	$55.00
IBM Paddles	$39.00

PROFESSIONAL SOFTWARE

PC Plus Word Processing	$299.00

KOALA

Koala Pad	
Apple	$85.00
IBM	$95.00
Atari	$75.00
CBM 64	$75.00

PAPER SUPPLIES

1or2" Address Labels(Tract.Feed)	$9.95
15" Report Paper(Tract Feed)	$24.95
8½" Blnk Wht Paper(Tract.Feed)	$19.95

SANYO

MBC-550PC	CALL
MBC-555PC	CALL
MBC 1100	$1599.00
MBC 1150	$2099.00
MBC 1200	$1999.00
MBC 1250	$2399.00
FDD 3200-320K Drive	$399.00
FDD 6400-64K Drive	$499.00
PR 5500 Printer	$699.00

PRINTERS

EPSON

MX80 FT, MX100, RX80, FX80, FX100	CALL

OKIDATA

82, 83, 84, 92, 93	CALL

STAR

Delta 10	$559.00
Gemini 10X	$299.00
Gemini P15	$449.00
Serial Board	$75.00

SMITH CORONA

TP-2	$469.00
Tractor Feed	$119.00

C.ITOH

Gorilla	$209.00
Prowriter 8510P	$379.00
Prowriter 1550P	$689.00
Starwriter F10-40P	$1149.00
Printmaster F10-55P	$1549.00
Tractor Feed	$199.00

DAISYWRITER

2000 Letter Quality	$1049.00
2500...NEW	CALL
Tractor Feed	$109.00

DIABLO

620	$949.00
630	$1749.00

IDS

Call for ALL Configurations on IDS PRISM PRINTERS.

NEC

8023 AN	$399.00
8025	$729.00
3510	$1449.00
3530	$1499.00
3550	$1799.00
7710/7730	$1949.00

BMC

401 Letter Quality	$699.00
BX-80 Dot Matrix	$259.00

COMREX

Comwriter II Parallel Printer	$549.00

MANNSEMAN TALLY

Spirit 80	$319.00
MT-160L	$589.00
MT-180L	$829.00
TRANSTAR	CALL
TOSHIBA	CALL

CABLES & CONNECTIONS

Atari to Parallel	$29.00
Atari to Serial	$29.00
Apple to Parallel	$69.00
Apple to Parallel/Graphics	$99.00
Apple to Serial	$89.00
IBM to Parallel	$35.00
IBM to Serial	$29.00
Parallel to Parallel	$29.00
Serial to Serial	$29.00
Grappler Plus	$129.00
PKASO	$139.00
Atari to Modem Cable	$29.00
CBM 64 to IEEE Board	$79.00
Apple 80-Column Card	$159.00
CBM Pet to Parallel	$99.00
CBM Pet to Serial	$89.00

HEWLETT PACKARD

HP 75	$749.00
HP 41C	$149.00
HP 41CV	$209.00
HP 41CX	$259.00
HP 10C	$52.00
HP 11C	$69.00
HP 12C	$92.00
HP 15C	$92.00
HP 16C	$92.00

For HP 41/41CV

HPIL Module	$99.00
HPIL Cassette or Printer	$359.00
Card Reader	$144.00
Extended Functions Module	$64.00
Time Module	$64.00

SHARP

PC-1500A $169

PC-1250A $89

POCKET COMPUTERS

CE-125 Printer/Cass. Int.	$129.00
CE-150 Color Print./Cass. Int.	$172.00
CE-155 8K RAM	$94.00
CE-161 16K RAM	$135.00
CE-500 ROM Library	each $39.00

Texas Instruments

TI-40.....$209

CALL FOR PRICING ON TIMEX SINCLAIR 1000

16K Memory	$44.95
2040 Printer	$99.95
Vu-Calc	$17.95
Mindware Printer	$99.00

=WEST=

1-800-648-3311

In NV call (702)588-5654, Dept. 1215
P.O. Box 6689, Stateline, NV 89449
Order Status #: 588-5654

=CANADA=

1-800-268-4559

In Toronto call (416)828-0866, Dept. 1215
2505 Dunwin Ct., Unit 1B,
Mississauga, Ontario, Canada LSLITI
Order Status #: 828-0866

=EAST=

1-800-233-8950

In PA call (717)327-9575, Dept. 1215
477 E. Third St. Williamsport, PA 17701
Order Status #: 327-9576

MasterCard

No risk, no deposit on C.O.D. orders. Pre-paid orders receive free shipping within the UPS Continental United States with no waiting period for certified checks or money orders. Add 3% (minimum $5.00) shipping and handling on all C.O.D. and credit card orders. Larger shipments may require additional charges. NV and PA residents add sales tax. All items subject to availability and price change. We stock manufacturer's and third party software for most all computers on the market. Call today for our new catalog.

COMPUTER MAIL ORDER

FRANKLIN

COLOR COMPUTERS
ACE 1000, ACE 1200, ACE 1200 OMS, ACE 1100 PRO PACK: ACE 1000, ACE 10 Drive & Controller, 80 Column Card, ACECalc & ACEWriter.....CALL

APPLE IIe STARTER PACK
64K Apple IIe, Disk Drive and Controller, 80 Column Card & Monitor III...
COMPLETE.....$1599.00

MICRO-SCI
Apple & Franklin

A2	$219.00
A40	$299.00
A70	$319.00
C2 Controller	$79.00
C47 Controller	$89.00

RANA

Elite I (Apple/Franklin)	$279.00
Elite II (Apple/Franklin)	$319.00
Elite III (Apple/Franklin)	$569.00

VISICORP
FOR APPLE, IBM & FRANKLIN

Visidex	$189.00
Visifile	$189.00
Visiplot	$159.00
Visiterm	$89.00
Visitrend/Plot	$229.00
VisiSchedule	$229.00
Desktop Plan	$189.00
Visicalc 4 (IBM)	$169.00
Visicalc Advanced IIe	$309.00
Stretch Calc	$89.00

Visicorp prices for IBM may vary slightly.

LJK

Letter Perfect Apple	$109.00
Data Perfect Apple	$75.00

AXLON

Apple/Franklin 128K Ram	$299.00
Apple/Franklin Ram Disk	$729.00

MPC

Bubdisk (128K Non Volitare)	$649.00

WICO

Joystick	$21.95
Famous Red Ball	$23.95
Power Grip	$21.95
Three-Way Deluxe	$22.95
Atari/VIC Trackball	$34.95
Apple Trackball	$59.95

KRAFT

Atari Single Fire	$13.00
Atari Switch Hitter	$16.00
Apple Joystick	$44.00
Apple Paddles	$39.00

commodore

CBM 8032 $599

CBM64 ...$219

VIC 20 .....$89

CALL ON
Executive 64 Portable

1520 Color Printer/Plotter	$169.00
1525 80 Column Printer	$219.00
1530 Datasette	$69.00
1541 Single Disk Drive	$249.00
1600 VIC Modem	$59.00
1650 AD/AA Modem	$89.00
1702 14" Color Monitor	$249.00
Pet 64	$569.00
Pet 4032	$599.00
CBM 8032	$599.00
Super Pet	$999.00
B128-80	$769.00
2031	$299.00
8050	$949.00
8250	$1199.00
9060	$1999.00
9090	$2199.00
4023	$379.00
6400	$1399.00
64K Upgrade	$269.00
Spell Master	$149.00
Z-Ram	$549.00
Silicon Office	$749.00
The Manager	$209.00
Soft Rom	$129.00
Jinsam	CALL
CalcResult 64	$139.00
CodeWriter 64	$75.00
VisiCalc	$169.00

PROFESSIONAL SOFTWARE

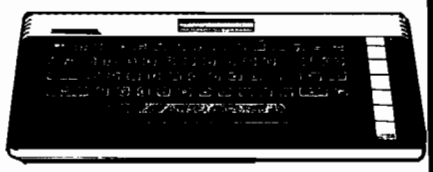
Word Pro 2 Plus	$159.00
Word Pro 3 Plus	$189.00
Word Pro 4 Plus	$279.00
Word Pro 5 Plus	$279.00
InfoPro	$179.00
Administrator	$399.00
Power	$79.00
Word Pro 64 Plus	$65.00

CARDCO
for VIC 20/64

Light Pen	$32.00
Cassette Interface	$29.00
Parallel Printer Interface	$69.00
3 Slot Expans. Interface (20)	$32.00
6 Slot Expans. Interface (20)	$79.00

HOME COMPUTERS

ATARI



ATARI 600XL.....$149

ATARI 800XL.....$269

ATARI 1200XL .....CALL

ATARI 400 .....CALL

ATARI 800 .....CALL

1010 Program Recorder	$74.00
1020 40 Col. Printer/Plotter	$249.00
1025 80 Col. Printer	$449.00
1027 Letter Quality Printer	$299.00
1050 Disk Drive	$379.00
1030 Direct Connect Modem	CALL
CX30 Paddles	$12.00
CX40 Joystick	$8.00
CX42 Remote Joystick	CALL
CX77 Touch Tablet	$69.00
CX80 Trak Ball	$49.00
CX85 Keypad	$105.00
CX418 Home Manager	$69.00
CX488 Communicator II	$229.00
KX7098 Atari Accountant	$209.00
KX7101 Entertainer	$69.00
KX7102 Arcade Champ	$75.00

ALIEN

Atari Voice Box	$119.00
Apple Voice Box	$149.00

MEMORY BOARDS

Axlon 32K Ram	$59.00
Axlon 48K Ram	$99.00
Axlon 128K Ram	$299.00
Intec 32K Board	$59.00
Intec 48K Board	$85.00
Intec 64K Board (400 only)	$99.00
Intec Real Time Clock	$39.00

DISK & CART HOLDERS

Flip-n-File 10	$4.00
Flip-n-File 50	$19.00
Flip-n-File 400/800 Cart	$19.00

PERCOM

AT 88-S1	$329.00
AT 88-A2	$269.00
AT 88-S2	$569.00
AT 88-S1 PD	$469.00
AT 88-DDA	$145.00
RFD 40-S1	$449.00
RFD 40-A1	$279.00
RFD 40-S2	$729.00
RFD 44-S1	$539.00
RFD 44-S2	$889.00
TX 99-S1 (Texas Inst.)	$279.00

RANA

1000 Atari Disk Drive	$319.00

TRAX

AT-D2	CALL

FLOPPY DISKS
MAXELL

MD-1	$29.00
MD-2	$39.00
FD-1 (8")	$40.00
FD-2 (8" DS DD)	$50.00

ELEPHANT

5¼" SS SD	$18.50
5¼" SS DD	$24.95
5¼" DS DD	$29.95

VERBATUM

5¼" SS DD	$26.00
5¼" DS DD	$36.00

HEAD

Disk Head Cleaner	$14.95

C.M.O. TOP 80

APPLE/FRANKLIN

1. Choplifter	$27.00
2. Bank Street Writer	$55.00
3. PFS: File	$89.00
4. Visicalc	$179.00
5. Home Accountant	$55.00
6. Zaxxon	$29.00
7. Most Amazing Thing	$28.00
8. Visifile	$189.00
9. Fathoms 40	$19.00
10. Deadline	$35.00
11. PFS: Report	$89.00
12. Zork III	$29.00
13. Frogger	$24.00
14. Facemaker	$24.00
15. Snooper Troops #1	$32.00
16. Delta Drawing	$35.00
17. Castle Wolfenstine	$24.00
18. Wayout	$29.00
19. Canyon Climber	$19.00
20. Bandits	$26.00

CBM 64

1. Word Pro 64	$65.00
2. Jumpman	$29.00
3. Gorf (20/64)	$14.95
4. Microspec Data Base 64	$69.00
5. Logo 64	$39.00
6. Microspec Gen. Ledger 64	$79.00
7. Zork III	$29.00
8. Frogger (64)	$23.00
9. Quick Brown Fox (20/64)	$49.00
10. Shamus	$29.00
11. Deadline	$29.00
12. Assembler 64	$14.95
13. Zork II	$29.00
14. 3-D Man	$14.00
15. Protector	$32.00
16. Starcross	$29.00
17. Easy Mail 64	$14.95
18. Grave Robber	$11.00
19. Wall Street	$19.00
20. Trash Man	$32.00

ATARI

1. Donkey Kong	$39.00
2. Zaxxon	$29.00
3. E.T. Phone Home	$39.00
4. Miner 2049er	$35.00
5. Dig Dug	$33.00
6. Choplifter	$29.00
7. Donkey Kong, Jr	$39.00
8. Canyon Climber	$25.00
9. Snooper Troops #2	$34.00
10. Word Wizard	$59.00
11. Picnic Paranoia	$34.00
12. Jumpman	$29.00
13. Shamus	$34.00
14. Letter Perfect	$109.00
15. File Manager 800	$69.00
16. Preppie	$24.00
17. Astro Chase	$25.00
18. Blade/Black Hole	$27.00
19. Pac Man	$33.00
20. Baja Buggies	$25.00
21. Crush, Crumble & Chomp	$24.00
22. Wayout	$27.00
23. Zork II	$29.00
24. Visicalc	$159.00
25. Atari Writer	$49.00
26. Three Little Pigs	$25.00
27. Upper Reaches of Apshai	$16.00
28. Starbowl Football	$24.95
29. Drelbs	$26.00
30. Protector	$34.00
31. Frogger	$25.00
32. Lunar Leeper	$24.00
33. Wizard of Wor	$34.00
34. Kindercomp	$21.00
35. Moon Shuttle	$22.00
36. Home Accountant	$55.00
37. Temple of Apshai	$29.00
38. Spell Wizard	$39.00
39. Nautilus	$26.00
40. O'Riley's Mine	$22.00

WEST
1-800-648-3311
In NV call (702)588-5654, Dept. 1215
P.O. Box 6689, Stateline, NV 89449
Order Status #: 588-5654

CANADA
1-800-268-4559
In Toronto call (416)828-0866, Dept. 1215
2505 Dunwin Ct., Unit 1B,
Mississauga, Ontario, Canada LSLITI
Order Status #: 828-0866

EAST
1-800-233-8950
In PA call (717)327-9575, Dept. 1215
477 E. Third St. Williamsport, PA 17701
Order Status #: 327-9576

VISA

CANADIAN ORDERS: All prices are subject to shipping, tax and currency exchange fluctuations. Call for exact pricing in Canada.
INTERNATIONAL ORDERS: All shipments outside the Continental United States must be pre-paid by certified check only. Include 3% (minimum $5.00) shipping and handling. **EDUCATIONAL DISCOUNTS:** Additional discounts are available to qualified Educational Institutions.
APO & FPO: Add 3% (minimum $5.00) shipping and handling.

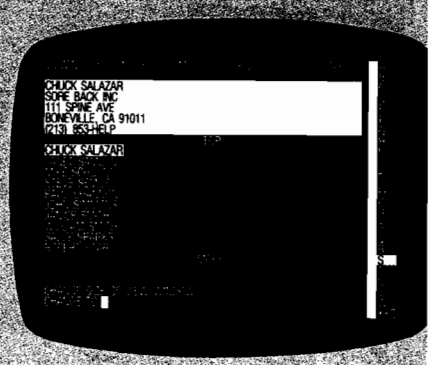
MAGIC MEMORY

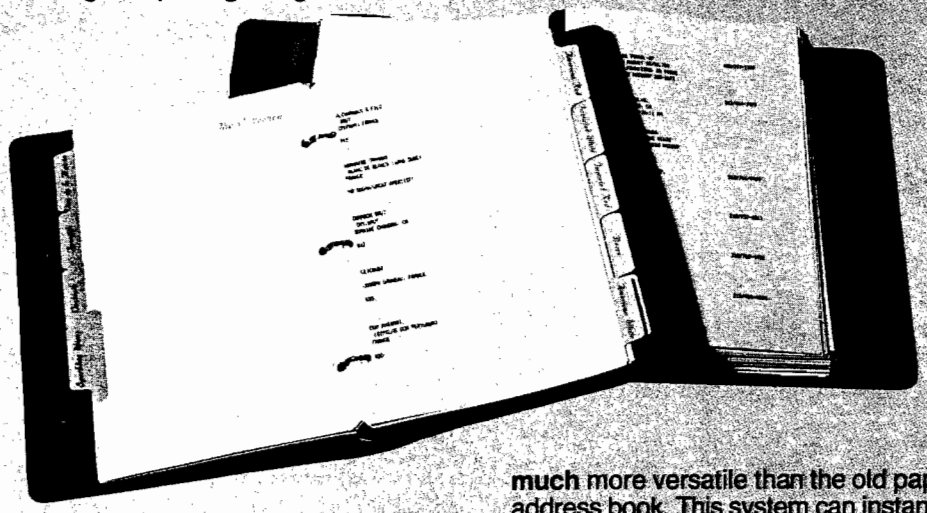
Will Remember Everything For You

(If you want it to)

Imagine a system that would record all the wonderous, valuable information you have assimilated onto a single tiny disk. (No more scattered bits of paper, business cards, etc.) Imagine the same system giving you a typed sheet you could put into a notebook or print out for a party and instantly change, or add to, at a moments notice. Imagine cross-referencing to suit both your business needs and personal desires so that all your data was organized into one little black book! On top of all this — imagine having **fun** putting it together.

MAGIC MEMORY™ is built for the computer rookie. Everyone can relate to **MAGIC MEMORY™** because its form is familiar. It looks like an address book but its not. Its *more*. Like the address book, **MAGIC MEMORY™** presents an A thru Z index tabulation on the right edge of the video display. The user simply selects a tab and the *book* is opened to the proper page(s). A second set of tabs are available that can be labeled by the user (i.e.: companies one deals with, birthdays, lists, wines, shops, etc.) Yet **MAGIC MEMORY™** is

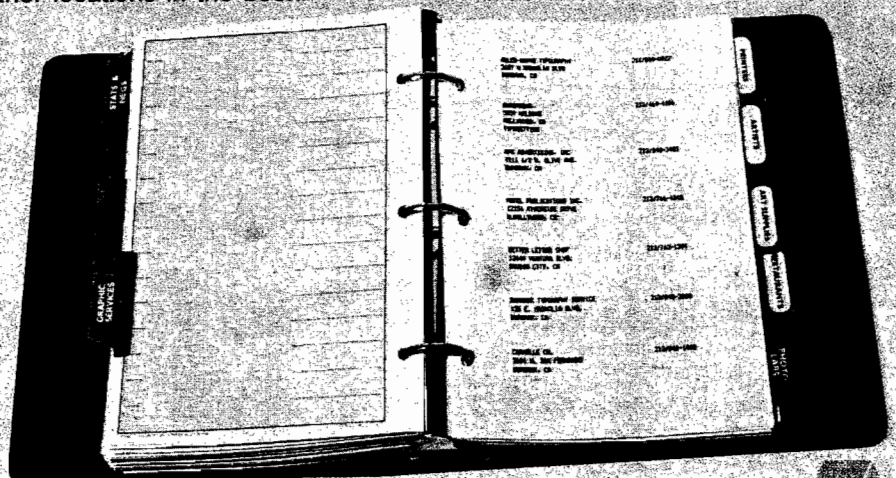




"Imagine" no more. ARTSCI, INC. proudly announces **MAGIC MEMORY™**, the complete advanced database system. **MAGIC MEMORY** is made to simplify the information storage process.

The typical first-time computer user has no *human* experience that will help him to relate to a *computer's* methods of handling information. Therefore, his learning ability is hampered and remains so for many hours of use, UNLESS the computer has been designed to run simply.

much more versatile than the old paper address book. This system can instantly add or delete information, sort alphabetically, and transfer data to other locations in the book.



One of the features in the print section of **MAGIC MEMORY™** is the ability to print out any or all of the *address* book,

or any of its special sections.

MAGIC MEMORY'S data presentatio screen will instantly adapt to any typ of information you may want to store It is only limited to your imagination an your needs.

MAGIC MEMORY™ is designed t operate on an APPLE //e and still re main totally compatible with APPLE I The system will operate in 40 column or 80 columns. You may also use th 70-column display that requires no ad ditional hardware.

MAGIC MEMORY™ retails for $99.95 Call 213/985-5763. Use your imagina tion to outwit your organizational woes

MAGIC MEMORY™ is a trademark of ARTSCI, INC.

213/985-5763
5547 Satsuma Avenue
North Hollywood, Ca 91601

artsci

```
        LDA $602            ;CHECK TO SEE IF WE
                             WERE
        CMP #$33            ;READIING OR WRITING
        BEQ <1              ;AND
        BNE <2              ;RETURN
ERRMESS ASC "DISK ERROR, PRESS ANY KEY TO
             CONTINUE"
        HEX 87878D00
*
* GET DISK AND FILENAME
*
GETNAME LDX #GETMESS
        LDY /GETMESS
        JSR PRINTMES
^2      JSR GETLNZ          ;GET FILENAME
        TXA
        BEQ >3
        JSR BLANKNAM        ;BLANK FILENAME
                             BUFFER
        LDY #$00            ;MOVE FILENAME TO
                             DOS NAME BUFFER
^1      LDA INBUFF,Y
        CMP #$8D
        BEQ SETPARMS
        STA DOSNAME,Y
        INY
        BNE <1
SETPARMS LDA #0             ;SET PARAMETERS
        STA VOL
        LDA #6
        STA SLOT
        LDA #1
        STA DRIVE
        LDA #NUMBER         ;SET STARTING ADDRESS
        STA ADDR
        LDA /NUMBER
        STA ADDR+1
^3      RTS
GETMESS ASC "INSERT DISK, AND ENTER FILENAME"
        HEX 8D8D00
*
*
* THIS IS WHERE THE SORT ROUTINE
* GOES. WILL BE IN NEXT MONTH
SORTCAT EQU *
        RTS
*
*
MULT    STY $65             ;MULTIPLY ROUTINE
        STX $64             ;$64,$65 * $AD,$AE
        LDA #$20
        STA $AD
        LDA #$00
        STA $AE
        JMP ROMULT
*
*
* THIS IS WHERE THE PRINTCAT ROUTINE
* GOES. ALSO WILL BE IN NEXT MONTH.
PRINTCAT EQU *
        RTS
*
*
* TEXT PRINTER
*
PRINTMES STX PRNTBUF
        STY PRNTBUF+1
        LDY #$00
^1      LDA (PRNTBUF),Y
        BEQ >2
        JSR COUT
        INY
        BNE <1
^2      RTS
;
POINTID STA IDBUFFR         ;GIVEN A AS AN INDEX
        LDA #$00            ;TO THE ID TABLE
        STA IDBUFFR+1       ;POINT THE THE
                             CORECT PLACE
        LDX #$00            ;BY MULTIPLYING BY 8
^1      ASL IDBUFFR
        ROL IDBUFFR+1
        INX
        CPX #$03
        BNE <1
        CLC
        LDA IDBUFFR
        ADC #IDBUFF
        STA IDBUFFR
        LDA IDBUFFR+1
        ADC /IDBUFR
        STA IDBUFFR+1
        RTS
;
MENU    JSR HOME            ;PRINT THE MENU BOX
        LDA #1
        JSR MAXFILES        ;SET MAXFILES = 1
        LDA #$60
        STA $A851           ;PATCH DOS
        STA $9EE0           ;  "   "
        LDA #$4C
        STA $A6EF
        LDA #DISKERR
        STA $A6F0
        LDA /DISKERR
        STA $A6F1
        JSR SETINO
        JSR SETOUT0
        LDX #MENUTEXT
        LDY /MENUTEXT
        JSR PRINTMES
BOX     LDA #$00            ;FRAME THE MENU
        STA CV
        JSR VTAB
        LDY #0
        STY CH
        LDA #"#"
        LDX #38
^1      JSR COUT
        DEX
        BPL <1
        LDX #08
^1      LDY #39
        STY CH
        JSR COUT
        JSR COUT
        DEX
        BPL <1
        LDX #38
^1      JSR COUT
        DEX
        BPL <1
        LDA #$0C            ;SET THE TEXT WINDOW
                             DOWN TO
        STA WINTOP          ;PROTECT THE MENU BOX
        RTS
MENUTEXT HEX 8D8D
        ASC "  1) CATALOG DISKS"
        HEX 8D
        ASC "  2) SAVE CATALOG     "
        ASC "ENTER THE NUMBER"
        HEX 8D
        ASC "  3) READ CATALOG"
        HEX 8D
        ASC "  4) SORT CATALOG     "
        ASC "OF YOUR CHOICE."
        HEX 8D
        ASC "  5) PRINT CATALOG"
        HEX 8D
        ASC "  6) END"
        HEX 8D00
FLDLEN  HEX 08              ;DISK ID LENGTH
        HEX 01              ;FILE TYPE LENGTH
        HEX 1E              ;FILE NAME LENGTH
        END
```

MICRO™

Apple Slices

by Phil Daley

Disk Dump Program

Here is a program to dump any disk file to screen, printer or whatever. The program asks for what type of disk you have, and then asks for the filename. If you don't know the name, a RETURN will present the possibilities. The dump can either be HEX or straight ASCII. If the file is not straight text, your printer will probably go through a few contortions in the ASCII mode. The program currently has subroutines for DOS and Pascal. I will add subroutines for CP/M, Flex and OS-9 soon.

Listing 1

```
5  DIM A(150),B(150):BU = 16384: GOSUB 9000
10  TEXT : HOME : VTAB 7: PRINT "DISK TYPE":
     PRINT "1. DOS 3.3"A: PRINT "2. PASCAL"
    A: PRINT "3. CPM".: PRINT "4. FLEX".: PRINT
    "5. OS-9"P: PRINT : PRINT "CHOOSE: ";: GET
    A$:A =  VAL (A$): IF A < 1 OR A > 5 THEN
    10
81  PRINT : PRINT "SLOT FOR OUTPUT? ";: GET
    A$:SL =  VAL (A$): IF SL < 0 OR SL > 7 THEN
    81
85  PRINT : PRINT "HEX OR ASCII?";: GET HE$:
    HE = 0: IF HE$ = "H" THEN HE = 1
90  TEXT : HOME : PRINT "INSERT DISK": PRINT
    : PRINT "FILENAME?": PRINT : PRINT "(RE
    TURN) FOR CATALOG ": INPUT B$:A$ = "":C
    F = 0: IF  LEN (B$) = 0 THEN CF = 1: GOTO
    110
105 A$ = B$
110  HOME : VTAB 7: PRINT "READING CATALOG..
    ..": ON A GOSUB 1000,2000,3000,4000,500
    0: INPUT "PRESS (RETURN)";A$: GOTO 10: END

900 HI =  INT (H / 16):LO = H - HI * 16:D =
    HI: GOSUB 950:A$ = H$:D = LO: GOSUB 950
    :A$ = A$ + H$ + " ": RETURN
950  IF D < 10 THEN H$ =  STR$ (D): RETURN
960 H$ =  CHR$ (D + 55): RETURN
1000  FOR I = 1 TO  LEN (B$):A$ = A$ +  CHR$
    ( ASC ( MID$ (B$,I,1)) + 128): NEXT : POKE
    780,17: POKE 785,64: FOR C = 15 TO 2 STEP
     - 1: POKE 781,C: CALL 768: FOR Y = 0 TO
    6:E$ = "":EN = 16395 + Y * 35:Q =  PEEK
    (EN): IF Q = 0 THEN C = 2:Y = 6: GOTO 1
    070
1030  FOR X = EN + 3 TO EN + 32:E$ = E$ +  CHR$
    ( PEEK (X)): NEXT : IF CF THEN  PRINT E
    $: GOTO 1070
1065  IF A$ =  LEFT$ (E$, LEN (A$)) THEN 110
    0
1070  NEXT : NEXT
1085  IF CF THEN  RETURN
1090  FLASH : PRINT : PRINT "NOT FOUND": NORMAL
    : RETURN
1100  POKE 780, PEEK (EN): POKE 781, PEEK (E
    N + 1): CALL 768:J = 0: FOR I = 0 TO 12
    1
1135  IF I > 121 THEN 1170
1140 A = I * 2 + 12 + BU: IF  PEEK (A) = 0 THEN
    I = I + 1: GOTO 1135
1155 J = J + 1:A(J) =  PEEK (A):B(J) =  PEEK
    (A + 1)
1170  NEXT :U = J: PRINT : PRINT  CHR$ (4)"P
    R#"SL: FOR J = 1 TO U: POKE 780,A(J): POKE
    781,B(J): CALL 768: FOR I = 0 TO 255:A =
    BU + I: IF HE THEN H =  PEEK (A): GOSUB
    900: PRINT A$;: GOTO 1230
1220  PRINT  CHR$ ( PEEK (A));
1230  NEXT : NEXT : PRINT : PRINT  CHR$ (4)"
    PR#0": RETURN
2000  POKE 780,0: FOR C = 11 TO 4 STEP  - 2:
     POKE 781,C: POKE 785,64: CALL 768: POKE
    781,C - 1: POKE 785,65: CALL 768:EN = B
    U + 26
2030 SB =  PEEK (EN) +  PEEK (EN + 1) * 256:
    EB =  PEEK (EN + 2) +  PEEK (EN + 3) *
    256:LG =  PEEK (EN + 6):E$ = "": IF LG =
    0 THEN 2100
2060  FOR I = EN + 7 TO EN + 7 + LG - 1:E$ =
    E$ +  CHR$ ( PEEK (I)): NEXT : IF CF THEN
     PRINT E$: GOTO 2100
2090  IF A$ = E$ THEN 2200
2100 EN = EN + 26: IF EN < BU + 512 - 26 THEN
    2030
2110  NEXT : GOTO 1085
2200  PRINT : PRINT  CHR$ (4)"PR#"SL:T =  INT
    (SB / 8):S = SB - T * 8:T1 =  INT (EB /
    8):S1 = EB - T1 * 8:S2 = S1:T1 = T1 - 1
    : FOR I = T TO T1:S3 = S1: IF T1 > I THEN
    S3 = 7
2250  FOR J = S2 TO S3: POKE 785,64: POKE 78
    0,I: POKE 781,TS(0,J): CALL 768: GOSUB
    2470: POKE 780,I: POKE 781,TS(1,J): CALL
    768: GOSUB 2470: NEXT :S2 = 0: NEXT : PRINT
    : PRINT  CHR$ (4)"PR#0": RETURN
2470  FOR K = 0 TO 255:A = BU + K: IF HE THEN
    H =  PEEK (A): GOSUB 900: PRINT A$;: GOTO
    2490
2485  PRINT  CHR$ ( PEEK (A));
2490  NEXT : RETURN
9000  FOR I = 768 TO 805
9010  READ A: POKE I,A: NEXT
9011  FOR I = 0 TO 7: FOR J = 0 TO 1: READ T
    S(J,I): NEXT J,I
9015  RETURN
9020  DATA  169,3,160,8,32,217,3,96,1,96
9030  DATA  1,0,17,15,30,3,0,64,0,0
9040  DATA  1,0,254,96,1,0,0,0,0,0
9050  DATA  0,1,239,216,0,0,0,0
9060  DATA 0,14,13,12,11,10,9,8,7,6,5,4,3,2,
    1,15
```

MICRO

MICRO™ Apple Reviews

Product Name: **Ultra ROM Board/Editor**
Equip. req'd: Apple II
Price: $190.00
Manufacturer: Hollywood Hardware
6842 Valjean Ave.
Van Nuys, CA 91406

Description: A plug-in ROM board with Neil Konzen's GPLE included with 25 ampersand utilities for an on-line editor/utility package. The Global Program Line Editor is a handy set of line editing commands and is available at any time, even with a program already loaded. The utilities include switching in other "&" commands, BLOAD information, control character display, free sectors, line finder, HIMEM and LOMEM settings, graphics screen commands without clearing screen, IF,THEN,ELSE structures, program restore (not new), PRINT USING, memory search, clear end-of-line and -page, help and macro definitions, for single key entry.

Pluses: The program is always waiting to be called. If you forget to load a line editor while working on a program, then you have to save, load the editor and reload the program. With Ultra ROM, a PR# <slot > command will activate the editor, program intact. If you program a lot and haven't used a line editor, get one right away.

Minuses: The "&" additions will only run on a similar system. (A new runtime package is being included for transportability.)

Documentation: A 50-page manual clearly explains how the programs work and how to manage your own vectors.

Skill level required: Some programming experience is necessary for full use.

Reviewer: Phil Daley

Product Name: **Robographics CAD-1**
Equip. req'd: Apple II
Price: $1095.00
Manufacturer: ROBO Graphics
125 Pheasant Run, Suite 2B
Newton, PA 18940

Description: An extremely sophisticated computer-aided graphics and drafting package for the Apple which has functions, speed and accuracy previously available only on expensive CAD systems. The basic system includes 4 disks, manual, interface module (a copy protection device) and a precision controller much more accurate and versatile than a joystick. It has such features as zoom, pan, angle locks, grid locks, scale drawing, move, find, exchange, line color and type, text entry and more. Pictures can be stored on a library disk with unique picture labeling and retrieval system. Optional equipment includes dot-matrix printer, plotter, color printer, and digitizer.

Pluses: The system is menu driven and easy to become aquainted with. Scale drawing is accurate and easy to do. Zoom works at many levels of nesting, (greater than 1 part in a billion) giving effectively unlimited screen resolution. Picture complexity is only limited by space on disk. This system has to be seen in operation to appreciate its power: especially its ability to produce highly detailed technical drawings.

Minuses: On complex pictures this system can be slow. Redrawing a picture on screen can take several minutes.

Documentation: An easy to read and well indexed manual answers all questions on operation.

Skill level required: Some drafting experience will help get the full benefit of all the sophisticated features.

Reviewer: Phil Daley

Product Name: **Cdex Training for VisiCalc**
Equip. req'd: Apple II+
Price: $49.95
Manufacturer: Cdex Corporation
5050 El Camino Real, Suite 200
Los Altos, CA 94022
Developer: Dr. Steven C. Brandt

Description: A real bargain. A program to teach you how to use VisiCalc and to use as reference. 2 disks lead step by step in major concepts and commands of VisiCalc; have review questions, hints, positive reinforcements. 3rd disk is quick reference of commands. Manual supplements material with exercises and reference.

Pluses: Very interactive; easy to use. A professional, top-quality package.

Minuses: Disk lessons do not cover all commands, such as window and title commands, but are covered in the reference disk.

Documentation: Well-written, indexed manual contains command reference, examples and exercises.

Skill level required: Anyone interested in learning about VisiCalc. Very little computer experience needed.

Reviewer: Mary Gasiorowski

(Continued on next page)

Reviews *(continued)*

Product Name: **KoalaPad Touch Tablet**
Equip. req'd: Apple II
Price: $129.95
Manufacturer: Koala Technologies
253 Martens Ave.
Mt. View, CA 94040

Description: A graphics tablet operating from the game controller port with extreme smoothness and precision. The 4 x 4 inch active surface can be activated with finger or stylus. It includes two controller buttons.

Pluses: This product is a great refinement over a joystick. It is much easier and more natural to control than paddles or conventional joysticks. I immediately improved my previous high scores on every game I tried it on. Programming is identical to paddle programming.

Minuses: The *KoalaPad Touch Tablet* does not have self-centering such as a joystick has, and removing your finger from the tablet may result in untimely moves during the progress of a game.

Documentation: A very complete, clear and well written booklet is included with the tablet.

Skill level required: No prior skill needed.

Reviewed: Phil Daley

Product Name: **Personal Finance Manager**
Equip. req'd: Apple II+, or Apple II with Applesoft Firmware Card or Language System; 48K RAM; and one disk drive (DOS 3.3).
Price: $75.00
Manufacturer: Apple Computer Inc.
10260 Badley Drive
Cuppertino, CA 95014

Description: A financial program that allows you to budget twenty-four separate accounts which you define. Also available are credit card accounts, and checkbook reconciliation. Defaults make date input and editing a snap. PFM prints out any display you wish and will also move each year's records to another disk for long-term storage.

Pluses: Ample room for the average person, PFM has superb error-handling checks and messages to guide you along. The monthly/yearly updates are helpful and are backed up with a bar graph.

Minuses: You can't track income which would give you a better income vs. expenses picture. Having to continually load modules from disk slows PFM's speed. Not being able to make financial projections will annoy some of you.

Documentation: An attractive and concise booklet is provided with the master disk and backup.

Skill level required: Any person able to turn the computer on and follow directions.

Reviewer: Mike Cherry

MICRO

Hollywood Hardware™

SOFTWARE ENHANCEMENT SYSTEM, APB-102 ..... $189.00
W/GPLE, APU-1, FMP, DISK, MANUAL, QUICK REFERENCE GUIDE
APU-2, UTILITY ROM #2 ..... $35.00
W/RENUMBER, MERGE, HOLD, ETC.
ROM DEVELOPMENT PKG. $99.00
W/DISK, INSRUCTIONS & EMULATION ROM
A/D, 12 BIT, 16 CHANNEL $299.00
W/5 VOLTAGE RANGES, 25 μ SEC CONVERSION
PRO-1, XTRA-LARGE PROTOTYPE BOARD $29.95
UP TO 52 IC's, NUMBERED & LETTERED PINS, HANDY POWER AND GROUND CONNECTIONS, NUMBERED I/O CONNECTIONS
48 LINE PARALLEL I/O BOARD, CPU-1 ..... $249.00
25 BUFFERED LINES IN W/FILTERING, 23 BUFFERED LINES OUT, INTERRUPT INPUT

#1 SELLER AT A.P.P.L.E.*

Powerful New Software Enhancement System For Apple II, //e. Triples Programming Speed!

APB-102 Ultra-Rom Board/Editor Includes:

- **Advanced G.P.L.E.* (Global Program Line Editor) in Firmware**
  —With Insert, Delete, Find, Tab, Zap, Pack, Restore, End, Etc.
- **Firmware Management Program (FMP) Overlays 32K of ROM in 2K Space**
  —Allows jumps and calls between banks—searches for utilities by name
  —Recognizes new ROMs and utilities automatically
- **APU-1 with over 25 Language Extensions & Ampersand Utilities**
  —If/Then/Else, Print Using, Ultra Fast Search, Damaged Program Recovery
- **Always in the Machine—No Searching for a Disk**
- **Never in the way—No Program RAM used**
  —Connect with 4 Keystrokes / Disconnect with 2!

"If you program & haven't used a line editor, get one right away" —MICRO MAGAZINE
"The most powerful program development tool I have" —ROBERT WILSON, PROGRAMMER
"Excellent Product, flawless" —PHILIP DALEY, PROGRAMMER
"Great product, exceeds my expectations" —DR. STEVE COOK
"An elegant solution... well thought out... worth it" —SOFTALK REVIEW, SEPT. 1983
"The best thing for the Apple since the disk" —EDWARD DECKER, PHARMACIST/PROGRAMMER

Hollywood Hardware (213) 989-1204
6842 VALJEAN AVENUE, VAN NUYS, CA 91406

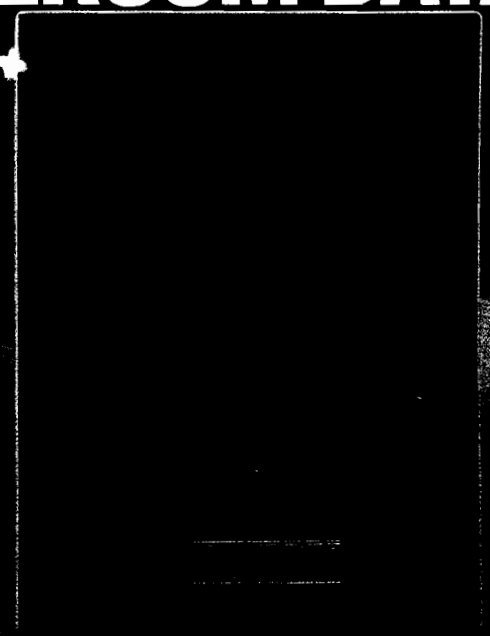
USE OUR 60 DAY UNCONDITIONAL MONEY BACK TRIAL:
IF YOU CAN PART WITH IT—WE'LL BUY IT BACK!
ASK ABOUT DEALER/USER GROUP DISCOUNTS!

VISA MasterCard

*G.P.L.E. © 1983 NEIL KONZEN. SOLD UNDER LICENSE FROM SYNERGISTIC SOFTWARE / APPLE IS A REGISTERED TRADEMARK OF APPLE COMPUTER INC. A.P.P.L.E. IS APPLE PUGETSOUND PROGRAM LIBRARY EXCHANGE, THE WORLD'S LARGEST APPLE USERS' GROUP WITH 25,000 MEMBERS.

ATARI COMPUTER OWNERS:

Pick the positively perfect, practical, peripheral package, from PERCOM DATA!



That's right ... the positively perfect PERCOM DATA 5¼" floppy disk drive with a BUILT-IN PRINTER-PORT, for your Atari® 400/800 is now available!

Until now, Atari computer owners who wanted to hook a printer to their computer had only one choice... spend about $220 for an interface device. THOSE DAYS ARE OVER. PERCOM DATA has built a parallel printer-port right into its new AT88 PD model. Now you can add a quality disk drive system AND have a place to plug in a printer ... WITHOUT BUYING an interface.

The AT88 S1 PD™ disk drive operates in both single density (88K bytes formatted) and double density (176K bytes formatted).

What more could you want? NO INTERFACE ... a high quality PERCOM DATA disk drive ... AND a built-in PRINTER-PORT ... all with a price of $599.

Pick up a positively perfect PERCOM DATA disk drive with printer-port ... pronto!

Perfectly Priced

$599.

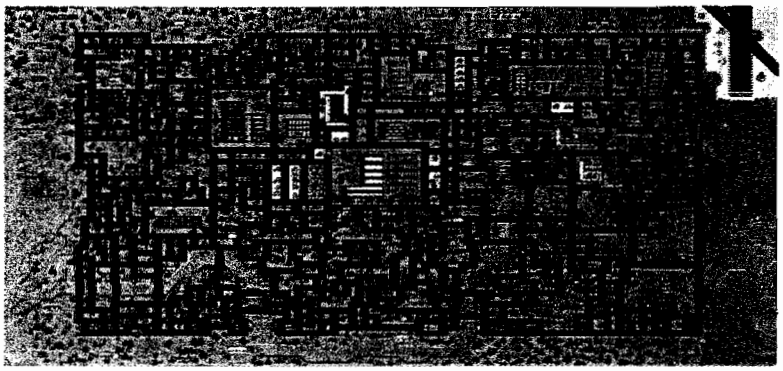
For the name of an authorized PERCOM DATA Dealer near you, call our **TOLL-FREE HOTLINE 1-800-527-1222 NOW**, or write for more information.

PERCOM DATA

Expanding Your Peripheral Vision

GETAWAY!

You've got the loot ... now, *GETAWAY!* to a great new game from the Atari® Program Exchange!



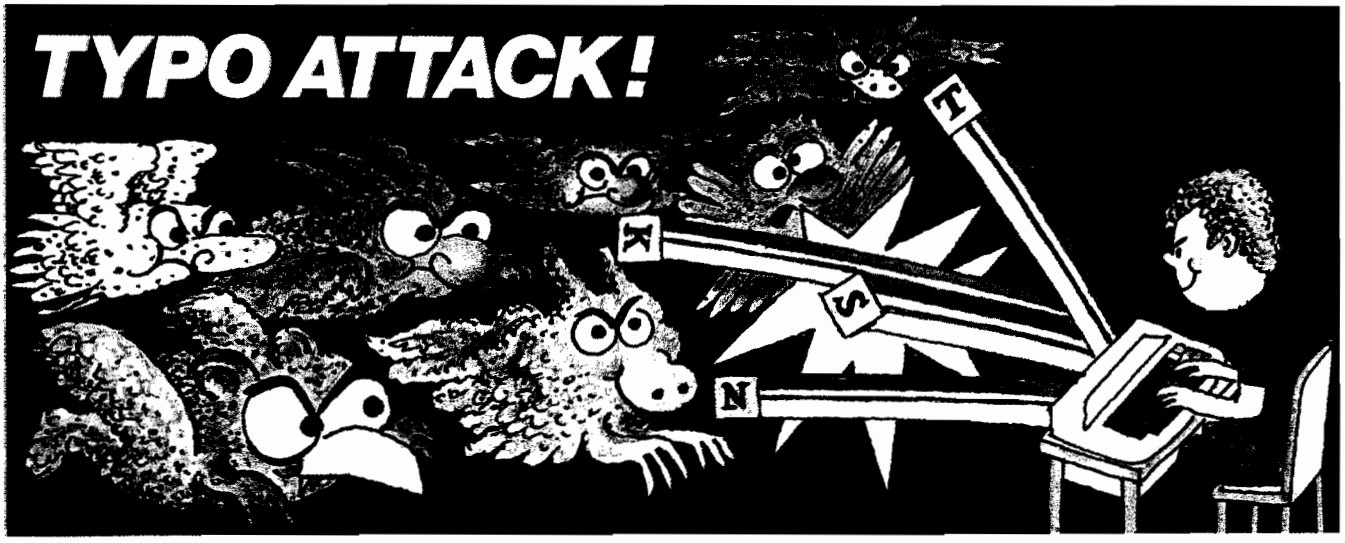
It's *all* there! The squall of sirens, the crazy turns down endless city streets, the anxious search for ill-gotten gain, the race against time for a safe place to stash your cash! Now your gas tank is nearly empty and night is about to fall. The coppers are closing in fast. Before you learn again that crime doesn't pay...Quick! ***GETAWAY!***

Ask for *GETAWAY!* at your local Atari software retailer, or order direct. Phone 800-538-1862, or 800-672-1850 in California. Or write Atari Program Exchange, P.O. Box 3705, Santa Clara, CA 95055.

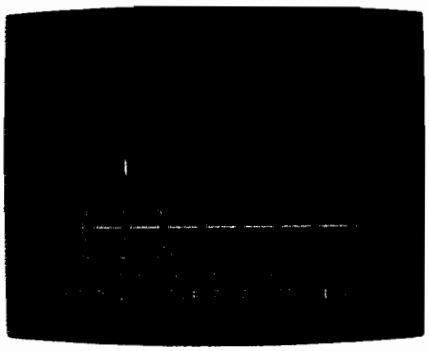
Cassette (410)	APX-10195 32K	$29.95
Diskette (810)	APX-20195 32K	$29.95

For direct orders, add $2.50.

TYPO ATTACK!



You're in for a nasty spell ... unless you stop the Typos!



In the dusky world beneath your keyboard the gruesome Typos dwell ...waiting to attack! Term paper due tomorrow? Got to get that book report typed? Fool! The Typos will devour your letters as you type! That could spell D-O-O-M-E for *you!*

Before you start typing, get down to the real work: destroy the Typos before they destroy your proase... uh, proze...prrrose...Oh NO! THE TYPOS!!! Get ***TYPO ATTACK***, a grand and glorious game from Atari® Program Exchange. It might even improve your typing!

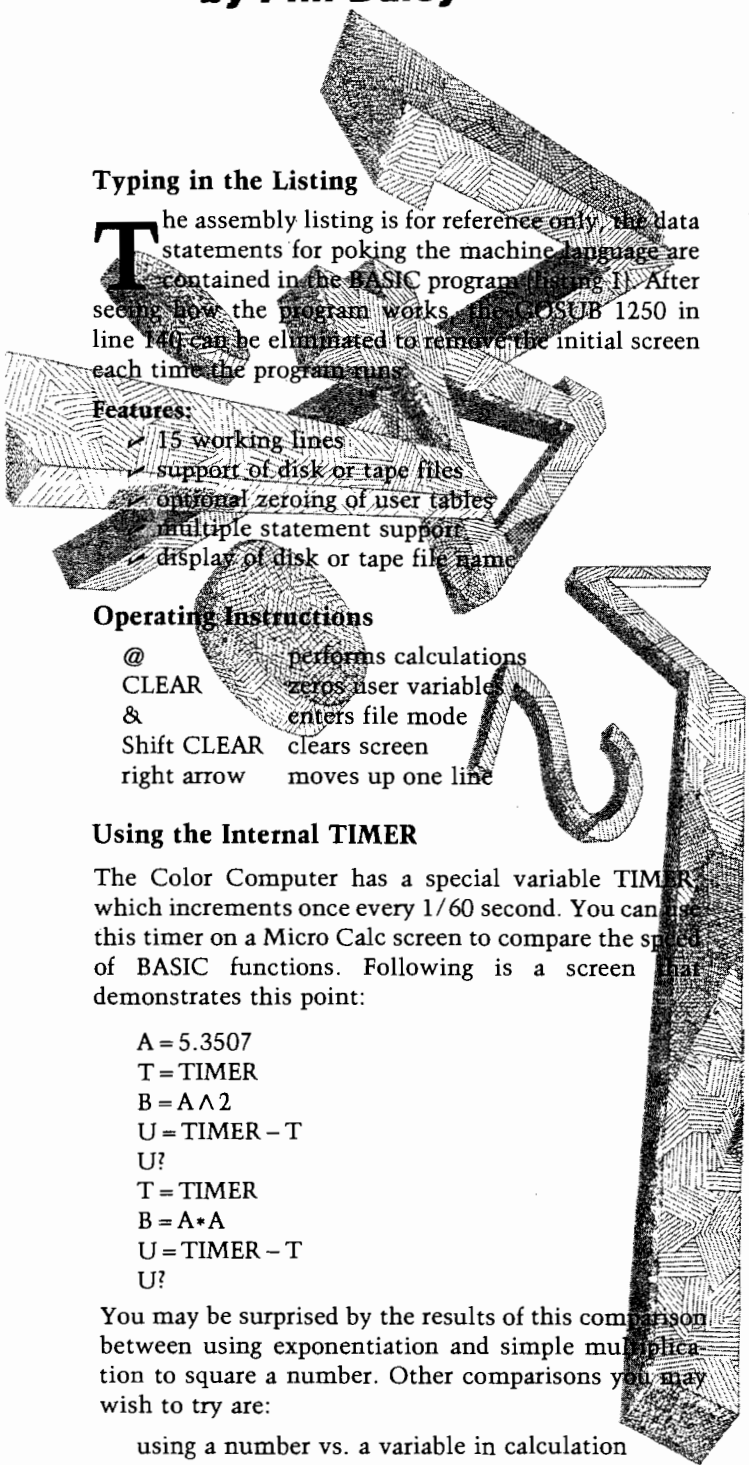
Ask for ***TYPO ATTACK*** at your local Atari software retailer, or order direct. Phone 800-538-1862, or 800-672-1850 in California. Or write Atari Program Exchange, P.O. Box 3705, Santa Clara, CA 95055.

Cassette (410):	APX-10180	8K	$29.95
Diskette (810):	APX-20180	16K	$29.95

For direct orders, add $2.50.

MICRO CALC

by Phil Daley



Typing in the Listing

The assembly listing is for reference only, the data statements for poking the machine language are contained in the BASIC program (listing 1). After seeing how the program works, the GOSUB 1250 in line 140 can be eliminated to remove the initial screen each time the program runs.

Features:

- 15 working lines
- support of disk or tape files
- optional zeroing of user tables
- multiple statement support
- display of disk or tape file name

Operating Instructions

@	performs calculations
CLEAR	zeros user variables
&	enters file mode
Shift CLEAR	clears screen
right arrow	moves up one line

Using the Internal TIMER

The Color Computer has a special variable TIMER which increments once every 1/60 second. You can use this timer on a Micro Calc screen to compare the speed of BASIC functions. Following is a screen that demonstrates this point:

```
A = 5.3507
T = TIMER
B = A ∧ 2
U = TIMER – T
U?
T = TIMER
B = A*A
U = TIMER – T
U?
```

You may be surprised by the results of this comparison between using exponentiation and simple multiplication to square a number. Other comparisons you may wish to try are:

using a number vs. a variable in calculation
the SQR() function vs. raising to the .5 power
SIN() vs. COS()

Listing 1

```
10 'MICROCALC for the TRS80CoCo
20 'By P. Daley
30 'Version 1.0 : 10/14/83
40 'Copyright (C) 1983
50 'by MICRO Ink
60 '10 Northern Blvd.
70 'Amherst, NH 03031
80 'PRETEND IT'S A 16K MACHINE
90 'AND SAVE A FEW BYTES FOR M/L
100 CLEAR1000,16282
110 GOSUB420: GOSUB1190
120 B1=32
130 DEF USR0=16283: CLS2
140 DIM B$(15): GOSUB1250
150 C$=STRING$(32,159)                                    L
160 GOTO1150
170 PRINT@32*X1,B$(X1);CHR$(B1);LEFT$(C$,30-LEN(B$(X1)));
180 RETURN
190 X1=0:Y1=0
200 GOSUB170
210 A$=INKEY$                                             G
220 IFA$=""THEN210
230 IFA$="@"THENB1=32:GOSUB170:GOTO490                    D
240 IFA$=CHR$(12)THENGOSUB1230:GOTO210
250 IFA$=CHR$(92)THENFORI1=0TO14:B$(I1)="":NEXT:
    B1=32:GOTO1150
260 IFASC(A$)>39ANDASC(A$)<96THEN340
270 B1=32:GOSUB170
280 IFA$=CHR$(13)THENX1=X1+1:IFX1>14THENX1=0
290 IFA$=CHR$(8)ANDY1>0THENB$(X1)=LEFT$(B$(X1),
    LEN(B$(X1))-1):Y1=Y1-1
300 IFA$=CHR$(9)THENX1=X1-1:IFX1<0THENX1=14               B
310 IFA$=CHR$(10)THENX1=X1+1:IFX1>14THENX1=0
320 IFA$="&"THEN990
330 GOTO380
340 Y1=Y1+1:IFY1>29THENB1=32:GOSUB170:X1=X1+1:Y1=0
350 IFLEN(B$(X1))=>29THENX1=X1+1:GOTO370
360 B$(X1)=B$(X1)+A$
370 IFX1>14THENX1=0
380 Y1=LEN(B$(X1)): B1=95
390 IFMID$(B$(X1),2,1)="?"THENB$(X1)=LEFT$(B$(X1),2)
400 GOSUB170
410 GOTO210
420 'M/L ROUTINE TO EVALUATE
430 'EXPRESSION AND RETURN
440 FORI1=16283TO16283+26                                 K
450 READA1:POKEI1,A1:NEXT:RETURN
460 DATA 158,166,52,16,142,2,221,159,166
470 DATA 189,184,33,142,2,220,159,166,166,132
480 DATA 189,173,198,53,16,159,166,57
490 'ROUTINE TO POKE EXPRESSIONS
500 'SET UP VARIABLES AND PRINT
510 FORI1=0TO14: CT=0: BUF=732
520 IFLEN(B$(I1))<2THEN 660                               E
530 IFMID$(B$(I1),2,1)<>"="THEN610
540 FORJ1=1TOLEN(B$(I1))
550 IFMID$(B$(I1),J1,1)=":"THEN FLAG=1:GOSUB580:GOTO570
560 POKEBUF+J1,ASC(MID$(B$(I1),J1,1)): CT=CT+1
570 NEXT
580 POKEBUF+J1,13: BUF=BUF-(CT+1)                         F
590 CT=0: Z1=USR0(Z1)
600 IF FLAG=1 THEN FLAG=0: RETURN
610 IFMID$(B$(I1),2,1)<>"?"THEN660
620 KK=ASC(LEFT$(B$(I1),1))-64
630 GOSUB700
640 B$(I1)=LEFT$(B$(I1),2)+STR$(Z1)
650 X1=I1:B1=32:GOSUB170
660 NEXT
670 X1=0:B1=95:GOSUB170
```

(Continued on next page)

Listing 1 *(continued)*

```
680 GOTO210
690 KK=ASC(LEFT$(B$(I1),1))-64
700 ON KK GOTO 720,730,740,750,760,770,780,790,800,810,
    820,830,840,850,860,870,880,890,900,910,920,930,940,
    950,960,970
710 RETURN
720 Z1=A:RETURN
730 Z1=B:RETURN
740 Z1=C:RETURN
750 Z1=D:RETURN
760 Z1=E:RETURN
770 Z1=F:RETURN
780 Z1=G:RETURN
790 Z1=H:RETURN
800 Z1=I:RETURN
810 Z1=J:RETURN
820 Z1=K:RETURN
830 Z1=L:RETURN
840 Z1=M:RETURN
850 Z1=N:RETURN
860 Z1=O:RETURN
870 Z1=P:RETURN
880 Z1=Q:RETURN
890 Z1=R:RETURN
900 Z1=S:RETURN
910 Z1=T:RETURN
920 Z1=U:RETURN
930 Z1=V:RETURN
940 Z1=W:RETURN
950 Z1=X:RETURN
960 Z1=Y:RETURN
970 Z1=Z:RETURN
980 RETURN
990 CLS:PRINT@64,"SAVE OR LOAD?"
1000 INPUTA$
1010 PRINT:INPUT"FILENAME: ";F$
1020 IFLEN(F$)=0THENF$="MICRO"
1030 F$=F$+".CAL"
1040 B$(15)=F$
1050 IFLEFT$(A$,1)="L"THEN1110
1060 OPEN "O",#TD,F$
1070 FORI1=0TO14
1080 WRITE #TD,B$(I1)
1090 NEXT: CLOSE #TD
1100 GOTO1150
1110 OPEN "I",#TD,F$
1120 FORI1=0TO14
1130 INPUT #TD,B$(I1)
1140 NEXT:CLOSE #TD
1150 CLS2:FORX1=0TO15
1160 GOSUB170
1170 NEXT:B1=95
1180 GOTO190
1190 CLS:PRINT"tAPE OR dISK SYSTEM: ";:INPUTA$
1200 IF LEFT$(A$,1)="D"THEN TD=1 ELSE TD=-1
1210 RETURN
1220 D=99
1230 A=0:B=0:C=0:D=0:E=0:F=0:G=0:H=0:I=0:J=0:K=0:L=0:M=0:
    N=0:O=0:P=0:Q=0:R=0:S=0:T=0:U=0:V=0:W=0:X=0:Y=0:Z=0
1240 RETURN
1250 FORI1=0TO14:READB$(I1):NEXT:RETURN
1260 DATAA=8000,,M=45,,I=11.9,,I=I/1200,,D=(1-(1+I)^-M)/I,
1270 DATAP=A/D,,P=INT(P*100+.5)/100,,P?
```

Listing 2

```
                *       MICRO CALC
                *       Modified for TRS 80C
                *       by P. Daley
                *       October 14, 1983

                *       CONSTANTS

        00A6  CHRPTR  EQU   $A6
        02DC  STRING  EQU   $2DC
        ADC6  COMMAND EQU   $ADC6
        8821  TOKEN   EQU   $8821


9E   A6    LDX   CHRPTR    Get current pointer
34   10    PSHS  X         Save it
8E   02DD  LDX   #STRING+1 Load pointer to input buffer
9F   A6    STX   CHRPTR    Set pointer
BD   8821  JSR   TOKEN     Tokenize string
8E   02DC  LDX   #STRING   Reset pointer to tokenized
9F   A6    STX   CHRPTR    string and store it
A6   84    LDA   ,X        Get first character
BD   ADC6  JSR   COMMAND   Execute immediate mode
35   10    PULS  X         Get old pointer and
9F   A6    STX   CHRPTR    restore it
39         RTS             Return

           END   START
```

MICRO

NOT ONLY ANOTHER TECHNOLOGICAL BREAKTHROUGH

BUT ALSO EASY TO USE AND TO INSTALL

Modems are the most important device used to link two products together through ordinary telephone lines at miles apart. The INCOMM STARCOM, a 300/1200 bps Auto Dial, Auto Answer, Auto Log On modem was introduced to link two high speed devices together with having absolutely NO knowledge of computers or communications in general. Anybody can install and operate the STARCOM Family in a few minutes.

- STARCOM is a 300/1200 bps Auto Dial/Auto Log On/Auto Answer Modem.
- OSCOM is a 300/1200 bps Auto Dial/Auto Log On/Auto Answer with Osborne (TM) Computer Software (included) modem.
- COMSOFT is a communication software package.
- OEM MODEM BOARD is also available for custom installation.

THE STARCOM $450.

THE OSCOM $520.

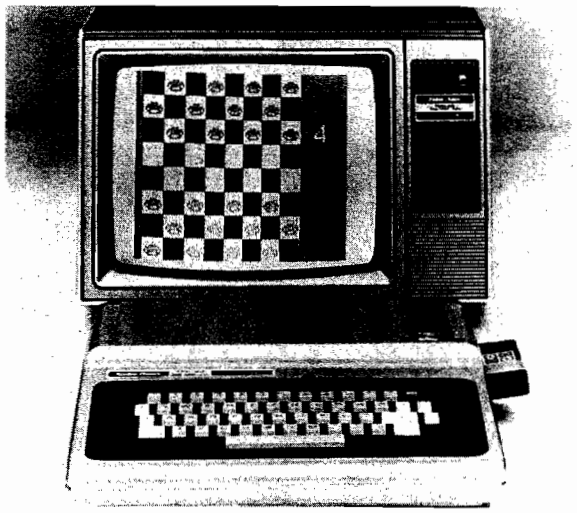
SEND FOR FULL LINE CATALOG of peripherals including Modems, Muxes, Switches and Breaker Boxes.

DEALER INQUIRIES INVITED
(312) 459-8881
or 1-800-323-2666

INCOMM
115 N. WOLF RD. WHEELING, IL 60090

SO COMPACT:
Only 1¼" x 6½" x 7½" 1.5 lbs.
Fits in your coat pocket!

Radio Shack Color Computer Memory Map



(All Numbers in Hex)

Overview

0000-03FF	Ram used by BASIC Interpreter
0400-05FF	Video Display (May be moved)
0600-0FFF	RAM for user program
1000-3FFF	Additional RAM in 16K system
4000-7FFF	Additional RAM in 32K system
8000-9FFF	Extended BASIC ROM
A000-BFFF	Basic Interpreter ROM
C000-FEFF	Cartridge ROM
FF00-FFFF	I/O and Control

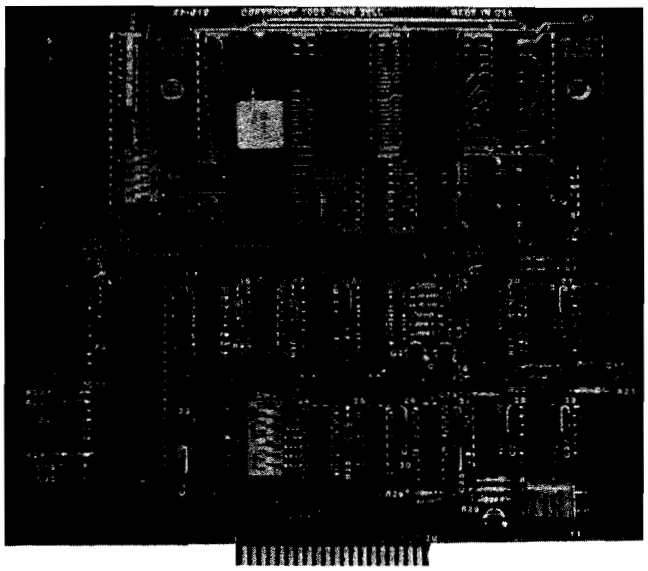
Extended

0003	General Counter
0006	String Flag
0007	Flag if Garbage Collected
0019	Start of User RAM
0019-001A	BASIC Program Begin
001B-001C	Pointer to Top of Program/Begin Variables
001D-001E	Pointer to Top of Variables/Start of Arrays
001F-0020	Pointer to End of Arrays/Start of Available Memory
0021-0022	Top of Stack/Start of String Pool
0023-0024	Start of Used Area of String Pool
0025-0026	Pointer to BASIC Memory Limit
0027-0028	End of String Pool/Start of User Space
0033-0034	Pointer to Current Data Read Position
0037-0038	Current Variable Name
0041	4 Bytes Used by Tokenize
0041-0048	Start and End Address of Block Move
0041	Highest Address to Move to
0043	Highest Address to Move
0045	Lowest Address Moved to
0047	Lowest Address to Move
0047	Highest String Found
004B	Address of Descriptor of Highest String Found
004F-0054	Floating Point Accumulator #1 (6 bytes)
0056	String Length
005C-0061	Floating Point Accumulator #2 (6 bytes)
0062	Sign Comparison
0063	Extended Precision Byte
0068-0069	Current Program Line
006C	Current Column Position
006F	Device Number for Output Character (0 = Screen, $FE = Printer, $FF = Tape, 1-16 = Disk BASIC File#)
0070	EOF on Tape File Flag
0071	Reset Flag = $55 for Warmstart
0072-0073	Restart Pointer (contains $80C0-BASIC Warmstart)
0074-0075	Pointer to End of Memory
0078	File Mode (0 = None, 1 = Input, 2 = Output)
0079	Tape Working Buffer Length
007A-007B	Tape Working Buffer Pointer
007C	Tape File Block Type (0 = Header, 1 = Data, $FF = EOF)
007D	Number of Data Bytes in Cassette I/O Block
007E-007F	Program End Address 1 after a CLOADM
0080	Checksum
0081	Cassette Error #
0082	General Counter
0083	Pulse Width Count
0084	Rise/Fall Flag
0085	Last Sine Value
0087	Last Key Entered
0088-0089	Pointer to Current Cursor Position
008A-008B	Serial Read # of Tries
008C	Sound Frequency
008D-008E	Duration of Sound
008F	Start of Area Downloaded from ROM
0092	Controls Length of Unmodulated Carrier Preceeding Casette I/O
0094	Cursor Color
0095-0096	High and Low bytes of Baud Rate Code (Normally $0057)
0097-0098	Carriage Return Delay (Normally $0001)
0099	Comma Field Width (Normally $10)
009A	Last Comma Field (Normally $70)
009B	Printer Line Width (Normally $84)
009C	Affects positions of Vars. Line-printed in Comma Fields ($00)

009D-009E	Transfer Address after CLOADM
009F	Start of get next character subroutine
00A5	Start of get same character subroutine
00A6	Next Character Pointer
00A8-00AA	Jump Vector to Print OK
00AB-00AE	Extended Product Area
00AF	Trace Flag
00B5	Current Color
00B6	Current PMODE
00B7-00B8	End of Screen1
00B9	Number of Bytes per Line
00BA-00BB	Address of Graphics Page
00BC	$E = Disk system, $6 = No disk
00BD	X1
00BF	Y1
00C1	Color Set 1 = 8
00C3	X2
00C5	Y2
00D7	Temp
00DB	Change Flag
00E6	DLOAD Baud Rate
00E7	Input Timeout Constant
00EA	Operation Code
00EB	Drive Number
00EC	Track
00ED	Sector
00EE	Buffer Address
00F0	Status Returned
0100-0102	Software Interrupt 3 Called by Vector at $FFF2
0103-0104	Software Interrupt 2 Called by Vector at $FFF4
0105-0108	Software Interrupt 1 Called by Vector at $FFFA
0109-010B	Non-Maskable Interrupt Called by Vector at $FFFC Set to $D7AE by Disk
010C-010E	Interrupt Request Called by Vector at $FFF8 Set to $A9B3/Set to $894C by Extended/Set to $D7BC by Disk
010F-0111	Fast Interrupt Vector Called by Vector at $FFF6/Set to $A0F6
0112-0113	High and low bytes of TIMER
0116-0117	Seed for RND Function
011A	Shift Lock Flag
011C	Keyboard Delay Constant
011D-011F	Jump vector to $8489-Print OK
0120-013C	Token Table Directory(Byte 1 = # of Keywords,Byte 2,3 = Address of Table, Byte 4,5 = Address of Subroutines)
0120-0124	BASIC Commands
0125-0129	BASIC Functions
012A-012E	Extended BASIC Commands
012F-0133	Extended BASIC Functions
0134-0138	Disk BASIC Commands
0139-013C	Disk BASIC Functions
013E-013F	Address for USR0
0140-0141	Address for USR1
0142-0143	Address for USR2
0144-0145	Address for USR3
0146-0147	Address for USR4
0148-0149	Address for USR5
014A-014B	Address for USR6
014C-014D	Address for USR7
014E-014F	Address for USR8
0150-0151	Address for USR9
0152-0159	Keyboard Rollover Table
015A-015D	Joystick Readings
015A	Left Joystick Up/Down
015B	Left Joystick Left/Right
015C	Right Joystick Up/Down
015D	Right Joystick Right/Left
015E-0160	Open Device Hook Called at $A5F6/Set to $C426 by Disk
0161-0163	Device Number Check Called at $A5B9/Set to $C838 by Disk
0164-0166	Return Device Parameters Called at $A35F/Set to $C843 by Disk
0167-0169	Character Output Called at $A282/Set to $8273 by Extended/Set to $CB4A by Disk
016A-016C	Character Input Called at $A176/Set to $BCF1 by Extended/Set to $C58F by Disk
016D-016F	Check File OPEN for Input Called at $A3ED/Set to $C818 by Disk
0173-0175	Close All Open Files Called at $A426/Set to $CA3B by Disk
0176-0178	Close One File Called at $A42D/Set to $8286 by Extended/Set to $CA4B by Disk
0179-017B	Print Using Called at $B918/Set to $8E90 by Extended
017C-017E	File Item Scanner Called at $B061/Set to $CC5B by Disk
017F-0181	Break Key Check Called at $A549/Set to C859 by Disk
0182-0184	Get Line From Keyboard Called at $A390/Set to JMP RTS by Disk
0185-0187	Finish Loading ASCII File Called at $A4BF/Set to $CA36 by Disk
0188-018A	Check End Of File Called at $A5CE/Set to $C860 by Disk
018B-018D	Evaluate Operand Called at $B223/Set to $8846 by Extended/Set to $CDF6 by Disk
018E-0190	User Error Called at $AC46/Set to JMP RTS by Disk
0191-0193	Error Called at $AC49/Set to $88F0 by Extended/Set to $C24D by Disk
0194-0196	Run Called at $AE75/Set to $829C by Extended/Set to $C990 by Disk
0197-0199	Hex & Octal Called at $BD22/Set to $87E5 by Extended
019A-019C	Execute Line Called at $AD9E/Set to $82B9 by Extended
019D-019F	Graphics Address Called at $A8C4
01A0-01A2	CLS,GET,PUT etc. Called at $A910,$975C,$8AFA,$8162 Set to $C29A by Disk
01A3-01A5	Tokenize Called at $B821/Set to $8304 by Extended
8000-9FFF	Extended BASIC ROM
01D1	Tape File Length
01D2-01D9	Tape File Name
01DA-02D8	Cassette Buffer
01DA-01E1	CLOADM File Name
01E5-01E6	EXEC Address from Tape

VIDEO TERMINAL BOARD 82-018

This is a complete stand alone Video Terminal board. All that is needed besides this board is a parallel ASCII keyboard, standard NTSC monitor, and a power supply. It displays 80 columns by 25 lines of UPPER and lower case characters. Data is transferred by RS232 at rates of 110 baud to 9600 baud — switch selectable. The UART is controlled (parity etc.) by a 5 pos. dip switch.

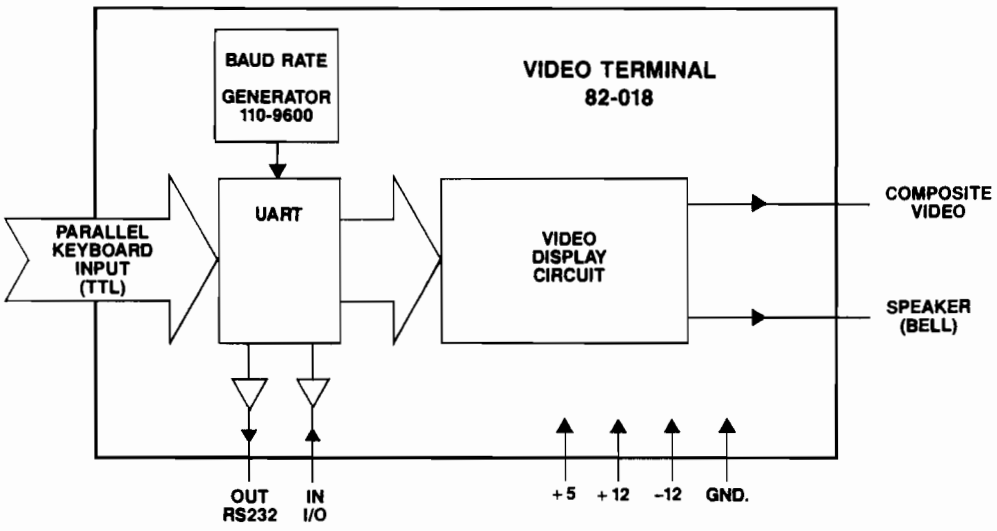


Complete source listing is included in the documentation. Both the character generator and the CRT program are in 2716 EPROMS to allow easy modification to your needs.

This board uses a 6502 Microprocessor and a 6545-1 CRT controller. The 6502 runs during the horz. and vert. blanking (45% of the time). The serial input port is interrupt driven. A 1500 character silo is used to store data until the 6502 can display it.

Features

- **6502 Microprocessor**
- **6545-1 CRT controller**
- **2716 EPROM char. gen.**
- **2716 EPROM program**
- **4K RAM (6116)**
- **2K EPROM 2716**
- **RS232 I/O for direct connection to computer or modem.**
- **80 columns x 25 line display**
- **Size 6.2″ x 7.2″**
- **Output for speaker (bell)**
- **Power +5 700Ma.**
  +12 50Ma.
  –12 50Ma.



This board is available assembled and tested, or bare board with the two EPROMS and crystal.

Assembled and tested	#82-018A	$199.95
Bare board with EPROMS and crystal	#82-018B	$ 89.95

Both versions come with complete documentation.

JOHN BELL ENGINEERING, INC.

ALL PRODUCTS ARE AVAILABLE FROM **JOHN BELL ENGINEERING, INC.** • 1014 CENTER ST., SAN CARLOS, CA 94070
ADD SALES TAX IN CALIFORNIA • ADD 5% SHIPPING & HANDLING 3% FOR ORDERS OVER $100

MasterCard

SEND $1.00 FOR CATALOG

(415) 592-8411
WILL CALL HOURS: 9am - 4pm

10% OUTSIDE U.S.A.
ADD $1.50 FOR C.O.D.

VISA

#249

01E7-01E8	Load Address from Tape
02DC	Contains token for first keyword in BASIC Statement
02DD-03DC	Console I/O Buffer
0400-05FF	Lo-res screen
0600-35FF	Posible Graphic Screens
0600	Bottom of program area/No Disk
0600-06FF	Disk Buffer
0700-07FF	Disk Buffer
0800-0927	Drive Table
097E	Table of Current Tracks
0982	NMI in use flag
0983	NMI JMP
0985	Motor shutoff counter
0986	Current latch data
0C00	Program Start/Disk System
0FFF	Top of memory (4K)
3FFF	Top of memory (16K)
7FFF	Top of memory (32K)
8000-9FFF	Extended BASIC ROM
807F	Cold Start to BASIC without size Search and Workspace init. Resets pointers to Start of BASIC Program
80C0	Warmstart to BASIC. Does not Reset Pointers to Start of BASIC Prog
8183-81EF	Extended Command Token Table
81F0-821D	Subroutine Entry Addresses
821E-8256	Extended Function Token Table
8257-8272	Subroutine Entry Addresses
82B9	Break or Stop Routine
82BB	Extended interpret loop
8378	COSine
8381	TANgent
83B0	ArcTaNgent
8446	LOG
8480	SQuare Root
84F2	EXPonential
8524	FIX
8533	EDIT
86A7	TRace ON
86A8	TRace OFF
86AC	POSition
86BE	VARiable PoinTeR
874E	STRING$
877E	INSTRing
8871	DEFine
8968	TIMER
8970	DELete
8A09	RENUMber
8BDD	HEX$
8C18	DownLOAD
8DBC	Input Serial Character
8E06	Output Serial Character
928F	Find Byte/Bit Routine
92A6	Byte/Bit; PMODES 0,2,4
92C2	Byte/Bit; PMODES 1,3
92DD	Bit Tables
9339	PPOINT
9361	PSET
9365	PRESET
93BB	LINE
9444	Draw Horizontal Line
946C	Draw Vertical Line
94A1	Draw Line
94E2	The Draw Line Loop
9506	Move Up, Down, Left, Right Routines
9532	PCLS
9546	COLOR
9621	PMODE
9670	SCREEN
968B	PCLEAR
9710	Compare Two Points
9723	PCOPY
9755	GET
9758	PUT
98EC	PAINT
9A22	PLAY
9CB6	DRAW
9E9D	CIRCLE
A000-BFFF	BASIC ROM
A000-A001	Address of Check Keyboard
A002-A003	Address of Character Out
A004-A005	Address of Cassette Read On
A006-A007	Address of Block In
A008-A009	Address of Block Out
A00A-A00B	Address of Joystick In
A00C-A00D	Address of Header Out
A00E	Secondary Reset
A027	Primary Reset
A06E	Hardstart (After Reset)
A0A6	Check for Disk ROM
A0CB	Check for Extended ROM
A0D7	Print Version
A0E8	Softstart (After Reset)
A0F6	FIRQ Entry (ROM Pack Check)
A10D	Start of Area Downloaded to RAM at $8F
A129	Start of Area Downloaded to RAM at $10C
A171	Input Character, Bit 7 Clear
A176	Input Character
A199	Blink Cursor Color
A1B1	Wait for Keypress and Read Kybd; Char Returned in A Register
A1C1	Check Keyboard and Get Key if pressed; Z = 1,A = 0 if no key Z = 0,A = key, B and X Preserved
A26E	Table of Codes for non-alpha keys
A282	Output Character to Device Specified by $6F, All But CC Preserved
A2BF	Output Character in A to Printer (RS232)
A30A	Output Character in A to Screen
A390	Input Line from Keyboard into Buffer at $02DD; Return X$02DC; Zero byte at End of Buffer
A416	CLOSE
A44C	CSAVE
A46C	Perform CSAVEM Function; Requires Start of Memory Block in $19-A0 and in $01E7-8, Transfer Address in $01E5-6, and File Name in $01D2-9. Enter with A = 2 and X = 0.
A498	CLOAD
A4FE	CLOADM
A53E	EXEC

A564	INKEY$
A59A	Transfer Block
A5CE	EOF
A5EC	SKIPF
A5F6	OPEN
A629	Open Tape File
A681	Find Filename
A6FE	Blink Screen Corner
A701	READ Block from Tape
A70B	Read a Block from Cassette; Must be On and In Bit Sync. $7C Contains File Block Type:0 = File Header, 1 = data, $FF = EOF. $7D Contains Number of Data Bytes in File (0-$FF). Z = 1, A = 0 if no Errors, Z = 0, A = 1 if Checksum Error, Z = 0, A = 2 if Memory Error. X = Buffer Start Block Length if no Error, X Points to Beyond Bad Address if Error. U and Y Preserved
A77C	Start Cassette and Get Into Bit Sync for Reading. U and Y Preserved, FIRQ and IRQ Masked.
A7BD	MOTOR
A7D8	Turn Cassette On and Write Leader
A7E5	Write Tape File
A7E9	Turn Off Motor
A7F4	Write Block to Cassette; Tape to Speed and Leader Written, $7E = Buffer Address, $7C = Block Type, $7D = Number of Data Bytes, X = Buffer Address Data Bytes, All Registers Modified
A85C	Sine Table for Cassette Out
A880	SET
A8B1	RESET
A8F5	POINT
A910	CLS
A928	Clear Screen and Home Cursor
A937	Print Copyright (CLS 9)
A94B	SOUND
A956	Generate Sound
A992	AUDIO
A9B3	Interrupt Processor (60 Hz Counter)
A9C6	JOYSTICK
A9DE	Read and Store Joystick Values; Left:Up/Down is $15A,Rt/Lft is $15B; Right:Up/Down is $15C,Rt/Lft is $15D. Y is Preserved
AA29	Function Address Table
AA51	Operation Table for +, −, *, /, , AND,OR (3 bytes each-Addresses and Precedence Values)
AA66	Command Name Table
AB1A	Function Name Table
AB67	Command Address Table
ABAF	Error Code Table
ABE1	Text Strings
ABF9	Search Stack for GOSUB or FOR
AC1E	Open up space in memory
AC20	Move Block of Memory Starting at Top; $41-2 is Destination Top Address, $43-4 is Source Top Address, $45-6 is Destination Bottom Address after Move, $47-8 is Source Bottom Address
AC46	Error Handler
AC73	Idle Loop
AD17	NEW (Clear Memory)
AD19	Execute NEW
AD47	FOR
AD9E	Interpret Loop
ADC6	Execute line
ADE4	RESTORE
ADEB	Check for Break or Pause
AE02	END
AE09	STOP
AE30	CONTinue
AE41	CLEAR
AE75	RUN
AE86	GO
AE92	GOSUB
AEA4	GOTO
AEC0	RETURN
AEE0	DATA
AEE3	REM or '
AEE8	ELSE
AF14	IF
AF42	ON
AF67	Get Unsigned Integer
AF89	LET
AFF5	INPUT
B046	READ
B0F8	NEXT
B156	Get Expression
B1CB	Another Entry in Operation Table
B223	Get Operand
B290	Execute Functions
B2D4	AND/OR Operations
B2F4	Relational Operations
B34E	DIMension
B38F	Variable Creation
B3E4	Evaluate Integer Expression
B3ED	Convert Number in FPAC into 16-bit Two's Complement Integer Left in D Register; Overflow, return to BASIC if > +32767 or <-32768
B4EE	MEM
B4FD	STR$
B518	Get String
B56D	Allocate string routine
B591	Garbage Collect
B5D8	Process one descriptor
B5EF	Compact one string
B671	LEN
B68C	CHR$
B6A0	ASC
B6AB	LEFT$
B6C8	RIGHT$
B6CF	MID$
B716	VAL
B750	PEEK
B757	POKE
B75E	LLIST Command
B764	LIST Command
B7C2	Untokenize

B7E6	Untokenize one token
B821	Tokenize
B892	Tokenize one word
B8F7	PRINT
B97E	TAB
B99C	Print Text String
B9AC	Print a Space
B9B4	Start of Floating Point Routines-Rounding
B9B9	Subtract from FPAC1
B9C2	Add to FPAC1
BA79	Two's Complement FPAC1
BAC5	Constant 1.0
BACA	Multiply
BB2F	Move [X] to FPAC2
BB7D	Constant 10.0
BB91	Divide
BC4A	Move FPAC2 to FPAC1
BC5F	Move FPAC1 to FPAC2
BC6D	Test FPAC1 for Zero and Sign
BC7A	SiGN
BC93	ABSolute value
BCEE	INTeger
BD12	Convert String to Floating Point
BDB6	Constants 99999999.9, 999999999, 1E09
BDCC	Display the Decimal Value in D Register
BDD9	Convert FPAC1 to ASCII
BEC0	Constant 0.5
BEC5	Series of 4 Byte Constants
BF1F	RaNDom
BF78	SINe
BFBD	Constants 2 pi, 0.25
BFC8	Series of 5 Byte Constants
BFF2	Interrupt and Reset Vectors
BBF2-BBF3	SWI3
BBF4-BBF5	SWI2
BBF6-BBF7	FIRQ
BBF8-BBF9	IRQ
BBFA-BBFB	SWI1
BBFC-BBFD	NMI
BBFE-BBFF	RESET
C000-D7FF	Disk BASIC ROM
C004	Address of DSKCON
C0D4	Warm Start to Disk BASIC
C17F-C1DA	Disk Command Token Table
C1DB-C200	Disk Subroutine Addresses
C6C2	KILL
C932	SAVE
C98B	MERGE
C99A	LOAD
CBCF	DIRectory
CD1A	CVN
CD28	MKN$
CD36	LOC
CD5B	LOF
CDC0	FREE
CDE9	DRIVE
CF3F	RENAME
CF8A	WRITE
CFE0	FIELD
D025	RSET
D026	LSET
D080	FILES
D146	UNLOAD
D175	BACKUP
D2CC	COPY
D3FF	DSKI$
D474	DSKO$
D4AB	DSKINI
D65B	VERIFY
D66C	DSKCON
D6C5	Restore
D6DE	Get Status
D6FD	Delay 78 msec
D705	Read/Write sector
D7A2	Command Address Table
D7AA	Bit Table for Drives
D7AE	NMI Handler
D7BC	IRQ Handler
FF00-FFFF	I/O and Control
FF00-FF03	PIA U8

FF00 Bit 0-Keyboard Row 1 and Right joystick switch
Bit 1-Keyboard Row 2 and Left joystick switch
Bit 2-Keyboard Row 3
Bit 3-keyboard Row 4
Bit 4-Keyboard Row 5
Bit 5-Keyboard Row 6
Bit 6-Keyboard Row 7
Bit 7-Joystick comparison input

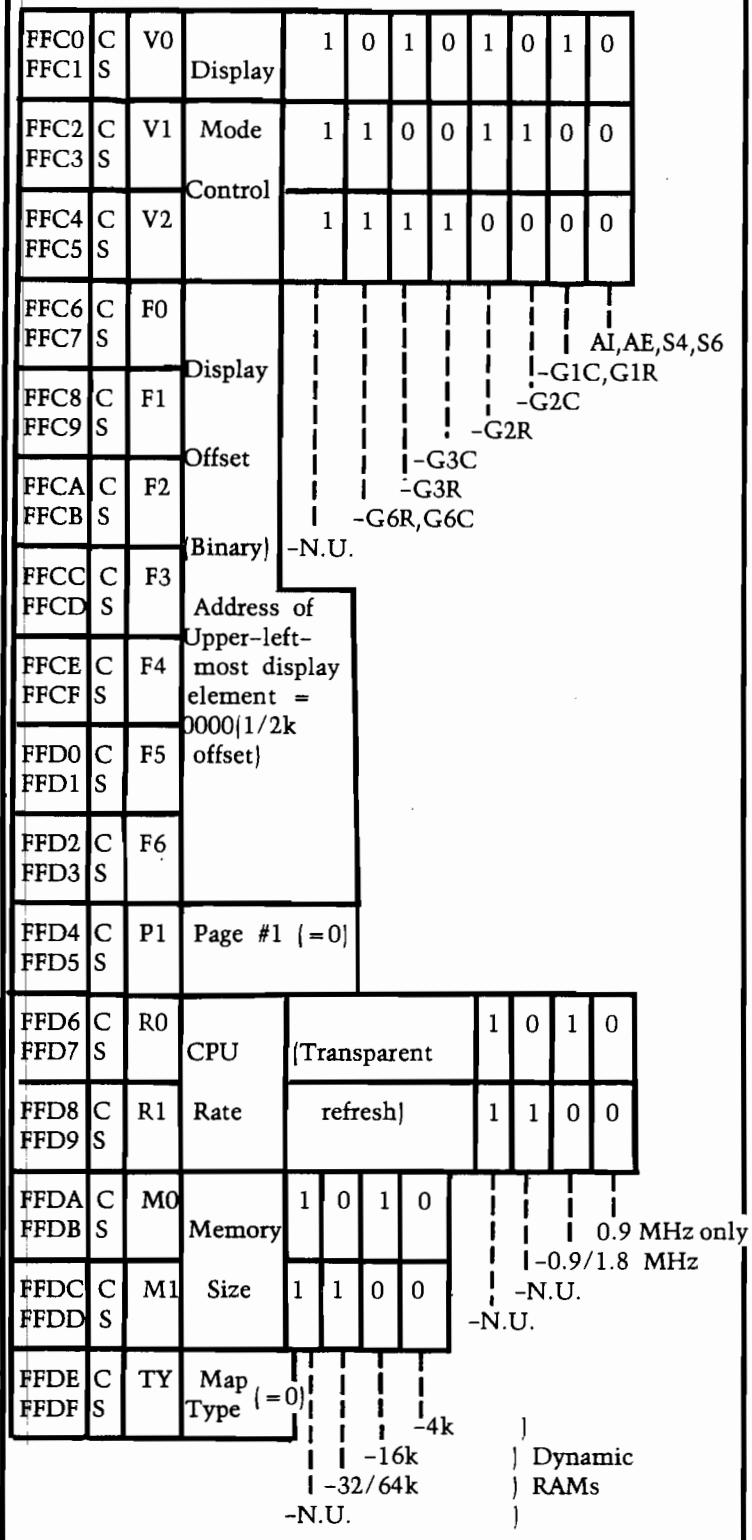
FF01 Bit 0-Control of the Horizontal sync clock(63.5 microsec)
Bit 1-;interrupt input
;
Bit 2-Normally 1 — 0 = Changes FF00 to data direction register
Bit 3-SEL 1: — LSB of the two analog MUX select lines
Bit 4-1 Always
Bit 5-1 Always
Bit 6-Not used
Bit 7-Horizontal sync interrupt flag

FF02 Bit 0-Keyboard Column 1
Bit 1-Keyboard Column 2
Bit 2-Keyboard Column 3
Bit 3-Keyboard Column 4
Bit 4-Keyboard Column 5
Bit 5-Keyboard Column 6
Bit 6-Keyboard Column 7
Bit 7-Keyboard Column 8

FF03 Bit 0-Control of the field ;sync clock 16.667 MS
Bit 1-;interrupt input
;
Bit 2-Normally 1 — 0 = changes FF02 to data direction register
Bit 3-SEL 2 — MSB of the two analog MUX select lines
Bit 4-1 Always
Bit 5-1 Always
Bit 6-Not used
Bit 7-Feld sync interrupt flag

FF20-FF23 PIA U4

FF20 Bit 0-Cassette data input
Bit 1-RS-232 data output
Bit 2-6 bit D/A LSB
Bit 3-6 bit D/A
Bit 4-6 bit D/A
Bit 5-6 bit D/A
Bit 6-6 bit D/A
Bit 7-6 bit D/A MSB
FF21 Bit 0-Control of the CD;
;RS-232 status input
Bit 1-;
;
Bit 2-Normally 1
Bit 3-Cassette motor control 0 = Off 1 = On
Bit 4-1 Always
Bit 5-1 Always
Bit 6-Not used
Bit 7-CD interrupt flag
FF22 Bit 0-RS-232 Data input
Bit 1-Single bit sound output
Bit 2-RAM size input
Bit 3-VDG Control output
Bit 4-VDG Control output
Bit 5-VDG Control output
Bit 6-VDG Control output
Bit 7-VDG Control putput
FF23 Bit 0-;Control of the
;Cartridge interrupt
Bit 1-;input
;
Bit 2-Normally 1 — 0 changes FF22 to data direction register
Bit 3-Six bit sound enable
Bit 4-Always 1
Bit 5-Always 1
Bit 6-Not used
Bit 7-Cartridge interrupt flag
FF40 Output latch
Bit 0-Drive Select 0
Bit 1-Drive Select 1
Bit 2-Drive Select 2
Bit 3-Motor On
Bit 4-Precomp
Bit 5-Double Density
Bit 6-Drive Select 3
Bit 7-Halt Enable
FF48 Disk Status
FF49 Disk Track Number
FF4A Disk Sector Number
FF4B Disk Data



FFE0-FFF1 Not used
FFF2-FFF3 SWI3 Vector
FFF4-FFF5 SWI2 Vector
FFF6-FFF6 FIRQ Vector
FFF8-FFF8 IRQ Vector
FFFA-FFFB SWI1 Vector
FFFC-FFFD NMI Vector
FFFE-FFFF Reset Vector

Tokens for CoCo, Dragon 32 and MC – 10

HEX	DEC	COCO	DBL	DRAG	DBL	MC – 10
80	128	FOR	SGN	FOR	SGN	FOR
81	129	GO	INT	GO	INT	GOTO
82	130	REM	ABS	REM	ABS	GOSUB
83	131	'	USR	'	POS	REM
84	132	ELSE	RND	ELSE	RND	IF
85	133	IF	SIN	IF	SQR	DATA
86	134	DATA	PEEK	DATA	LOG	PRINT
87	135	PRINT	LEN	PRINT	EXP	ON
88	136	ON	STR$	ON	SIN	INPUT
89	137	INPUT	VAL	INPUT	COS	END
8A	138	END	ASC	END	TAN	NEXT
8B	139	NEXT	CHR$	NEXT	ATN	DIM
8C	140	DIM	EOF	DIM	PEEK	READ
8D	141	READ	JOYSTK	READ	LEN	LET
8E	142	RUN	LEFT$	LET	STR$	RUN
8F	143	RESTORE	RIGHT$	RUN	VAL	RESTORE
90	144	RETURN	MID$	RESTORE	ASC	RETURN
91	145	STOP	POINT	RETURN	CHR$	STOP
92	146	POKE	INKEY$	STOP	EOF	POKE
93	147	CONT	MEM	POKE	JOYSTK	CONT
94	148	LIST	ATN	CONT	FIX	LIST
95	149	CLEAR	COS	LIST	HEX	CLEAR
96	150	NEW	TAN	CLEAR	LEFT$	NEW
97	151	CLOAD	EXP	NEW	RIGHT$	CLOAD
98	152	CSAVE	FIX	DEF	MID$	CSAVE
99	153	OPEN	LOG	CLOAD	POINT	LLIST
9A	154	CLOSE	POS	CSAVE	INKEY$	LPRINT
9B	155	LLIST	SQR	OPEN	MEM	SET
9C	156	SET	HEX$	CLOSE	VARPTR	RESET
9D	157	RESET	VARPTR	LLIST	INSTR	CLS
9E	158	CLS	INSTR	SET	TIMER	SOUND
9F	159	MOTOR	TIMER	RESET	PPOINT	EXEC
A0	160	SOUND	PPOINT	CLS	STRING$	SKIPF
A1	161	AUDIO	STRING$	MOTOR	USR	TAB(
A2	162	EXEC	CVN	SOUND		TO
A3	163	SKIPF	FREE	AUDIO		THEN
A4	164	TAB(	LOC	EXEC		NOT
A5	165	TO	LOF	SKIPF		STEP
A6	166	SUB	MKN$	DEL		OFF
A7	167	THEN	AS	EDIT		+
A8	168	NOT		TRON		–
A9	169	STEP		TROFF		*
AA	170	OFF		LINE		/
AB	171	+		PCLS		^
AC	172	–		PSET		AND
AD	173	*		PRESET		OR
AE	174	/		SCREEN		>
AF	175	^		PCLEAR		=
B0	176	AND		COLOR		<
B1	177	OR		CIRCLE		SGN
B2	178	>		PAINT		INT
B3	179	=		GET		ABS
B4	180	<		PUT		USR
B5	181	DEL		DRAW		RND
B6	182	EDIT		PCOPY		SQR
B7	183	TRON		PMODE		LOG
B8	184	TROFF		PLAY		EXP
B9	185	DEF		DLOAD		SIN
BA	186	LET		RENUM		COS

(Continued on next page)

BB	187	LINE
BC	188	PCLS
BD	189	PSET
BE	190	PRESET
BF	191	SCREEN
C0	192	PCLEAR
C1	193	COLOR
C2	194	CIRCLE
C3	195	PAINT
C4	196	GET
C5	197	PUT
C6	198	DRAW
C7	199	PCOPY
C8	200	PMODE
C9	201	PLAY
CA	202	DLOAD
CB	203	RENUM
CC	204	FN
CD	205	USING
CE	206	DIR
CF	207	DRIVE
D0	208	FIELD
D1	209	FILES
D2	210	KILL
D3	211	LOAD
D4	212	LSET
D5	213	MERGE
D6	214	RENAME
D7	215	RSET
D8	216	SAVE
D9	217	WRITE
DA	218	VERIFY
DB	219	UNLOAD
DC	220	DSKINI
DD	221	BACKUP
DE	222	COPY
DF	223	DSKI$
E0	224	DSKO$

TAB(	TAN
TO	PEEK
SUB	LEN
FN	STR$
THEN	VAL
NOT	ASC
STEP	CHR$
OFF	LEFT$
+	RIGHT$
–	MID$
*	POINT
/	VARPTR
^	INKEY$
AND	MEM
OR	
>	
=	
<	
USING	

MICRO™

Information was gleaned from the following sources in addition to personal observation:

Color Computer News
The Rainbow
80 Micro
John Beckett
John Steiner
Ralph Tenny

FORTH-79

Ver. 2 For your APPLE II/II+

The complete professional software system, that meets ALL provisions of the FORTH–79 Standard (adopted Oct. 1980). Compare the many advanced features of FORTH–79 with the FORTH you are now using, or plan to buy!

FEATURES	OURS	OTHERS
79-Standard system gives source portability.	YES	_____
Professionally written tutorial & user manual	200 PG.	_____
Screen editor with user-definable controls.	YES	_____
Macro-assembler with local labels.	YES	_____
Virtual memory.	YES	_____
Both 13 & 16-sector format.	YES	_____
Multiple disk drives.	YES	_____
Double-number Standard & String extensions.	YES	_____
Upper/lower case keyboard input.	YES	_____
LO-Res graphics.	YES	_____
80 column display capability	YES	_____
Z-80 CP/M Ver. 2.x & Northstar also available	YES	_____
Affordable!	$99.95	_____
Low cost enhancement option:		
Hi-Res turtle-graphics.	YES	_____
Floating-point mathematics. Powerful package with own manual, 50 functions in all, AM9511 compatible.	YES	_____

FORTH–79 V.2 (requires 48K & 1 disk drive) $ 99.95
ENHANCEMENT PACKAGE FOR V.2
Floating point & Hi-Res turtle-graphics $ 49.95
COMBINATION PACKAGE $139.95
(CA res. add 6% tax: COD accepted)

MicroMotion
12077 Wilshire Blvd. #.506
L.A., CA 90025 (213) 821-4340
Specify APPLE, CP/M or Northstar
Dealer inquiries invited.

FOR YOUR APPLE II

Industry standard products at super saver discount prices

SOFTWARE

	List	SGC
ARTSCI		
Magicalc	$149.00	$ 99.00
Magic Window	99.00	65.95
ASHTON TATE		
DBASE (apple)	700.00	449.00
BRODERBUND		
Payroll	$395.00	$255.95
Acct/Rec.	395.00	255.95
Arcade Machine	59.95	39.95
Choplifter	34.95	24.95
Sea fox	29.95	20.95
Galaxy Wars	24.95	20.95
DATAMOST		
Property Mgr.	$295.00	$194.95
Real Estate		
Investment Package	179.95	118.95
Tax Beater	129.95	89.95
Zaxxon	39.95	27.95
Casino	39.95	27.95
Conquering Worlds	29.95	20.95
Mating Zone	29.95	20.95
Pandoras Box	29.95	20.95
Series Baseball	29.95	20.95
INFOROM		
Zork 1, 2, 3	$ 39.95	$ 27.95
Planet Fall	49.95	32.95
Starcross	39.95	27.95
PEACHTREE		
Peachcalc	$150.00	$ 92.95
PeachPak (G1/Ar/Ap/)	395.00	226.95
Mailing List Mgr.	250.00	151.95
SENSIBLE SOFTWARE		
Medical dictionary	$ 99.95	$ 72.95
DOS Plus	25.00	18.95
Legal Dictionary	99.95	72.95
SIR TECH SOFTWARE		
Legacy of Llylgamyn	$ 39.95	$ 27.95
Wizardry	49.96	34.95
Galactic Attack	29.95	21.95
SIRIUS		
Bandits	$ 34.95	$ 24.95
Epoch	34.95	24.95
Fly Wars	29.95	20.95
Beer Run	29.95	20.95
Outpost	29.95	20.95
Snake Byte	29.95	20.95
Sneakers	29.95	20.95
Joy Port	49.95	37.95
STRATIGIC SIMULATIONS		
Germany	$ 59.95	$ 39.95
North Atlantic 1986	59.95	39.95
Rapid Deployment Force	34.95	25.95
Road to Gettysburg	59.95	39.95
Computer Baseball	39.95	28.95
Computer Quarterback	39.95	28.95
Bomb Alley	59.95	39.95
Napoleons Campaigns	59.95	39.95

SPECIAL AND NEW

ALBERT COMPUTER

64K Ram Upper and lower case character set, digitized audio control, RGB graphics, Graphics power pad, 110/220 volts, 8 to 32 dc power source, Magicalc, Magic Window, Magic Speller, Magic Writer.

Special Price **$1395.00**

"New" from Prometheus Products

PRO MODEM 1200 Bell 212A 1200/300 Baud

A. Can be expanded into a telecommunication system as a stand-alone package.
B. Real time clock and calendar
C. Help Commands
D. Built-in diagnostics
E. Auto dialer
F. Programmable Intelligent Dialing
G. RS 232 interface
H. Internal Power Supply

OPTIONS

On line phone directory
Expandable buffer from 16K to 64K

SPECIAL ..... **$ 475.00**

FRANKLIN ACE 1000

64K, 2 disk drives, monitor, 80 column card

SPECIAL **$1499.00**

PROMETHEUS PRODUCTS

16K Card	**$ 65.00**
64K Expand-A-Ram	**$ 225.00**
128K Expand-A-Ram	**$ 275.00**
Graphitti Card (Specify Printer)	**$ 89.00**
P.S. 16K Buffer Card Expandable to 64K	**$ 105.00**
Applesurance Card	**$ 99.00**
VersaCard Four cards in one 1. Serial 2. Parallel 3. Real Time clock and calendar 4. BSR Port	**$ 169.00**
With Graphics	**$ 203.00**
PRT-1 Printer Card	**$ 79.00**

PRINTERS

NEC 8023	**$ 395.00**
with interface card and cables	**$ 449.00**
OKI 82A 80 col./120cps	**$ 439.00**
OKI 83A 136 col./120 cps	**$ 695.00**
Brother Letter Quality	**$ 889.00**
Transtar 315 Color Printer	**$ 469.95**

ACCESSORIES

T.G. Joystick	**$ 39.95**
Select a Port	**$ 39.95**
System Saver Fan	**$ 69.95**
Super Fan 11	**$49.95**
Slimline Disk Drives	**$ 225.00**
Disk Drives Full Ht.	**$ 195.00**
Floppy Disc Box 10	**$ 25.00**
Floppy Disc w/order	**$ 19.90**

BUSINESS SPECIAL

Includes 80 Column Z-80 Card • 16K Card • Wordstar • Supercalc

Special **$825.00**

D.O.S. Programming SPECIAL

80 Column Card • Magicalc • Magic Window • Magic Speller • 16K Card **Special** .. **$449.00**

3-CARD SYSTEM

80 col/16K card/Z-80 card **Special** ... **$329.95**

MODEMS

Smartmodem 300	$219.95
Smartmodem 1200	$499.00
D-Cat Modem	$164.00

MONITORS

Nec Green	$110.00
Taxan Amber	$129.95
Taxan RGB color	$395.00
USI Hi/Res green	$141.95

80 COLUMN CARDS

Smarterm 11	$129.00
Videx Video Term	$215.00
Ultra Term	$289.00
View Master	$141.95
Apple IIe 64K, 80 col.	$ 99.00

Z-80 CARDS

Microsoft Softcard	$229.95
Z-80 Plus	$125.00
Z-Card 11	$131.95

EDUCATIONAL SOFTWARE

Spelling/Reading Primer	$ 29.95
Perception	$ 20.95
Counting Bee	$ 23.95
Word Scrambler	$ 15.95
Fractions	$ 36.95
Intro. Algebra	$ 21.95
Frenzy/Flip Flop	$ 25.95
Wordstar	$244.95
Spellstar	$159.95
WStar/MMerge	$395.00
WStar/SStar	$422.00
Supercal	$129.95
PSF File	$ 88.75
PSF Graph	$ 88.75
PSF Report	$ 88.75
PSF 3PAK	$246.75

All equipment shipped factory fresh. Manufacturers' warranties included. California customers add 6½% tax. Include payment by personal check, money order, or cashier's check with order and SGC will pay shipping charge. Call for amount of shipping charge when paying by credit card.

All items are normally in stock

(415) 490-3420

... And we'll be here to help after you receive your order. Feel free to call the SGC Technical Staff for assistance.

SGC

The mail order specialists

342 Quartz Circle, Livermore, CA 94550

MICRO™

CoCo Bits

by John Steiner

This month we will take a look at interfacing the Color Computer to a disk drive. Last month I promised a new phone number for the Dakota Database. It is 701-281-0233, and is available 24 hours a day, except for maintenance. Since mid-July, we have logged over 1000 calls, and have nearly fifty regular users. Most of the users have computers that are not CoCos. Lots of Osbornes and IBM machines check in daily, in addition to Model IIIs and several CoCos. There are even a few data terminals who make a regular appearance.

64K COCO

I have had a chance to check out the new 64K CoCo, and find it to be not much different from the older units. The new keyboard is nice, and is really the same style board with new keycaps. I like both keyboards very well, and prefer the new one, but those who like a longer throw on the keys should look into the Mark Data model, or one of the other professional keyboards.

The formatting problem I was working on last month has been solved. I found my drives to be out of time, just as was suggested to me. My drive zero was way off, and that was probably the majority of my initialization problems with the 1.1 ROM card. The ROM works well with either computer, and my old drives are purring again.

Interfacing a Drive Unit

I promised a look at drive interfacing with the CoCo, so let's take a look at what is required. First of all, any standard Model III drive will work on a CoCo if it or the cable has been configured properly. This opens up a wide market for drive selection, and CoCo users can either shop for price or quality or both. My BBS has two Tandon TM-100 drives which have performed 24 hours daily for over three months with no I/O errors. On top of that, they are quiet, far quieter than the TEC drive that Radio Shack sells. I have also used the Teac drive units from J&M, and find them to be just as quiet as the Tandon. J&M feels the Teac drive is better for CoCo than the Tandon, but as a practical matter, either seem to work fine.

The Tandon drive is a popular unit, and available from many sources, so we will look at installing these units. The first requirement is a controller board. There are several different brands available, but only two that I know of that are compatible with the Radio Shack format. These are Shack's card and the J&M controller which I have talked about earlier. The Radio Shack card is available as a replacement part, and you can order a replacement case, putting a complete controller together.

The next requirement is a drive cable, which can be ordered from Radio Shack, or you can use any external drive cable for a Model III if you configure your drives (see below). Drive cables are available either in two or four drive versions. The Color Computer drive cables are configured, which means that the cable determines which drive becomes drive zero, and which becomes drive one, etc. Many companies configure the drives, instead. Configuring the cables allows you to swap drives zero and one at any time without internal modification of the units. On the other hand, it is easier to configure the Tandon drive unit, than to configure the cable.

You can buy the configured cable from Radio Shack, or you can order an unconfigured cable from the place you get your Tandon drive units. My recommendation is to order a cable from the drive supplier that has gold plated connectors and configure your drives.

The Tandon TM-100 is a forty track single sided, double density drive. Having forty tracks is of no value to RS-DOS, which writes only to 35 tracks anyway, however if you have FLEX or another operating system, you may be able to use those tracks by formatting your diskettes for forty track use. In FLEX, the NEWDISK command will allow you to specify number of tracks when it initializes a diskette.

Tandon sells their drives without case and power supply, so be sure to ask about this before you buy a drive unit. A bare drive can be found for around $200, and a case and power supply will cost about $50.00. The Dakota Database drive units are housed in a two drive case which cuts down on cost and space. If you are planning on two drives, you might look into that combination.

Drive Configuration

Configuring a Tandon drive is easy. The configuration process allows the controller card to distinguish between drive units. On the Tandon, the configuration is done by jumpering a programmable shunt socket. This 16 pin IC socket is located toward the right rear of the circuit board near the drive connector (See figure 1). Some companies provide a DIP shunt which is sitting in the socket, while others leave you to your own devices. If you did not receive a shunt, a common staple will perform the job quite satisfactorily. Figure two is a diagram and pinout of the socket.

Configuration is easy. Make sure to connect pins 9 and 8 together. This is done on all drives. To configure a drive as drive 0, connect pins 2 and 15. Drive 1 requires connections between 3 ant 14. Drive 2 connections are to pins 4 and 13, while drive 3 connections are made to pins 5 and 12. Make sure no other pins are connected, except 9 and 8 and the desired drive number pins. Figure two shows the illustration for a drive 0. Once a drive is configured it can be used as that drive with either a configured or non-configured cable. To use a drive with a configured cable, just connect it as above. Configuration can be changed at any time, should you desire to switch drives.

One last comment, there is a terminator socket (marked 2F) on the circuit board, located near the edge connector. This contains a termination resistor pack. Remove the pack from the socket on drive 0, and any intermediate drives. Leave it in the last drive on the line. I have heard con-

(Continued on next page)

Figure 1

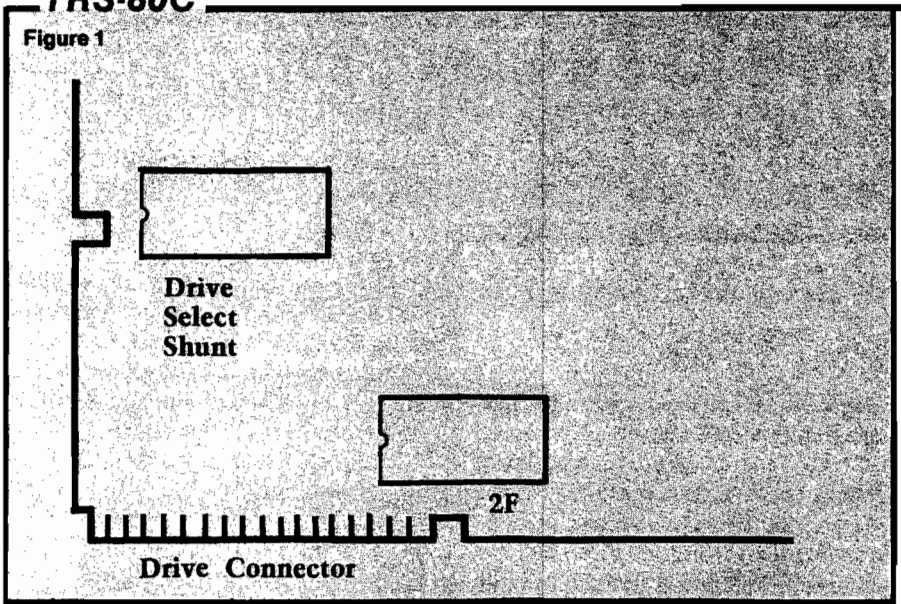
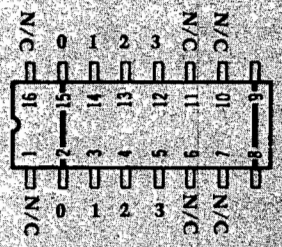


Figure 2



flicting information from different sources about this pack, with some people telling me it can be omitted from all drives. We have left it in our drive 1 without noting any adverse effects. If you have any information about this pack, drop me a line, we will pass it along.

We have installed several Tandon drive systems on both CoCo and J&M cards, and have had no problems If you would like assistance or more information about drives, etc.; give me a call, or drop me a line. I will be glad to help. Send a stamped return envelope for a reply.

Tape Utility

One of the most used utilities in my software collection has been TAPE UTILITY from Spectrum Projects. The program is designed to make it easy to copy files from tape to tape, tape to disk, and vice versa. The programs most useful function is a disk to tape backup. The command BAC is used to transfer all disk files on a given disk to tape. Operation from that point is unattended, and in a little while, all disk files are on tape. There are commands to copy individual files from one media to the other, and a set of directory commands that allow printed and screen directories of both the disk and tape.

There is a tape to disk command that will copy the next tape file to disk, and present you with an option to continue or exit. The command works well, however if you want to dump an entire tape to diskette, you have to be around to prompt the computer to read in the next file. Ken Christiansen of Fargo, ND passed along the following patch that will bypass the prompt and allow the computer to continue to read in tape files. The only disadvantage to this is that when the tape is finished, you have to stop the program with the RESET key.

If you are interested in a patch that will allow the program to work with disk 1.1 ROM, drop me a line with an SASE and cassette. Ken will give you a BASIC program to patch UTIL. Spectrum Projects tells me their latest version now checks for the ROM and will operate properly with either ROM installed. Two things Tape Utility will not do are copy protected programs or copy segmented binary files. It will copy any binary file that was created using CSAVEM. MICRO™

You may contact Mr. Steiner at 508 Fourth Ave. NW, Riverside, ND 58078.

Listing 1

```
5 REM TAPE UTILITY
PATCH TO PROVIDE CON-
TINUOUS CASSETTE READ
7 REM BY KEN
CHRISTIANSEN
10 POKE &H0DF7,&H96
20 POKE &H0DF8,&H0
30 POKE &H0DF9,&H0B7
40 POKE &H0DFA,&H0FF
50 POKE &H0DFB,&H20
60 POKE &H0DFC,&H0BD
70 POKE &H0DFD,&H0A9
80 POKE &H0DFE,&H9E
90 POKE &H0DFF,&H39
100 POKE &H14E0,&H12
110 POKE &H14E1,&H12
120 POKE &H14E2,&H12
130 POKE &H14E3,&H12
140 POKE &H14E4,&H12
150 POKE &H14E5,&H12
160 POKE &H14E6,&H12
170 POKE &H14E7,&H0BD
180 POKE &H14E8,&H0D
190 POKE &H14E9,&H0F7
200 POKE &H1514,&H7E
210 POKE &H1515,&H14
220 POKE &H1516,&H63
230 POKE &H1288,&H7E
240 POKE &H1289,&H12
250 POKE &H128A,&H22
```

Sure it's insured?

SAFEWARE℠ Insurance provides full replacement of hardware, media and purchased software after a low $50 deductible.
As little as $35/yr covers:
• Fire • Theft • Power Surges
• Earthquake • Water Damage • Auto Accident
Select the coverage you want from the table.

Amount of Insurance	Annual Premium
Up to $ 2,000	*$ 35*
$ 2,001-$ 5,000	*$ 60*
$ 5,001-$ 8,000	*$ 75*
$ 8,001-$11,000	*$ 90*
$11,001-$14,000	*$105*

Call for higher coverages.
Not avail. in AK, DC, HI, KY, LA, ME, MS, NV, SC, or WY.
Call for immediate protection.

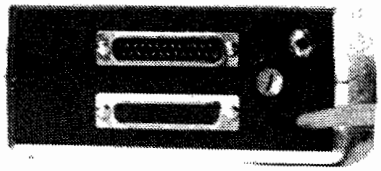
1-800-848-3469
(In Ohio call 1-800-848-2112)

SAFEWARE™
COLUMBIA NATIONAL GENERAL AGENCY

P.O. Box 4364
Flint, Michigan 48504
(313) 233-5731
(313) 233-3125

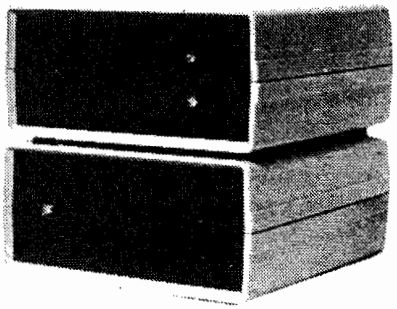
ZANIM SYSTEMS

HOME CONTROL AT YOUR FINGERTIPS



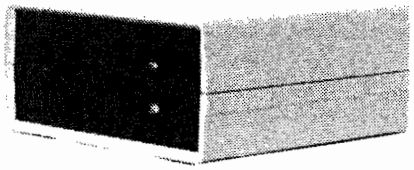
The ZCM-1 is the Master Control module that provides the interface between your computer and our line of Zanim Application Modules. Up to 15 Application Modules can be piggy-backed to the ZCM-1 Master Control module. The ZCM-1 is compatable with any standard RS-232 (serial) interface. A special Master Control module, the ZCM-1V is available for the VIC-20 and Commodore-64 computers.

*The ZCM-1V is available for VIC-20 and C-64 users.



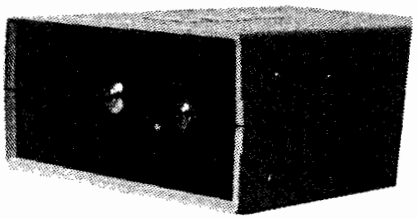
ZAM-1 is the home control interface module that provides a fully versatile computer controlled environment. ZAM-1 can control up to 256 different lamps and appliances in your home or business providing you with an effective and easy to implement energy management and electric control system. No special modifications are necessary to your building as all control signals are sent over your existing wiring. ZAM-1 can be programmed in BASIC or optional home control software is available. ZAM-1 requires one ZCM-1 Master Control module.

*The ZCM-1/ZCM-1V Master Control module is required to use the ZAM-1 Home Control module.



The ZAM-2 allows your computer to continuously monitor up to 15 different doors or windows around your home or business. ZAM-2 is a basic building block in a complete computer controlled home security system. With our ZAM-1 Home Control module, you can have a fully integrated security and environment control system. Upon an intrusion, your computer can take the action most appropriate, whether that is to ring an alarm bell, flash all the lights around your home, or dial the police.

*The ZCM-1/ZCM-1V Master Control module is required to use the ZAM-2 Security module.



The ZAM-3 is a complete telephone answering and dialing system. It is capable of taking the phone off-hook and dialing a number under computer control or of answering the phone when it rings. With the ZAM-1 Home Control module and the ZAM-2 Security module, the ZAM-3 Phone Dialer module can be integrated into a complete home or business security/monitoring system. Applications include security, auto phone dialing, and computer-answering systems.

*Pulse dialing option is available as ZAM-3P.

*The ZCM-1/ZCM-1V Master Control module is required to use the ZAM-3/ZAM-3P Phone Dialer module.

SERIAL OR PARALLEL (CENTRONICS) PORT SWITCHER

DOES YOUR COMPUTER LOOK LIKE THIS?

A PORT SWITCHER NOW AVAILABLE FOR YOUR COMPUTER (ZSW1)

P.O. BOX 4364
Flint, Michigan 48504
(313) 233-5731
(313) 233-3125

☐ Please send me more information or catalogue!

Name _____

Company _____

Address _____

City _____ State _____ Zip _____

What Make/Model Computer do you own?

_____

APPLE 11 IS TRADEMARK OF APPLE COMPUTERS, INC.
CP/M IS REGISTERED TRADEMARK OF DIGITAL RESEARCH, INC.
TRS – 80 IS REGISTERED TRADEMARK OF TANDY CORP
VIC - 20 AND COMMODORE - 64 ARE REGISTERED TRADEMARKS OF COMMODORE. INC

MICRO™ TRS-80C Reviews

Product Name: **C.C. Calc Disk Version**
Equip. req'd: TRS-80 Color Computer 32K
Price: $25.00 tape or disk
Manufacturer: Transformation Technologies
194 Lockwood Lane
Bloomingdale, IL 60108

Description: *C.C. Calc* is an electronic spreadsheet for the Color Computer. The program provides many of the spreadsheet features found on much more expensive software packages. The spread sheet is 26 x 26, which allows 676 cells. Like most spreadsheets, cells can contain labels, formulas or values. One powerful feature is the ability of a cell to contain both a label and a value. This effectively gives a larger sheet as formulas can be hidden under labels. The smaller size and format allows the personal computer user access to a power spreadsheet.

Pluses: One sheet can be merged with another, allowing the creation of larger effective files. Program documentation includes data file configuration, allowing you to read and write data in your own BASIC programs. Files on the disk version can be loaded with only a single key. The program is very powerful considering its low purchase price.

Minuses: Recalculations are not done automatically, you must recalculate each time data is changed. Because the program is in BASIC, recalculation takes a minimum of eight seconds. Parentheses are not evaluated within formulas, so care must be exercised as to proper formula layout.

Documentation: Seventeen pages of documentation and two sample spreadsheets accompany the software. The documentation is thorough, and allows the creation of usable spreadsheets in a short period of time. Users of Visicalc will find many similar commands and capabilities, only on a smaller scale.

Skill level required: Though I have never used a spreadsheet program before, I was easily tutored, and found operation of the program quite easy to learn and use.

Reviewer: John Steiner

Product Name: **Disassembler for 6809**
Equip. req'd: 6809 computer w/Flex
Price: $75.00
Manufacturer: Granite Computer Systems
Route 2 Box 445
Hillsboro, NH 03244
Author: Gilman C. Shattuck

Description: A menu driven 6809 disassembler with user symbol tables. Creates source files compatible with the TSC Editor/Assembler. Output can be to the screen, printer, disk or tape. It has look-up table for Flex and Monitor references, local and global labels and expressions, and equate table for all external references. There is an option for single-step disassembly, data areas are user definable, the program is supplied on 5 or 8 inch disks or tape.

Pluses: The disassembler is menu driven making the use simple and direct. The program is very fast and offers many options for changing data areas, labels, equates, and output. The output listing is the same format as the TSC Assembler and the disk output can be used as input for the assembler. Input is carefully screened to eliminate mistakes.

Minuses: You must have the program loaded before calling the disassembler.

Documentation: The 12 page guide is well written, although a little brief. It describes the menu functions, gives some hints on disassembly and lists some references for more in-depth study.

Skill level required: The documentation assumes familiarity with 6809 machine language programming.

Reviewer: Phil Daley

Product Name: **TRS-80 Model 100 portable computer**
Equip. req'd: 4 AA batteries ($3.00)
Price: 8K $799, 24K $999
Manufacturer: Tandy Corporation
Fort Worth, TX

Description: Gets my vote for product of the year! A truly useful portable computer. Includes all needed software and hardware interfaces for effective use alone or with other computers. 90 day limited warranty. Highly recommended.

Pluses: CMOS 80C85 processor and memory allows up to 20 hours of operation at 2.4MH without a cord. Large 8 line by 40 column LCD display is easily usable indoors and out. Includes full-size full-stroke keyboard, able to generate all ASCII codes, character and high-resolution graphics, and emulate a numeric keypad. Alpha lock, function keys, and cursor controls also supported. Interfaces include a 300 cursor controls also supported. Interfaces in-

clude a 300 baud modem (direct connect cable $20, acoustic coupler planned), RS232 and Centronics parallel ports (cable $15), and a 1500 baud cassette interface (cable $6), as well as sockets for added RAM, ROM and an expansion bus.

Software is fully integrated, menu-driven and supported by function keys, providing ease-of-use comparable to Apple's "LISA" at 1/10th the price. The built-in word processor is simple but elegant, with all needed features. The smart terminal is entirely adequate for even serious use, as is the highly-extended Microsoft BASIC. A primitive address list and notebook are also included.

Minuses: Cursor controls are not supported from BASIC, and line-feeds after carriage returns are not selectable. The quick reference manual and the LCD display could use protective covers, and a built-in microcassette recorder and TV output would be welcome.

Documentation: Comes with a tiny quick reference manual and large spiral-bound user guide with index. Includes enough information in the first few pages to use all programs effectively, and covers all details of use later in a format that is ideal for reference. Does not attempt to teach BASIC to beginners.

Skill level required: My 7 year old used it easily the first day. What more can I say?

Reviewer: Jim Strasma

Product Name: **64K Disk Utility Package**
Equip. req'd: TRS-80 Color Computer Disk system 64K
Price: $21.95 + $3.00 shipping
Manufacturer: Spectrum Projects
93-15 86th Drive
Woodhaven, NY 11421

Description: The *64K disk utility package* is a collection of three useful programs for the 64K Color Computer. Now that Tandy is producing a 64K compatible computer, and many users are upgrading their machines to support 64K, commercial software is starting to use the capacity. The program includes 40K, ROMCRACK, and a print spooler. 40K is a program that moves BASIC from ROM to RAM, and relocates it so that your BASIC programs have access to larger data areas. ROMCRACK will transfer ROMpacks to disks, and the software spooler will allow you to run and use BASIC while the printer is getting spooled output from a buffer in upper memory.

Pluses: 40K provides extra data storage area for large string arrays, etc. The print spooler will allow you to continue programming or working with your computer while printing from a large buffer in the upper RAM. ROMCRACK will transfer most ROMpacks to disk with very little hassle.

Minuses: 40K has a limiting factor in program size, and the extra memory must be used as variable and string space, or the program could crash. The print spooler works

(Continued on next page)

PERRY PERIPHERALS

COMPLETE PRODUCT LINES FOR
D.C. Hayes • HDE • Microsoft • Nashua • Okidata

APPLE • FRANKLIN • IBM-PC/XT
We carry many products at competitive prices to expand these fine computers.
Request Catalog No. AFP*

Dealer Inquiries Invited

HDE • AIM • SYM • KIM
HDE hardware and software for ASK, DLC, OMNI-65 systems
KIM replacement modules (1K — 4K) and keyboards
AIM-65 (1K — 4K) and accessories. Request Catalog No. TASK*

COMPUTER REPAIR SERVICE

Professional Workmanship
Guaranteed Repairs
Quick Turn-around

Apple • Franklin • Atari
PET • HDE
AIM • SYM • KIM

Perry Peripherals

*Catalogs
AFP $2.00 TASK $1.50
Catalog price refunded with first order

Repair Center
6 Brookhaven Drive
Rocky Point, NY 11778

Orders
P.O. Box 924
Miller Place NY 11764
(516) 744-6462
9AM—5PM Weekdays

well with three reservations, it must be available by the time you read this. It slows the computer down, programs run slightly slower. Lastly, the program data being spooled must use BASIC's character output routine (not usually a problem). ROMCRACK won't handle some ROM packs that test to see if the program is residing in RAM before executing. Some packs fit this category.

Documentation: A single sheet of information instructs thoroughly in the operation of the three utilities.

Skill level required: These utilities are for the average BASIC programmer, no great skills are required of the user.

Reviewer: John Steiner

Product Name: **Disk COLORCOM/E Smart Terminal Software**
Equip. req'd: TRS-80/TDP 100 Color Computer/MODEM
Price: $49.95 + $2.00 shipping
Manufacturer: Eigen Systems
P.O. Box 10234
Austin, TX 78766

Description: *The Disk COLORCOM/E Smart Terminal* program is a sophisticated terminal program that supports up/down loading, disk files, and a full complement of RS-232 functions and features.

Pluses: The program is easy to use, and very sophisticated. It is menu driven, and the user can set up just about every possible printer/modem computer parameter desired. All 127 ASCII codes may be sent from the keyboard. The receiver buffer can be opened for saving of data, and closed as desired if you decide to eliminate excess information from your disk. The software handles graphics characters easily, and does an impressive job on Spectrum Projects BBS graphics displays. Initialization files can be saved and loaded for maximum convenience.

Minuses: The software comes on a copy protected diskette, so you must load and run the program from it, transferring to a file disk when loading is complete.

Documentation: A 23-page manual is included that steps the user through the program with ease. Some functions needed further explanation for me, for example, "capture characters".

Skill level required: The program is easy to learn, and beginners can use it with little trouble, ignoring its advanced features. As the user becomes more expert, the extra capability can be put to use.

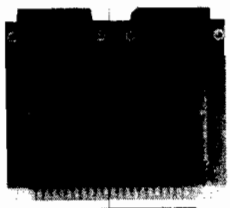
Reviewer: John Steiner

MICRO™

SYM® THE SYSTEM AIM®

SUBSCRIBE TO SYM-PHYSIS! Call (916) 895-8751

DUAL ACIA BOARD
Essential for telecommunications. Two independent full duplex RS232 channels. Uses 6551 ACIAs. Crystal-driven TTL clock. Supports TTL level or RS232C (+/-12v). Supports all standard baud rates to 19,200. Can support interrupt-driven I/O. Demonstration source code provided.
COM-1 $139

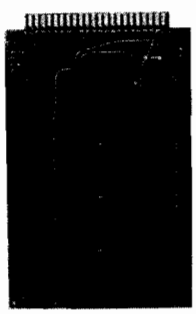


EPROM PROGRAMMER
Particular combination of hardware and software (machine language) used results in VERY LOW programming overhead. Flexible design; supports future EPROM designs. Will program EPROMS up to 256K bits (32K bytes).
Programs all these EPROMS:
2508, 2516, 2532, 2564, 2758A, 2758B, 2716, 27C16, 2732, 2732A, 27C32, 2764, 27C64, 27128, MCM68764
All personality modules INCLUDED.
Menu driven program for easy operation.
PRG-1 $250

REAL-TIME CLOCK/CALENDAR
Demonstration software with source and data sheet. Uses the OKI MSM5832. Provision for user-supplied battery backup.
On board crystal for accurate timekeeping.
CMOS circuitry · Low power Year, month, date, day-of-week, hours, minutes & seconds! 12/24 hour · Leap Year correction 4 interrupt rates available.
CLK-1 $60

I/O EXPANSION BOARD
For microcomputers that use 6522 VIAs for I/O and do not provide full address decoding on board. This board has physical space for four additional 6522 VIAs, and provides additional decoding for a total of 16 devices. Two versions of this board are available. The I/OX-122 mounts above, and directly plugs into, an on-board 6522 socket, and relocates the original VIA to the expansion board. Where there are space limitations, the I/OX-222 uses a dip header and an 8" cable for remote installation.
I/OX-122 $60 I/OX-222 $72

32K CMOS STATIC RAM BOARD



Models MB-132/32K, $299
/16K $241,/8K $197
Features:
- 200ns Low Power CMOS, STATIC RAM
- Extends your expansion connector
- Plug compatible with 2716 EPROMS
- First 8K are jumper selectable
- Entire board may be bank-switched

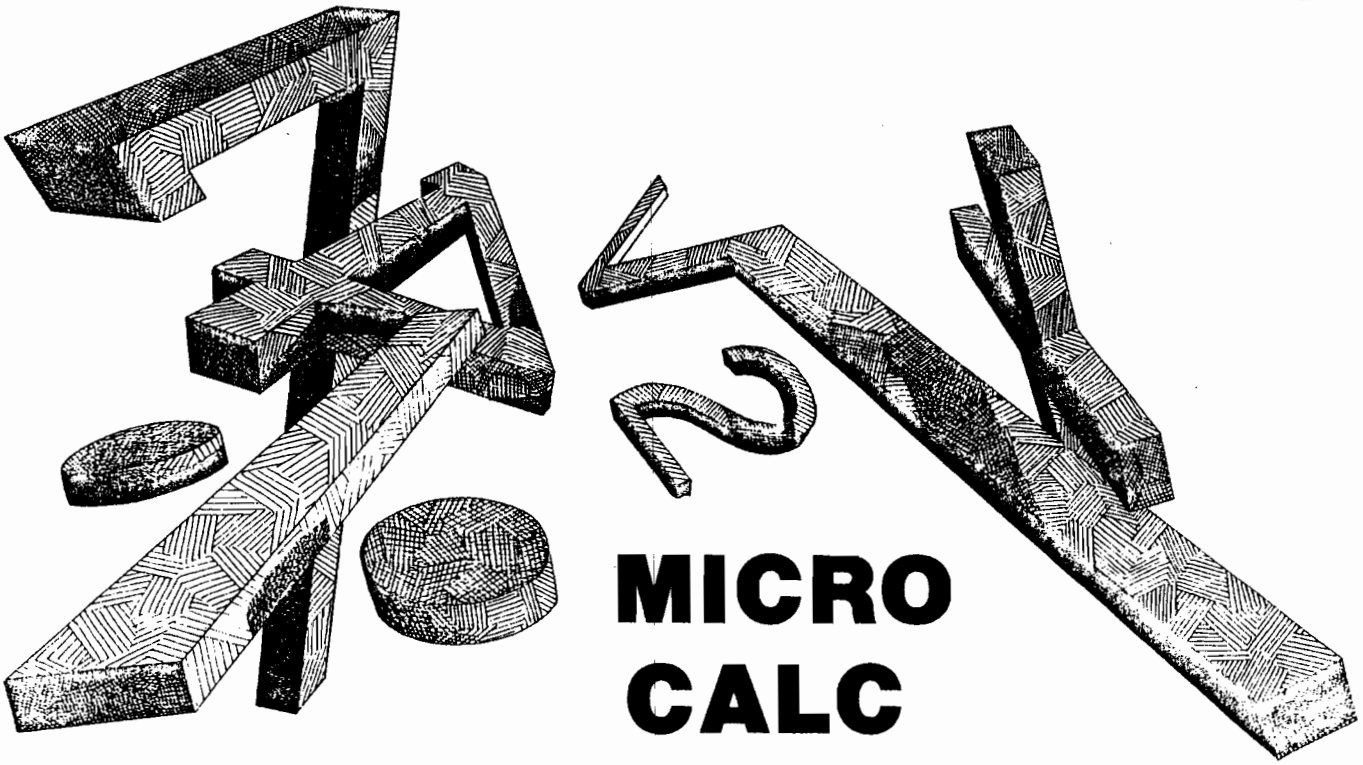
All boards feature G-10 glass/epoxy, solder mask, and gold plated connectors. All boards carry a full 1-year limited werranty. When ordering specify SYM or AIM version; add S & H in U.S. and Canada $2.50, overseas $4.00 (US). Calif. residents add sales tax.

ALTERNATIVE ENERGY PRODUCTS

Dealer and Distributor Inquires Invited.

P.O. Box 1019 • Whittier, CA 90609

(213) 941-1383



MICRO CALC

by Tom Marshall

Comments on Atari listings

Starting this month, our Atari listings are being output on the EPSON FX-80 printer. This printer allows redefining some or all of the Epson ROM character set. After much testing, we arrived at a compromise set of characters. Since many of the reversed characters would be difficult to read at the size of these listings, we thought that it would be clearer for the reader typing these programs into his computer to underline the reversed characters. The Atari programs that follow utilize this new style of listing. If anyone has any comments, pro or con, drop us a line with your viewpoint.

Features:

- 24 working lines
- 24 corresponding comment fields
- support of disk or tape files
- multiple statement support
- BASIC screen editing features

Operating Instructions

ESC START	performs calculations
ESC OPTION	enters file mode
ESC SELECT	enters comment field
Shift CLEAR	clears screen

Listing 1

```
  3 REM MCALC
  4 REM
  9 POKE 106,PEEK(106)-4:GRAPHICS 0
 10 DIM TBL0(255),ST0$(800),DX0(3),DY0(3),TEMP0$(200)
    ,UL0$(25),F0$(15),M0$(54),QM0(20),SPACE0$(40)
 20 GOSUB 30000:GOTO 300
100 GET #1,C0:TY0=TBL0(C0):IF  NOT (TY0) THEN 100      G
110 ON TY0 GOTO 120,130,200,140,160,150,170
120 POSITION X0,Y0:PRINT CHR$(C0);:P0=(Y0-1)*40+X0+1:
    ST0$(P0,P0)=CHR$(C0):X0=X0+1-LL0*(X0=MAX0):
    GOTO 100
130 X0=X0+DX0(C0-28):Y0=Y0+DY0(C0-28):
    IF X0<MAX0-LL0+1 THEN X0=MAX0                      B
132 IF X0>MAX0 THEN X0=MAX0-LL0+1
134 IF Y0<1 THEN Y0=20
136 IF Y0>20 THEN Y0=1
138 POSITION X0,Y0:PRINT "›‹";:GOTO 100
140 X0=X0-1+LL0*(X0=MAX0-LL0+1):POSITION X0,Y0:        L
    PRINT "_‹";:P0=(Y0-1)*40+X0+1:ST0$(P0,P0)=" ":
    GOTO 100
150 X0=MAX0-LL0+1:Y0=Y0+1-20*(Y0=20):POSITION X0,Y0:
    PRINT "›‹";:GOTO 100
160 POSITION X0,Y0:PRINT "?";:X0=X0+1-LL0*(X0=MAX0):
    IF X0<>MAX0 THEN PRINT UL0$(1,MAX0-X0+1);
162 P0=(Y0-1)*40+X0:ST0$(P0,P0)="?":QM0(Y0)=P0:
    POSITION X0,Y0:PRINT "›‹";:GOTO 100
170 POSITION 0,22:PRINT "▯   Clear screen:
    Are you sure...?";:GET #1,C0:
    IF CHR$(C0)<>"Y" THEN 300
172 FOR Q0=1 TO 20:POSITION 1,Q0:
    PRINT UL0$(1,11);"  ";UL0$(1,25):NEXT Q0
174 ST0$=" ":ST0$(800)=" ":ST0$(2)=ST0$:GOTO 300
```

(Continued on page 82)

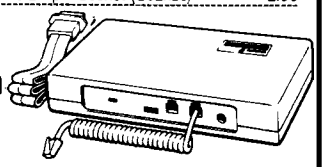
Hewlett Packard

Write or call for prices.

SPECIALS on INTREGATED CIRCUITS

6502	7.45	10/ 6.95	50/ 6.55	100/ 6.15
6502A/6512A	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		2.45	25/ 2.30	100/ 2.15
2716 EPROM		4.45	5/ 4.20	10/ 3.95
2532 EPROM		5.90	5/ 5.75	10/ 5.45
6116 2KX8 CMOS RAM		6.45	5/ 6.20	10/ 5.95
4116 RAM				8 for 14
4164 RAM				6.50
Zero Insertion Force 24 pin Socket (Scanbe)				2.00

Anchor Automation Signalman Modems



FREE SOURCE MEMBERSHIP WITH SIGNALMAN

All Signalman Modems are Direct Connect, and include cables to connect to your computer and to the telephone. Signalman Modems provide the best price-performance values.

Dealer and OEM inquiries invited

Volksmodem	**(79)**	**59**
RS232 or Atari Cable for Volksmodem		9
Mark VI for IBM Personal Computer	(279)	179
Mark VII Auto Dial/Auto Answer	(159)	99
Mark XII Smart Model 1200/300	(399)	299
OC HAYES Smartmodem		**219**
DC Hayes Smartmodem 1200/300		529

PROM QUEEN for VIC	**178**
Apple Emulator for Commodore 64	**Call**
STAT Statistics Package for C64	**95**
Solid Oak 2 Level Stand for C64 or VIC	**29**
C64/VIC Switch (networking)	125
BACKUP V1.0 tape copier for C64 or VIC	**20**
CARDBOARD/6 Motherboard - VIC	64
CARDBOARD/5 Motherboard - C64	56
CARD PRINT G Printer Int. with Graphics	72
CARD PRINT B Printer Interface—C64/VIC	40
CARDBOARD/3s Motherboard - VIC	22
CARDCO C64/VIC Calculator Keypad	32
CARDRAM/16 RAM Expansion - VIC	44
Complete CARDCO Line in stock	
CIE and VIE IEEE Interfaces in stock	
MSO SuperDrive for C64 or IEEE	**365**
MAE Assembler for C64	50

APPLE—FRANKLIN ITEMS

KRAFT Apple Joystick	**40**
Kraft Apple Paddle Pair	30
SPINNAKER Software in stock	
Broderbund Software in stock	
16K RAM Card for Apple	59
Multiplan—Microsoft	**185**
Solid Oak 2 Level Stand for Apple	**29**
Serial Card for Apple	99
MCP RAM/80 column card for IIe (AP/TXT)	**139**
Z80 Softcard and CP/M (Microsoft)	235
RANA Elite I with Controller	389
Parallel Printer Interface/Cable	69
Microtek Interfaces in stock	
Apple Dumpling with 16K Buffer	**160**
Grappler + Interface	129
Kraft Products for Apple in stock	
DC Hayes Micromodem II	299
PFS: File	**95**
PFS: Report	**95**
Videx 80 Column Card	209
Hayden Software for Apple 20% OFF	
Apple Blue Book	**19**

commodore

See us for Personal, Business ,and Educational requirements. Educational Discounts available.

PETSCAN I $245 base price

Allows you to connect up to 30 CBM/PET Computers to shared disk drives and printers. Completely transparent to the user. Perfect for schools or multiple word processing configurations. Base configuration supports 2 computers. Additional computer hookups $100 each.

COMPACK/STCP $115

Intelligent Terminal Package for PET, CBM, C64

Includes ACIA Hardware / STCP Software

VIC/C64 Switch —Networking	**125**
Allows 8 computers to share drive and printer.	
SCREENMAKER 80Column Adapter for C 64	**145**
Provides big screen capability for business applications.	
GENESIS Computer Corp	
VIController (for C64 as well)	**50**
combine with BSR modules for home or business control	
COMSENSE Remote Sensing Adapter for C64 or VIC	**35**
COM VOICE Synthesizer for C64 or VIC	**139**
includes software for text to speech, pitch ,etc.	
COM CLOCK Real Time Clock with battery backup	**45**

VIC 20 Products and Software in stock

Thorn EMI Software		**UMI Software**	
ABACUS Software		**HES Software**	
16K RAM for VIC	64	Vanilla Pilot	27

VICTORY Software for VIC and C 64

Street Sweepers (VIC)	12	Kongo Kong (VIC)	16
Night Rider (VIC)	11	Cosmic Debris (VIC)	12
Annihilator	16	Adventure Pack I	16
Adventure Pack II	16	Metamorphosis	11
Educational Pack I	11	Trek	12
Strategy Pack I	16	Grave Robbers	12

PAPER CLIP Word Processor	**109**
ORACLE Data Base from Batteries Included	**125**
Super BusCard from Batteries Included	**179**
Commodore 64 Programmers Reference Guide	16
MicroChess for C 64—8 levels of play	**19**
excellent graphics and color	
SPINNAKER Software C64, Apple ,IBM , Atari	
Compute!'s First Book of PET/CBM	11
C64 or VIC SWITCH	**125**
POWER ROM Utilities for PET/CBM	78
WordPro 3+/64 with Spellmaster	**85**
WordPro 4+ - 8032,disk, printer	279
SPELLMASTER spelling checker for WordPro	
VISICALC for PET, ATARI, or Apple	189
PET-TRAX PET to Epson Graphics Software	**40**
SM-KIT enhanced PET/CBM ROM Utilities	**40**
Programmers Toolkit - PET ROM Utilities	35
Calc Result Advanced—C64	**135**
EASY CALC for C64	**65**
PET Spacemaker II ROM Switch	36
COPYWRITER Word Processor for C64	**49**
2 Meter PET to IEEE or IEEE to IEEE Cable	40
Dust Cover for PET, CBM, 4040, or 8050	8
CmC Interfaces (ADA1800, ADA1450, SADI in stock)	
Programming the PET/CBM (Compute!) — R West	**20**
Compute! First Book of VIC	11
OMNICALC (HES)	**79**
HES MODEM with Software	**65**
HES Software and Hardware in stock	
UMI products in stock	

FlexFile for PET/CBM/ C64 $110

Database, Report Writer with calculations, Mailing Lists.

FORTH for PET/C64 full FIG model —Cargill/Riley	**$50**
Metacompiler for FORTH for independent object code	**30**
KMMM PASCAL IV.1 for PET/C64	**99**
EARL for PET/CBM Disk-based ASSEMBLER	**65**
Super Graphics — BASIC Language Extensions	**45**
Fast machine language graphics routines for PET/CBM	
RAM/ROM for PET/CBM 4K $75 8K $90	
Commodore Public Domain Software for C64	**75**

DISK SPECIALS

Scotch (3M) 5" ss/dd	10/ 2.20	50/ 2.00	100/ 1.95
Scotch (3M) 5" ds/dd	10/ 3.05	50/ 2.80	100/ 2.75
Scotch (3M) 8" ss/sd	10/ 2.30	50/ 2.10	100/ 2.06
Scotch (3M) 8" ss/dd	10/ 2.85	50/ 2.70	100/ 2.65

We stock VERBATIM DISKS

Write for Dealer and OEM prices.

Sentinal 5" ss/dd	10/ 1.90	50/ 1.85	100/ 1.80
Sentinal 5" ds/dd	10/ 2.55	50/ 2.50	100/ 2.45

We stock Dysan disks

Wabash 5" ss/sd	10/ 1.60	50/ 1.55	100/ 1.45
Wabash 5" ss/dd	10/ 1.90	50/ 1.85	100/ 1.75
Wabash 8" ss/sd	10/ 2.00	50/ 1.95	100/ 1.85

We stock MAXELL DISKS

Write for dealer and OEM prices.

Disk Storage Pages	10 for $5	Hub Rings 50 for $6	
Disk Library Cases	8"—3.00	5"—2.25	
Head Cleaning Kits	11		

CASSETTE TAPES—AGFA PE-611 PREMIUM

C-10	10/ .61	50/ .58	100/ .50
C-30	10/ .85	50/ .82	100/ .70

DATASHIELD BACKUP POWER SOURCE $265

Battery back up Uninterruptible Power Supply with surge and noise filtering. The answer to your power problems.

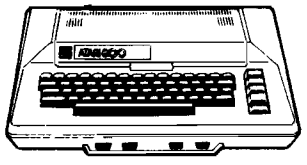
MultiPlan—IBM or Apple	**185**
Quadboard for IBM available	
Peachtext 5000 Software Package	**219**
PFS Software for IBM and Apple in stock	
VOTRAX Personal Speech System	280
BMC 9191 Color Monitor	**229**
BMC 12A 12" Green Monitor	85
Dynax (brother) DX-15 Daisy Wheel Printer	**469**
Itoh Prowriter Parallel Printer	379
Panasonic 1090 Printer with Correspondence Mode	
Daisywriter 2000 with 48K buffer	1050
Gemini 10X	**299**
EPSON, Okidata, other printers in stock	
USI CompuMOD 4 R F Modulator	29
We Stock AMDEK Monitors	
Amdek DXY-100 Plotter	590
A P Products	15% OFF
Watanabe Intelligent Plotter 990	6-pen 1290
BROOKS 6 Outlet Surge Suppressor/Noise Filter	**54**
Electrohome 1302-2 13" Hi-res RGB Monitor	335
Panasonic 12" Monitor (20 MHz) with audio	139
Synertek SYM-1 Microcomputer	**189**

ALL BOOK and SOFTWARE PRICES DISCOUNTED

USI Video Monitors—Green or AMBER 20 MHz hi-res.
Dealer and OEM inquiries invited

ZENITH data systems

ZVM-122A	109	ZVM-123G	99
ZVM-131	300	ZVM-135	510
HERO 1 Robot (factory assembled)			2145
Z29 Terminal (DEC and ADM compatible)			680
ZT-10 Intel. Terminal with Serial Port			340
Z100 16-bit/8-bit Systems			CALL
We stock entire Zenith line.			



ATARI SPECIALS

WE STOCK ENTIRE LINE—write for prices.
SPINNAKER and Broderbund Software in Stock.

215-822-7727
252 Bethlehem Pike
Colmar, PA 18915

A B Computers

WRITE FOR CATALOG. Add $1.50 per order for United Parcel. We pay balance of UPS surface shipping charges on all prepaid orders (add extra for mail, APO/FPO, air). Prices include cash discount. Regular prices slightly higher. Prices subject to change.

Listing 1 *(continued)*

```
 200 POKE 752,1:POSITION 0,22:
     PRINT "■    ..SELECT, OPTION, or START ..";:
     POKE 752,0
 210 CON0=PEEK(53279):IF CON0=7 THEN 210
 220 IF CON0=6 THEN 1000:REM START
 222 IF CON0=5 THEN 250:REM .SELECT
 224 IF CON0=3 THEN 2000:REM OPTION
 226 GOTO 210
 250 IF LL0=25 THEN LL0=11:MAX0=11:X0=1:GOTO 300
 260 LL0=25:MAX0=38:X0=14
 300 POSITION 0,22:POKE 752,1:
     IF MAX0=11 THEN PRINT "■    Editing comment
     field.":POKE 752,0:GOTO 320                          C
 310 PRINT "■    Editing expression field.":POKE 752,0
 320 POSITION X0,Y0:PRINT "↑↓";:GOTO 100
1000 QL0=PEEK(88):QM0=PEEK(89):POKE 752,1
1100 FOR Q0=1 TO 800 STEP 40
1102 GOSUB 1950:POSITION 13,INT(Q0/40):                   E
     PRINT " ↓↑◇↓↑ ";
1103 IF ST0$(Q0+14,Q0+38)=SPACE0$(1,25)
     THEN POSITION 14,INT(Q0/40)+1:PRINT UL0$(1,25):
     GOTO 1290
1104 FOR QQ0=38 TO 14 STEP -1:
     IF ST0$(Q0+QQ0,Q0+QQ0)=" " THEN NEXT QQ0:
     GOTO 1108                                            F
1105 IF QQ0=38 THEN 1108
1106 POSITION QQ0+1,INT(Q0/40)+1:PRINT UL0$(1,38-QQ0);
1108 GOSUB 1960:POKE START0,0:
     U0=USR(ADR(M0$),START0,START0+1,959)
1110 P0=QM0(INT(Q0/40)+1):IF ST0$(P0,P0)="?" THEN 1200
1120 POSITION 0,4:PRINT ST0$(Q0+14,Q0+38):? :? :
     PRINT "CONT"
1130 POSITION 0,0:TRAP 1900:POKE 842,13:STOP
1140 POKE 842,12:IF PEEK(START0+200)<>0 THEN 1990
1150 GOTO 1290
1200 POSITION 0,4:
     PRINT "A990=";ST0$(Q0+14,QM0(INT(Q0/40)+1)-1):? :
     ? :PRINT "CONT"
1210 POSITION 0,0:TRAP 1900:POKE 842,13:STOP
1215 POKE 842,12:IF PEEK(START0+200)<>0 THEN 1990
1220 GOSUB 1950:
     POSITION 39-LEN(STR$(A990)),INT(Q0/40)+1:
     PRINT A990;:GOSUB 1960
1290 NEXT Q0:POKE 752,0
1300 GOSUB 1950:POSITION 13,INT(Q0/40):PRINT " ";:
     POKE 752,0:GOTO 300
1900 GOSUB 1950
1902 POKE 842,12:POSITION 0,22:
     PRINT "■■   ?Error ";PEEK(195);" at line ";
     INT(Q0/40)+1;".";:POKE 752,0
1910 X0=MAX0-LL0+1:Y0=INT(Q0/40)+1:POSITION X0,Y0:
     PRINT "↑↓";:GOTO 100
1950 POKE 88,QL0:POKE 89,QM0:RETURN
1960 POKE 88,L0:POKE 89,M0:RETURN
1990 GOSUB 1950:POSITION 0,22:
     PRINT "■■   ?Syntax Error at line ";INT(Q0/40)+1;
     ".";:POKE 752,0:GOTO 1910
2000 POKE 752,1:POSITION 0,22:
     PRINT "■..(S) to save, (L) to load (E) to edit..";
     :POKE 752,0                                          H
2010 GET #1,C0:
     IF CHR$(C0)<>"S" AND CHR$(C0)<>"L" AND CHR$(C0)
     <>"E" THEN 2010
2030 IF CHR$(C0)="E" THEN 300
2050 POSITION 0,22:PRINT "■    Enter filename...";:
     INPUT F0$:TRAP 2900
2060 IF CHR$(C0)="S" THEN 2200
2100 ST0$=" ":ST0$(800)=" ":ST0$(2)=ST0$:ST0$=""
2110 OPEN #2,4,0,F0$
2120 FOR Q0=1 TO 4:INPUT #2,TEMP0$:
     ST0$(LEN(ST0$)+1)=TEMP0$:NEXT Q0
2122 FOR Q0=1 TO 20:INPUT #2,QM0:QM0(Q0)=QM0:NEXT Q0
2130 CLOSE #2:TRAP 65535
2150 FOR Q0=1 TO 800 STEP 40
2152 POSITION 1,INT(Q0/40)+1:FOR QQ0=11 TO 1 STEP -1:
     IF ST0$(Q0+QQ0,Q0+QQ0)=" " THEN NEXT QQ0:
     ? UL0$(1,11);:GOTO 2162
2154 PRINT ST0$(Q0+1,Q0+QQ0);:
     IF QQ0<11 THEN PRINT UL0$(1,11-QQ0);
2160 POSITION 14,INT(Q0/40)+1:FOR QQ0=38 TO 14 STEP -1:
     IF ST0$(Q0+QQ0,Q0+QQ0)=" " THEN NEXT QQ0:
     ? UL0$(1,25);:GOTO 2170
2162 PRINT ST0$(Q0+14,Q0+QQ0);:
     IF QQ0<38 THEN PRINT UL0$(1,38-QQ0)
2164 IF Q0/40=INT(Q0/40) THEN PRINT
2170 NEXT Q0
2180 LL0=25:MAX0=38:Y0=1:X0=14:GOTO 300
2200 OPEN #2,8,0,F0$
2210 FOR Q0=0 TO 3:PRINT #2;ST0$(Q0*200+1,Q0*200+200):
     NEXT Q0
2212 FOR Q0=1 TO 20:PRINT #2;QM0(Q0):NEXT Q0
2220 CLOSE #2:TRAP 65535:GOTO 300
2900 TRAP 65535:POSITION 0,22:
     PRINT "■■   ?File input/output error..."
2910 FOR Q0=1 TO 200:NEXT Q0:GOTO 2000
30000 REM ---INIT---
30010 M0=PEEK(106):L0=0:START0=256*M0
30020 FOR Q0=0 TO 26:TBL0(Q0)=0:NEXT Q0
30022 FOR Q0=32 TO 94:TBL0(Q0)=1:NEXT Q0                  A
30024 FOR Q0=95 TO 255:TBL0(Q0)=0:NEXT Q0
30026 FOR Q0=28 TO 31:TBL0(Q0)=2:NEXT Q0
30028 TBL0(27)=3:TBL0(126)=4:TBL0(63)=5:TBL0(155)=6:
     TBL0(125)=7
30030 LL0=25:MAX0=38:Y0=1:X0=14
30040 FOR Q0=0 TO 3:READ DX0,DY0:DX0(Q0)=DX0:
     DY0(Q0)=DY0:NEXT Q0
30042 READ M0$:POKE START0,0:                             K
     U0=USR(ADR(M0$),START0,START0+1,959)
30050 UL0$="                          "
30052 ST0$=" ":ST0$(800)=" ":ST0$(2)=ST0$
30054 SPACE0$="
     "
30060 FOR Q0=1 TO 20:QM0(Q0)=(Q0-1)*40+30:NEXT Q0
30080 OPEN #1,4,0,"K:"
30088 POKE 82,0:POKE 752,1:FOR Q0=1 TO 20:
     POSITION 0,Q0:PRINT " ";UL0$(1,11);"  ";UL0$:
     NEXT Q0
30090 POKE 752,0:POSITION X0,Y0:? "↑↓";:RETURN
32000 REM ---ARROW DISPLACEMENTS---
32010 DATA 0,-1,0,1,-1,0,1,0
32100 REM ---6502 MOVE (FILL)---
32110 DATA
     hh■Lh■Kh■Nh■Mh■Ph■0 ▯1K■MfKP ▯fLfMP ▯fNF0%0I■P ▯FP%0▯
PP■0
```

BREAK UP

By Phil Daley and Tom Marshall

A simple animation demonstration of PEEKing and POKEing screen memory

A Brick Wall Demonstration

Get ready to hit the bouncing ball with your bumper and knock out a few bricks. The farther away the brick is, the more points you will get for knocking it out. If you are dexterous enough to knock out the entire wall of bricks, don't get over confident, the game will continue with an even harder screen of bricks.

Breakup is a simple graphics display game that presents the principles of animation with player/missile graphics to move characters on the screen and test for collisions. It includes a "ball" that moves around the screen, rebounds from struck objects, and knocks out bricks in the walls of bricks. It also includes a player-controlled "bumper" to keep the ball from going out-of-bounds and being lost, a defined playing field with three walls from which to bounce the ball, and some eight rows of blocks, the amount of points received for hitting them dependent upon their color and distance from your bumper.

The game keeps score by color; 5 points for the green at the bottom, fifteen for the blue above it, and twenty for the yellow-orange blocks just above that. When you clear the entire screen, you are awarded an extra ball, the paddle shortens by one dot and moves closer to the blocks. This continues, screen after screen, until the bumper is as small and as close to the bricks as it can be. In addition, the points received for hitting the blocks are all increased by 3 points. That is, when you are playing the second screen, the green blocks at the bottom of the screen are eight points. If you manage to get to even the next screen, they will be worth eleven, and so on. Unlike the size of the bumper, the values for the bricks have no limit, and may increase in value for as long as you can play the game.

Operating Instructions

1. Key in 'BREAKUP' from the listing and save it on your tape or disk, and then RUN it.
2. First you are asked whether you will play from paddles or the keyboard. Choose the corresponding letter — P, or K.
3. The program will display the playing field, the brick walls, and your bumper. When you are ready to start play, press the button on the paddle, or the START key on the system console.
4. If you have chosen the keyboard use the cursor left arrow and cursor right arrow keys to move the bumper left and right. Holding the shift key at the same time increases the speed of the bumper.
5. If, for some reason, you halt program execution with the Break key, you must hit the SYSTEM/RESET key before re-RUNing. This will be further explained later.

The Program

The ball starts from a random position at the bottom of the screen and travels upwards, hitting a brick. This causes the brick to disappear, adds the appropriate amount of points to your score, and rebounds the ball towards the bottom. Here is the challenge: You must hit the ball back with your bumper to keep the ball from traveling out-of-bounds and off the screen, thereby losing the ball. If you are successful, the ball will simply hit another brick and bounce back. If you miss the ball, a buzzer will sound and the program halts until you hit the paddle or the START

button. You are allowed six balls total, plus an extra one for every screen you clear. Also, the angle and relative speed of the ball increase the closer you hit the ball to the ends of the bumper. Hitting the ball near the center of the bumper helps to restore the ball to a less radical angle.

Breakup's Animation: The Idea of Player-Missile Graphics

The animation in Breakup was done with the Atari's Player-Missle (PM) graphics capabilities. I used PM graphics because the speed of moving figures (players) around on the screen, such as the ball and paddle, is very fast. Also, PM graphics makes it very easy to test for collisions. This makes for a faster and more challenging game. In fact, even machine language versions of this game, which generally don't have to worry about speed due to the speed inherent in machine language programs, use PM graphics because of its ease of use.

A player is a zone on the screen that is eight pixels wide and extends vertically off both the top and bottom of the screen. A missile is generally a very thin player; it is only two pixels wide and likewise, extends past the top and bottom of the screen. There are several locations (registers) that correspond to the characteristics of each of the players and missiles, such as color, pixel width, priority, collision detection, and horizontal position of each. The reason the players and missiles are so relatively thin when compared to their height, stems from the fact that there is no vertical position register for them, only a horizontal position register. This means that in order to move a player vertically (as needed by the ball, for example), we have to physically redraw the player either higher or lower in memory. But before we deal any further with the describing locations of PM graphics, lets first uncover how the Atari handles PM graphics in the first place.

The Atari allows for four separate players on the screen and four missiles, or five players if you combine all four missiles and treat it like a player. There are in general, two types of players, those drawn in one line and those drawn in two line resolution. One line resolution is just that; the players are drawn out one scan line at a time. Two line resolution is simply drawing the players out two scan lines at a time. One line resolution, while it allows for better looking pictures, takes 2K of memory to store, while two line resolution takes 1K of memory to store. Each player in one line resolution takes 256 bytes to describe (one for each scan line from the very top of the screen to the bottom), and each player in two line resolution only takes up 128 bytes as each byte corresponds to two scan lines instead of simply one. Note that not only does the better resolution take up more room, but the memory used for it must start on a 2k boundary (the starting location must be divisible by 2048) while the two line resolution memory only has to start on a 1K boundary (starting location is divisible by 1024). Thus we have to be somewhat careful in our placement of the player-missile memory.

The Atari finds this memory through its base address register, which is location 54279. That is, location 54279 tells the Atari where to go to find out what the Players look like. But since the location is only one byte in size (it is only one location) it has to hold the *page number* of the PM memory. A single byte, which can hold any number from 0 to 255, will be able to address any one of the 256 pages in the Atari. The paging method is simply a way for the Atari to find its way around with only one byte telling it where to go.

That's the Way the Ball Bounces.

Another time saving feature of PM graphics is its collision detection capability. A collision occurs when any player or missile touches something other than the background. This capability allows the program to, with a single PEEK statement, find out if anything is hitting any one of the players or missiles, or if they are touching anything. This makes the whole checking routine for the ball and paddle collisions very fast.

The way this collision detection works is simple. There is a register for every possible PM collision. The Player to Playfield collisions register is the location that is read constantly to see if the ball (player 0) has hit something, so that the appropriate ball movement routine can be activated. Similarly, the player to player collision register is read to see if the paddle has hit the ball.

Combining a few things...

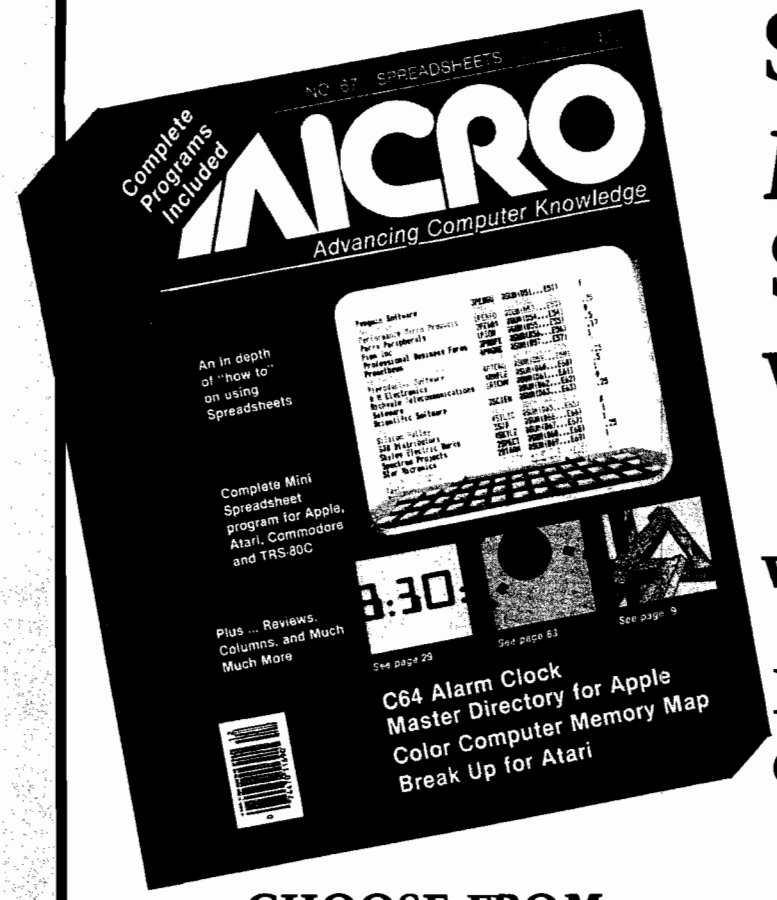
To make the colored bricks, we used redefined characters in graphics mode 2. We used characters simply because of the color capability, and ease of drawing and erasing. Characters in graphics mode 2 can be displayed as four different colors. So, we re-defined the character "$" to a 7 dot × 5 dot brick.

The first 384 bytes of memory (in double line mode) are always unused. And the first 512 bytes remains unused because this program did not enable the missiles (everything is done with the players.) This means that we have 512 bytes sitting there on a 1K boundary, doing nothing. This is perfect for a graphics 2 character set. By using this space for the somewhat altered character set, we can store something that would normally require 1½K (1K for the PM storage, and ½K for the character set) in only 1K.

The actual movement of the ball was calculated out in BASIC and executed in machine language. This is because, as mentioned earlier, PM graphics is great and quick for moving figures horizontally, but vertical movement must be done manually. BASIC proved to be much too slow for this. There are other ways around it, but having a machine language routine was the easiest.

Program Description

The routine to move the ball and the paddle, test for collisions, and do anything else involving animation is entirely contained in the lines 100 to 190. Note that this routine is almost at the very top of the program; all initialization and other routines are done below it. This is a programming trick to speed up the game, because the more lines that exist above a routine, the slower that routine will be. This has to do with the fact that when a GOTO is encountered, BASIC starts looking for the destination line number from the top and checks them all one by one until it finds where it has to go. This takes



Subscribe to MICRO... Save 20% and we'll send you a BONUS GIFT with your subscription!

Fill out the attached card and mail today!

CHOOSE FROM:

★ **Best of MICRO Vols. 2 and 3**
An $18.00 Value — FREE!

or

★ **MICRO Calc** for Apple, Atari, TRS-80, or C64
A $14.95 Value — FREE!

or

★ **What's Where in the Apple**
A $24.00 Value for Only $12.00

TWO NEW FROM

ALSO **Don't Miss**
These Great MICRO Publications

A Feast Of Computing Ideas

Use These Postage-Paid Cards To Order

SAVE 20%!

Use This Postage-Paid Card To Subscribe...

Or Call Toll-Free

1-800-345-8112

(In PA 1-800-662-2444)

(MasterCard and VISA Accepted)

MICRO™ SAVE 20%

☐ YES! Enter my subscription to MICRO for just $24.00! and with it send me the BONUS GIFT I've checked.

☐ **Best of MICRO Volumes 2 *and* 3.** An $18.00 Value — **FREE!**

☐ **MICRO Calc** for ☐ Apple ☐ Atari ☐ TRS-80 ☐ C64
A $14.95 Value — **FREE!**

☐ **What's Where in the Apple** at Half Price! A $24.00 Value — **Only $12.00!**

Please rush my subscription and the BONUS GIFT I've checked to:

Name _____

Address _____

City _____ State _____ Zip _____

(Allow 6-8 weeks for delivery)

I'm paying by: ☐ Check ☐ MO ☐ VISA ☐ MC

Total Enclosed: $ _____

Visa/MC # _____

Exp. Date: _____

BOOKS MICRO!

MICRO™'s Newest Books

NEW for VIC-20 Users!

Mastering Your VIC-20

With 8 BASIC Projects

A book that makes learning to program your VIC-20 fun! Contains 8 projects and programs. Games, utilities — even a VIC-20 version of "VisiCalc." All 8 programs on cassette to help you learn faster.

☐ Mastering the VIC-20 @ $19.95

NEW for OSI Users!

MICRO on the OSI

Includes Machine-Language enhancements and BASIC Aids, hardware modifications for enhanced/reversed video, programs for control code and upper/lower case entry. A valuable programming tool.

☐ MICRO on the OSI @ $19.95

Please rush the MICRO Books I've checked above to:

Name _____

Address _____

City _____ State _____ Zip _____

(Allow 6-8 weeks for delivery)

MA residents add 5% sales tax.

I'm paying by: ☐ Check ☐ MO ☐ VISA ☐ MC

Total Enclosed: $ _____
(Add $2.00 s/h per book.)

Visa/MC # _____

Exp. Date: _____

MICRO™'s Best Sellers

The Best of MICRO

Make your computer reference library complete by adding the "Best of MICRO" to your present book collection. Each volume full of articles and programs pulled from the pages of MICRO. Excellent software at a very low price!

☐ BOM Volume 2 @ $8.00
☐ BOM Volume 3 @ $10.00

What's Where in the Apple

Revised new addition with text added to the original atlas and gazetter. All Apple users will find this book helpful in understanding their machine and essential for mastering it.

☐ What's Where in the Apple @ $24.95
☐ The Guide @ $9.95

Best Sellers for APPLE Users!

MICRO on the APPLE

Programming aids, utilities, games, enhancements. Together Volumes 1, 2, and 3 contain over 100 programs on diskette. Fully documented and explained.

☐ 3-Volume Gift-Boxed @ $59.95
☐ Vol.1 ☐ Vol.2 ☐ Vol.3 $24.95 ea.

Please rush the MICRO Books I've checked above to:

Name _____

Address _____

City _____ State _____ Zip _____

(Allow 6-8 weeks for delivery)

MA residents add 5% sales tax.

I'm paying by: ☐ Check ☐ MO ☐ VISA ☐ MC

Total Enclosed: $ _____
(Add $2.00 s/h per book.)

Visa/MC # _____

Exp. Date: _____

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 20, AMHERST, NH

POSTAGE WILL BE PAID BY ADDRESSEE

MICRO™

10 Northern Blvd.
P.O. Box 6502
Amherst, NH 03031

SAVE 20%!!!!

Subscribe to MICRO™

Use This Postage Paid Card to Order the Next 12 Issues of MICRO and SAVE $6.00 Off Newsstand Price!

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 60, CHELMSFORD, MA

POSTAGE WILL BE PAID BY ADDRESSEE

MICRO™

34 Chelmsford Street
P.O. Box 6502
Chelmsford, MA 01824

A Feast Of Computing Ideas...

New Book From MICRO™

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 60, CHELMSFORD, MA

POSTAGE WILL BE PAID BY ADDRESSEE

MICRO™

34 Chelmsford Street
P.O. Box 6502
Chelmsford, MA 01824

Order These Best Sellers From MICRO™

time, and if you have a lot of lines above the routine, this will take a lot of time. Therefore all routines that are not time dependent, such as the initialization and score keeping routines, appear below the movement routine. In this way, no time is wasted during the movements.

Line 10 dimensions all the strings and arrays used by the program: M$ holds block move routine discussed in previous chapters, M2$ holds the ball movement routine, BALL$ holds the Player-Missile description for the ball (only twenty bytes worth), A holds the possible angles resulting from a collision with the bumper, P holds the points for each line of bricks on the screen, and PAD holds the descriptive byte that describes how the paddle looks from screen to screen. All of these will be covered a little better in a minute.

Line 20 calls the initialization routine at line 30000.

Line 30010 lowers the top of memory pointer by 1K (four pages) to make room for the player-missiles and new character set. Fortunately, location 106 points to a 4K boundary, so subtracting 1K from this location insures that the location will be on a 1K boundary (it will be divisible by 1024). The graphics 1 screen is initialized right after the pointer is moved, so that the computer can re-adjust the appropriate pointers to the new loss of 1K memory.

In line 30012, START is assigned the address of the new memory area, and the two machine language routines are loaded in.

Line 30014 pokes the starting location with a zero and propogates it through the entire 1K by moving 1023 bytes from the starting location to just the following location.

Line 30020 uses the Block Move routine to move the standard character set from ROM to the new memory allocated just before the PM memory area. This allows us to redefine the few characters we have to and keep the rest of them as they are.

Line 30030 and 30040 make players 2 and 3 into the left and right walls of the game. These walls could have been merely character, as is the top wall above the bricks, but they were made as players so that a single check could be made to determine if the ball should bounce horizontally or not.

Line 30044 puts the description of a 7 dot wide paddle into the player 1 area.

Lines 30050 to 30054 redefine the two characters whose internal value is 4 and 5 ("$", and "%" respectively), to the brick and solid block. The latter is used in the drawing of the wall on the top of the screen.

Lines 30060 to 30066 define the values of all the constants in the program. The majority of these are the locations for characteristic changes in the player missiles.

Line 30070 opens the keyboard for later input. It will remain open during the entire execution of the program.

Line 30080 sets up all the game values. (See the variable tables for details).

Lines 30082 to 30090 load in the values for the A, P and PAD arrays.

Lines 30100 to 30120 asks the user if he wants to play via paddles or the keyboard. CTRL holds the line number of the appropriate bumper routine.

The routine found at lines 30200 to 30260 initialize the screen. The PM graphics are enabled, the character set is

ROCKWELL Microcomputers from Excert, Inc.

• • SPECIALS • •

A65-1 (1K RAM) $435
A65-4 (4K RAM) $455
A65-4B,4F (4K, BASIC or FORTH*) . $495
A65-4AB (4K, BASIC & Assembler) . $525
A65/40-5000 (32K RAM) $1250

LANGUAGES for AIM-65® & AIM 65/40

Assembler $35
BASIC ROMs $65
FORTH* ROMs $65

ENCLOSURES & POWER SUPPLIES

A65-006 . . . . . $175
ENC4A $115
ENC5A $130
ENC6A $140

RM 65 SERIES
Deduct 5% from list if ordered with AIM 65® or AIM 65/40.

REPAIR SERVICE
(out of warranty only)
$25/hr. plus parts - $25 min.

SPARE PARTS
are available

CASH DISCOUNT -Deduct 5% for Prepaid Orders
(we pay shipping)

TERMS:
Net 30 from approved Companies & Institutions — otherwise COD.
Shipping will be added to order. Minnesota residents add 6% sales tax.
Prices subject to change without notice.

Educational Computer Division
EXCERT INCORPORATED
• SALES
• SERVICE
• INSTALLATION
• CONSULTING

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

Authorized Dealers for:
ROCKWELL INTERNATIONAL CORP.,
CUBIT, MTU, FORETHOUGHT PRODUCTS, GORDOS,
SEAWELL, DYNATEM, APPLIED BUSINESS COMPUTER

AIM-65 is a registered trademark of Rockwell International Corp.
*FORTH is a registered trademark of Forth, Inc.

enabled, and the bricks and walls are set up.

Line 50 pauses the game until either the paddle or START button is pressed. This gives the user time before the ball is released.

The entire game is controlled through lines 100 to 190.

In line 100, the horizontal and vertical displacements are added the the X and Y coordinates of the ball. Then the paddle is moved (CTRL is the line number of the appropriate routine). A machine language routine that moves the ball within the player is then called. This is what happens in the routine:

The routine is passed the following values:
x coordinate,
y coordinate,
the starting location of the ball description,
the start of Player 0 (where to put the ball),
and, how many bytes of the ball description to move.
Player 0 is moved horizontally (only one location to change)
Player 0 is moved vertically
The collision registers are cleared
The routine then waits for 1/60th of a second,
and then returns to BASIC

Clearing the collision registers is performed by the internal workings of the Atari whenever location 53278 is POKEd with any number. 1/60th of a second is waited out to allow the collisions to register.

Line 110 assigns the needed collision registers to the following variables: BPF (for the ball to character collisions), BPL (Ball to wall collisions), and PB (Paddle to ball collisions). Y is then checked to see if the ball has been missed.

Line 150 turns off any sound that may have turned on by some previous collision. BPF is then checked to see if it has hit playfield 0, 1 or 2 (one of the hittable bricks. If a collision has occurred, then control is passed to the brick routine at line 200.

Line 160: If the ball has hit playfield 3, then reflect (negate) the vertical displacement and make a sound.

Line 170: If the ball has hit either wall, then horizontally reflect it and make a sound.

Line 180: If the paddle has hit the ball then vertically reflect it. H is then assigned the appropriate angle of horizontal reflection. A sound is made.

Line 190 returns control back to line 100 in the event that none of the above has occurred.

Lines 200 to 210 handle the brick colliding routine:

Line 200 prints a space over the brick, effectively erasing it, adds the appropriate amount of points to the score, vertically reflects the ball, makes a sound, and subtracts one from the number of bricks variables (NB).

Line 202 prints the score. If NB is zero, then control is passed to the new screen routine.

Line 210 passes control back to the main loop.

The value of CTRL is set in the routine at 30100, and is either a 300 or a 400. CTRL is the line number of the appropriate bumper routine. If the game is controlled by the paddles then CTRL is 300, and if it is controlled by the keyboard, then CTRL is 400. Line 300 assigns the variable PP with the paddle position negated and moved to the right a little. The Paddle value was negated so that paddle movement would correspond to the bumper movement on the screen. Lines 400 to 420 move the paddle left or right one pixel depending upon whether the left or right arrow key was held down. If the shift key was held down then the paddle is moved in the direction specified by five pixels instead. This allows the paddle to speed up if it has too.

Lines 500 to 550 contain the missed ball routine. If the number of balls left is greater than zero then, the game values are re-initialized, the number of balls left is decremented by one, and the game resumes at line 50. If the number of balls is zero, then the game is over, and you are asked if you wish to try again. If you specify "N", then the top of memory pointer is reset to its original spot, and the program halts. If "Y" was specified, the top of memory pointer is reset, and the program is re-RUN. Note that if the program is stopped via the Break key, and rerun, the top of memory will be even lower than it was before. If this is continued, the computer will eventually run out of room and unrecoverably crash. It is for this reason, that whenever the program stops via the Break key, the user should hit SYSTEM/RESET.

Lines 600 to 690 handle the screen clear routine. If it can be done, the paddle is shorted by one pixel and moved up three lines. This is done at line 610 by block moving the description bytes for the paddle up one byte three times. Between each move upward, a sound is briefly made and a delay occurs, so that the changing of the paddle is more obvious. SZ is a flag telling the program that there is still room to move the paddle upward three lines and that the paddle can still be shortened. It is incremented every time the paddle is raised. If SZ ever reaches 7, then the paddle is no longer raised or shortened every time the screen is cleared. The points received for each brick struck is also increased by 3 for each consecutive screen. When this routine is done, the game values are re-initialized and the game resumes at line 50.

The DATA statements on lines 32010 and 32110 hold the two machine language routines in string form. These are read in to the appropriate strings during the initialization routine.

The rest of the DATA statements on lines 32210 and 32220, 32310, 32410, and 32510 hold the values for the new characters in the character set, the paddle angles, the points received for the blocks per line, and the paddle sizes per new screen, respectively. They are likewise read into their appropriate variables during the initialization routine.

MICRO™

You may contact the authors at MICRO Magazine, Box 6502, Amherst, New Hampshire 03031.

Listing 1

```
 10 DIM M$(54),M2$(99),BALL$(20),A(7),P(23),PAD(6)
 20 GOSUB 30000
 50 IF PTRIG(0) AND PEEK(53279)<>6 THEN 50
100 X=X+H:Y=Y+V:GOSUB CTRL:POKE P1,PP:
    U=USR(BALLXY,X,Y,BALL,STP0,14)
110 BPF=PEEK(P0PF):BPL=PEEK(P0PL):PB=PEEK(P1PL):
    IF Y>111 THEN 500
```

(Continued on next page)

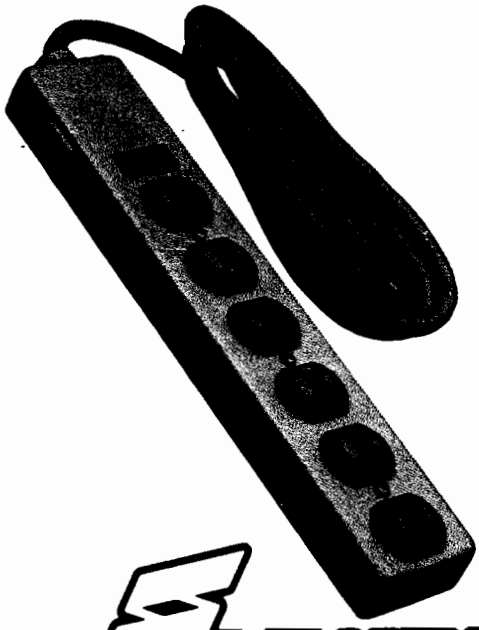
Listing 1 *(continued)*

```
150 SOUND 0,0,0,0:IF BPF>0 AND BPF<8 THEN 200
160 IF BPF>7 THEN V=-V:SOUND 0,80,10,10
170 IF BPL>3 THEN H=-H:SOUND 0,80,10,10
180 IF PB/2<>INT(PB/2) THEN V=-V:
    H=A(X-PP+1)*(BPL<=3)+H*(BPL>3):
    SOUND 0,50,10,10:GOTO 100
190 GOTO 100
200 RY=INT((Y-16)/4):POSITION INT((X-48)/8),RY:
    ? #6;" ";:SC=SC+P(RY):V=-V:SOUND 0,100,10,10:
    NB=NB-1
202 POSITION 15,0:PRINT #6;SC:IF NB=0 THEN 600
210 GOTO 100
300 PP=250-PADDLE(0):RETURN
400 I=1:P=PEEK(764):P=P-64*(P>64):P9=PEEK(53775):
    IF P9<248 THEN I=4
410 IF P9<>255 THEN PP=PP-I:IF P=7 THEN PP=PP+2*I
420 RETURN
500 POSITION 5,0:PRINT #6;BL:
    IF BL>0 THEN SOUND 0,200,12,14:FOR Q=1 TO 100:
    NEXT Q:SOUND 0,0,0,0:GOTO 550
502 FOR Q=200 TO 100 STEP -2:SOUND 0,Q,10,10:
    SOUND 1,300-Q,10,10:NEXT Q
504 FOR Q=1 TO 100:NEXT Q:SOUND 0,0,0,0:
    SOUND 1,0,0,0
510 POSITION 0,5:
    PRINT #6;
    "                    TRY again (Y/N)
    ?                    "
520 GET #1,C:
    IF CHR$(C)<>"Y" AND CHR$(C)<>"N" THEN 520
522 IF CHR$(C)="Y" THEN POKE 106,PEEK(106)+4:
    GRAPHICS 1:POKE GRACTL,0:RUN
530 CLOSE #1:POKE 106,PEEK(106)+4:GRAPHICS 0:
    POKE GRACTL,0:END
550 BL=BL-1:POSITION 5,0:PRINT #6;BL:
    X=INT(144*RND(0)+56):Y=111:H=+2:V=-2:PP=124:
    GOTO 50
600 U=USR(BMOVE,START+512,START+513,127)
602 FOR Q=200 TO 0 STEP -5:SOUND 0,Q,10,14:
    SOUND 0,Q/2,10,10:NEXT Q:SOUND 0,0,0,0:
    IF PY=82 THEN PY=85:ZERO=1
610 FOR Q=1 TO 3:
    U=USR(BMOVE,START+641,START+640+ZERO,127)
612 SOUND 0,30,8,14:FOR QQ=1 TO 20:NEXT QQ:
    SOUND 0,0,0,0:FOR QQ=1 TO 20:NEXT QQ:NEXT Q
620 SZ=SZ+1:IF SZ=7 THEN SZ=6
630 POKE COLP1,15:SOUND 0,200,10,14:PY=PY-3:
    POKE START+640+PY,PAD(SZ):SOUND 0,0,0,0:
    POKE COLP1,78
634 FOR Q=0 TO 23:IF P(Q)>0 THEN P(Q)=P(Q)+3
636 NEXT Q
640 BL=BL+1:NB=144:GOSUB 30200:X=INT(144*RND(0)+56):
    Y=111:H=-2:V=-2:PP=124
690 GOTO 50
30000 REM ---INITIALIZATION---
30010 POKE 106,PEEK(106)-4:GRAPHICS 17
30012 START=256*PEEK(106):READ M$,M2$:BMOVE=ADR(M$):
      BALLXY=ADR(M2$)
30014 POKE START,0:U=USR(BMOVE,START,START+1,1023)
30020 U=USR(BMOVE,256*PEEK(756),START,512)
30030 POKE START+788,255:
      U=USR(BMOVE,START+788,START+789,91):REM L WALL
30040 POKE START+916,255:
      U=USR(BMOVE,START+916,START+917,91):REM R WALL
30044 POKE START+740,254:REM PADDLE
30050 FOR CN=4 TO 5
30052 FOR Q=CN*8 TO CN*8+7:READ D:POKE START+Q,D:
      NEXT Q
30054 NEXT CN:REM NEW CHARS
30060 P0=53248:P1=53249:P2=53250:P3=53251:
      POPF=53252:P0PL=53260:P1PL=53261:HITCLR=53278:
      DMACTL=559:GRACTL=53277
30062 SIZEP0=53256:SIZEP1=53257:SIZEP2=53258:
      SIZEP3=53259:COLP0=704:COLP1=705:COLP2=706:
      COLP3=707
30064 PMBASE=54279:CHBASE=756:STP0=START+512-6
30066 BALL$="◆◆◆◆◆◆@@◆◆◆◆◆◆◆":BALL=ADR(BALL$)
30070 OPEN #1,4,0,"K:"
30080 X=INT(144*RND(0)+56):Y=111:H=+2:V=-2:BL=5:
      NB=144:PY=100:PP=124
30082 FOR Q=0 TO 7:READ D:A(Q)=D*2:NEXT Q:
      REM PADDLE ANGLES
30084 FOR Q=0 TO 23:READ D:P(Q)=D:NEXT Q:
      REM POINT VALUES
30090 FOR Q=0 TO 6:READ D:PAD(Q)=D:NEXT Q:
      REM PADDLE SIZES
30100 POSITION 0,5:PRINT #6;" pADDLES OR kEYBOARD";
30110 GET #1,C:
      IF CHR$(C)<>"P" AND CHR$(C)<>"K" THEN 30110
30120 CTRL=400:IF CHR$(C)="P" THEN CTRL=300
30200 POSITION 0,0:POKE PMBASE,PEEK(106):
      POKE CHBASE,PEEK(106)
30210 POSITION 0,1:PRINT #6;"■■■■■■■■■■■■■■■■■■■■■"
30212 PRINT #6;
      " $$$$$$$$$$$$$$$$$  $$$$$$$$$$$$$$$$$
      $$$$$$$$$$$$$$$$$  $$$$$$$$$$$$$$$$$ "
30220 PRINT #6;
      " ▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌  ▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌
      ▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌  ▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌"
30230 POKE P2,48:POKE P3,201:POKE COLP0,14:
      POKE COLP1,78:POKE COLP2,70:POKE COLP3,70
30240 POKE SIZEP0,0:POKE SIZEP1,0:POKE SIZEP2,0:
      POKE SIZEP3,0
30250 POKE DMACTL,42:POKE GRACTL,2
30260 POSITION 5,0:PRINT #6;BL:POSITION 15,0:
      PRINT #6;SC
30900 RETURN
32000 REM ---BLOCK MOVE ROUTINE---
32010 DATA
      hh■Lh■Kh■Nh■Mh■Ph■0 ◆1K■MfKP HLfMP HNFOXOI■P FP
      X0¬PP0◆
32100 REM ---BALL MOVE ROUTINE---
32110 DATA
      hhhg◆Phh■Nh■Lh■Kh■Nh⁴eM■MZNi◆■Nhh(■fiK■M@◆Pw■<P-
      "TI◆Py-"TI,Py◆
32200 REM ---NEW CHARS ($,%)---
32210 DATA 0,0,0,127,127,127,127,127
32220 DATA 255,255,255,255,255,255,255,255
32300 REM ---PADDLE ANGLES---
32310 DATA -2,-1.5,-1,-.5,.5,1,1.5,2
32400 REM ---POINTS PER LINE---
32410 DATA 0,0,0,20,20,15,15,0,5,5,5,5,0,0,0,0,0,0,
      0,0,0,0,0,0
32500 REM ---PADDLE SIZES---
32510 DATA 0,126,124,60,56,24,16
```

MICRO

another FLEXIDUCT® OFFICE SAFETY PRODUCT

When you least expect it, ZAP!!!



In a few millionths of a second, common electrical surges and spikes can enter your data processing equipment and cause memory loss, false logic and misregistration. Surges very often do permanent damage to microcircuitry.

FLEXIDUCT Surge Suppressors catch surges and spikes before they have a chance to enter your equipment. In billionths of a second (Nanoseconds), **FLEXIDUCT** Surge Suppressors dissipate surges and spikes from any side of the line (most protect only one side).

The multi-outlet unit is ideally suited to the computerized work-station. It gives you the advantage of circuit breaker protection with plenty of outlets for data processor and peripheral equipment.

No computer should be without the protection of a **FLEXIDUCT** Surge Suppressor...**especially yours!** Write or call for further information. Available from office products retailers.

FLEXIDUCT® Surge Suppressors

a product of Winders & Geist, Inc. P.O. Box 83088 Lincoln, NE 68501 402/474-3400

MICRO™

From Here To Atari

by Paul S. Swanson

The listings accompanying this column provide my Christmas greetings to you. The assembly code is for reference. The BASIC program contains the resulting machine language in the data statements. It is an example of using display list interrupts. Enter the BASIC listing to see a color display.

Several display list interrupts control the changing colors in the triangular "tree" in the display and two more are used to color the trunk and to change the text window background to black. Every line of the mode 5 triangle has an interrupt on it. The colors are rotated under the control of the BASIC program.

The BASIC program begins by drawing the tree using dots of random colors. The background color is the only one not used in that section. Later, the display list interrupt will constantly alter the contents of the referenced color registers. The trunk is drawn with the color from register 2, which is declared in BASIC with COLOR 3. This is the same color register used for the text background.

Establishing the points for the interrupts is done in lines 120 through 160. First, DL is set equal to the location of the display list. Next, all of the bytes controlling lines 1 through 31 are altered. The 138 used is the code for GRAPHICS 5, which is 10, plus 128, which sets the display list interrupt enable bit. The tree occupies screen lines 1 through 30 and screen line 31 is the first line on the trunk. The line before the text window gets the last interrupt, which will be used to set the text background to black.

The display list interrupt is read into page six in lines 170 through 190. The vector is set up to point to the routine in line 200 and the first statement in line 210 enables the interrupt. Q is used to control the color base for the interrupt routine and A controls whether the message is "MERRY CHRISTMAS" or "HAPPY NEW YEAR."

The BASIC loop that occupies lines 220 through 280 alters the color base and prints the messages. The interrupt is going constantly, so the BASIC program does not need to call anything. All that it changes is the contents of location 1664, which is used by the interrupt as the color base. The two phrases, controlled by A, are printed using the loop at lines 230 through 260. The FOR/NEXT loop within that loop controls the timing for printing the individual letters. Lines 270 and 280 dorm a delay at the end of each phrase, then set up A to point to the other phrase.

The assembler routine starts by saving the three registers on the stack. Since it is interrupting the program and it will use these three registers, they must be saved. Otherwise, the program that is interrupted will have the wrong values in the registers when the interrupt returns.

VCOUNT contains the number of the current screen scan line divided by two. This will serve to divide the interrupt into three parts. The triangle shape requires a color rotation, the trunk requires that register 2 be set to brown and the text window requires that register 2 be set to black. VCOUNT is 79 at the interrupt where the trunk color is to be selected, so it is compared to 79. If it is found to be 79, a branch is made to STUMP, where brown is stored in register 2. If it is found to be greater than 79, a branch is made to WINDOW, where register 2 is set to black.

If it is neither equal to nor greater than 79, the color rotation is performed. Since the interrupt happens during the last scan line of the previous mode line, a STA WSYNC, which stops the processor until the end of the current scan line, must be performed before the registers are changed. Preparation for this involves placing the proper colors into the 6502 registers.

The colors are based on whatever is in location 1664, which is controlled by the BASIC program. This is added to VCOUNT and placed in register Y. For register X, $15 is added to the color and $2A is added for register A.

The STA WSYNC is performed next, immediately followed by the three statements that place the colors in the registers. Although the timing is not critical in this program, because the colors affected are not near the left edge of the screen, the placing of the colors takes place totally within the horizontal blank period. The three store commands require 12 machine cycles and there are 26 in the horizontal blank period, although a few of these are stolen by DMA.

Since the color changes are not critical for the trunk and the text window, WSYNC is ignored and the colors are stored directly into the color registers. The $26 is equivalent to SETCOLOR 2,2,6, which is the brown used for the tree trunk. In WINDOW, the background is set to black and the text (register one) is set to a medium white.

The EXIT routine must restore the three registers in the reverse of the order in which they were stored on the stack. After restoring the three registers, the interrupt mask (processor I bit) is cleared and the return from the interrupt is performed. The processor I bit is set when the interrupt is called and leaving it set prevents other interrupts from altering the timing in this interrupt.

POKEY Timers

Another interesting set of interrupts are controlled by POKEY, which is the device responsible for the sounds and operation of the serial I/O bus. There are three POKEY timer interrupts available for general program use, referred to as POKEY timers 1, 2 and 4. These use the values in the AUDF registers, which are the same ones used for generation of sounds.

The advantage to the POKEY timers over the display list or vertical blank interrupts is that they are controlled through independent counters. Display list and vertical blank interrupts depend on the 60 Hz television frame rate and cycle at that frequency. The POKEY interrupts are completely controlled by frequencies which can be set by software.

To get an exact frequency with a POKEY timer is not that easy unless the frequency you want is an even multiple of the clock rates. There are three clock rates available, just as there are for the sound channels. In fact, they are

the same sources. The "normal" frequency, which is the one selected when the system is booted, is 63.9210 KHz. This may be changed to count at 1.78979 MHz or 15.6999 KHz. When the interrupt routine is enabled properly, an interrupt happens each time the counter reaches zero. The frequency set for the clock rate can be used to calculate the frequency of the interrupt. The interrupt frequency is equal to:

> the clock frequency /(2 * (1 + number in the AUDF register))

POKE the value N into the register and the frequency of the interrupt is the frequency set (the 64 KHz or 15.7 KHz)/(2*(N1)). For 1.79 MHz, there is a slight modification of the formula. Divide the 1.79 MHz by two times the sum of N plus 4. If you are clocking two channels together, use 7 instead of the four. If you don't know what that means, use 4.

When you use the timer interrupts, pay close attention to what is on the system stack. Before jumping through the timer interrupt vector, the operating system pushes the A register onto the stack. Before your routine starts, you should push the X and/or Y registers onto the stack if you intend to use them. Before you return from the interrupt, pull X and/or Y registers off the stack, then PLA and clear the interrupt with CLI. If all that was pushed onto the stack is not pulled off, or if more is pulled off than was put on, the system will crash or at least lock up as soon as you enable the interrupt.

The method for implementing the POKEY timers is stated inaccurately in the manuals. If you set it up the way the manual states, your system will lock up and you will have to press SYSTEM RESET to continue. Instead, first set up AUDCTL ($D208, or 53768) with zero for 64 KHz, one for 15 KHz or 96 for 1.79 MHz. Next, set the volume (AUDC1, 2 or 4 at $D201, $D202 or $D204 53761, 53762 or 53764). Now you can set up your software interrupt routine and change the interrupt vector to point to it. The three vectors are at $0210, $0212 and $0214 (528, 530 and 532). The above steps can be in any order. After all of them are completed, start the timer by POKE 53769,0 (actually, any number from 0 to 255 can be POKEd here and you will get the same effect). After all that is done (not before the POKE 53769, which is what the manual states), enable the timer interrupt.

Enabling the interrupt involves PEEK(16). Add 1, 2 or 4 to that value, which corresponds to the interrupt you are using, and POKE the number back into location 16 and also into location 53774. Once you do that, your interrupt will begin and an interrupt will be generated when the timer you set counts down to zero. As soon as the interrupt happens, the timer is automatically loaded with the value you origionally POKEd there, so the process repeats until you disable it.

Problems to Watch For

Problems associated with POKEY timer interrupts involve timing and other interrupts. DMA can alter, unpredictably, the amount of time between the interrupt and the first action taken by your interrupt routine, making the timing a little less precice. The average over several interrupts will be at your selected frequency, but the timing between two consecutive actions may be off by a few clock cycles if DMA is not disabled.

Other interrupts can also introduce problems. The major problem is the vertical blank interrupt. The only solution to this is to turn off the interrupt, and the display list interrupts if any are enabled, by POKEing a zero to location 559. Make sure you do all your SETCOLOR, GRAPHICS and other statements that depend on shadowing first or resort to using the hardware registers. That POKE also turns off the real time clock and keyboard auto-repeat.

Another interrupt source is the IRQ interrupts. These can be masked out by setting the corresponding bits in locations 16 and 53774 to zero, storing only the 1, 2 or 4 for the POKEY interrupt in those locations. Another possibility is to SEI at the beginning of the interrupt (don't forget CLI at the end).

If you do not disable the keyboard, you may get some additional delays on some of the interrupts. The keyboard click uses the STA WSYNC command, which stops all processing, including interrupt servicing, until the end of the current television scan line is complete. Also, any other interrupt that leaves the processor "I" bit set will cause the processor to ignore the interrupt. Peripheral access may do this.

If you set up the POKEY timers to do something for you, you may have few, if any, problems with them. The problems mentioned above can be used as places to check if the timing is found to be innaccurate. If you set everything up the way I have described and the system locks up when you enable the interrupt, your machine language may have a fatal error. If you find no error, turn the computer off then on to reboot and try it again.

One undocumented note on the POKEY timers is that you can change the frequency between interrupts. If the calculations for the desired frequency are not exactly what you want, maintain a counter somewhere in memory. At the beginning of the routine, use STA to put the value into AUDF that BASIC placed there. Increment the counter and test it to see if it counted to where you want to make an adjustment. If it is there, store the adjustment frequency into AUDF and reset the counter to zero. The next interrupt will obey the new frequency, then put the old frequency back into AUDF. Remember that this is not documented, so it may not work on all Atari computers. Test it out before you depend on it.

Enabling the POKEY timer interrupts involves a lot of calculation. However, if they are properly enabled, very precise timing can be done with them. I am preparing a project using those timers and I will be writing an article describing it completely.

Next Month

I recently acquired an ATR8000, which is a device containing a Z80 processor, memory, a printer port, an RS-232-C port and disk controller logic so that you can hook up "bare" disk drives to your Atari. The ATR8000 offers CP/M compatibility and, when the CP/M option is not in use, the ATR8000 will act as a printer buffer. A functional description of the ATR 8000, along with pricing, will be featured in next month's From Here to Atari.

You may contact Paul at 97 Jackson St., Cambridge, MA 02140.

Listing 1

```
                00005 * Listing 1
                00010 *
                00020 * DLI ROUTINE
                00030 *
                00040 * EQUATES
                00050 *
D40B:           00060 VCOUNT   .EQ $D40B * SCAN LINE COUNTER
D016:           00070 COLOR0   .EQ $D016 * FOR SE.0,
D017:           00080 COLOR1   .EQ $D017 * FOR SE.1,
D018:           00090 COLOR2   .EQ $D018 * FOR SE.2,
D40A:           00100 WSYNC    .EQ $D40A * WAITS FOR HBLANK
                00110 *
                00120 * INTERRUPT ROUTINE
                00130 *
                00140          .OR $600 * FOR PAGE 6
0600: 48        00150 DLIROUT  PHA  * SAVE REGISTERS *
0601: 98        00160          TYA
0602: 48        00170          PHA
0603: 8A        00180          TXA
0604: 48        00190          PHA
0605: AD 0B D4  00200          LDA VCOUNT * CHECK
0608: C9 4F     00210          CMP #79     * SCAN
060A: F0 1B     00220          BEQ STUMP    * LINE
060C: 10 21     00230          BPL WINDOW
060E: 18        00240          CLC
060F: 6D 80 06  00250          ADC 1664 * ADD COLORBASE
0612: A8        00260          TAY
0613: 69 15     00270          ADC #$15
0615: AA        00280          TAX
0616: 69 2A     00290          ADC #$2A
0618: 8D 0A D4  00300          STA WSYNC * WAIT FOR BLANK
061B: 8D 18 D0  00310          STA COLOR2 * STORE COLORS
061E: 8C 16 D0  00320          STY COLOR0
0621: 8E 17 D0  00330          STX COLOR1
0624: 4C 39 06  00340          JMP EXIT
0627: A9 26     00350 STUMP    LDA #$26 * BROWN TRUNK
0629: 8D 18 D0  00360          STA COLOR2
062C: 4C 39 06  00370          JMP EXIT
062F: A9 00     00380 WINDOW   LDA #0 * BLACK BACKGROUND
0631: 8D 18 D0  00390          STA COLOR2
0634: A9 0A     00400          LDA #10 * WHITE LETTERS
0636: 8D 17 D0  00410          STA COLOR1
0639: 68        00420 EXIT     PLA  * RESTORE REGISTERS
063A: AA        00430          TAX
063B: 68        00440          PLA
063C: A8        00450          TAY
063D: 68        00460          PLA
063E: 58        00470          CLI  * CLEAR INTERRUPT
063F: 40        00480          RTI   * AND RETURN
```

Listing 2

```
 5 REM *** Listing 2 - Christmas Greetings
 6 REM *** Program by Paul S. Swanson
 7 REM ***
10 GRAPHICS 5
20 FOR I=0 TO 30:REM * DRAW TREE *
30 FOR J=40-I/2 TO 40+I/2
40 COLOR INT(RND(0)*3+1)
50 PLOT J,I+1
60 NEXT J:NEXT I
70 COLOR 3
80 FOR I=36 TO 44:REM * DRAW TRUNK *
90 PLOT I,32
100 DRAWTO I,38
110 NEXT I
120 DL=PEEK(560)+PEEK(561)*256:
    REM * LOCATE DISPLAY LIST *
130 FOR I=DL+6 TO DL+36:REM * SET INTERRUPTS *
140 POKE I,138
150 NEXT I
160 POKE DL+44,138
170 LOC=1536:RESTORE :REM * READ DLI ROUTINE *
180 READ N
190 IF N<>256 THEN POKE LOC,N:LOC=LOC+1:GOTO 180
200 POKE 512,0:POKE 513,6:REM * ENABLE DLI *
210 POKE 54286,192:Q=255:A=0
220 RESTORE A+2000:? CHR$(125):
    REM * PRINT MESSAGE *
230 READ N:IF N=256 THEN 270
240 ? CHR$(N);" ";:POKE 1664,Q:Q=Q-1:
    IF Q<0 THEN Q=255:REM * Q CONTROLS COLORS *
250 FOR I=1 TO 50:NEXT I:
    REM * DELAY BETWEEN LETTERS *
260 GOTO 230
270 FOR I=1 TO 300:NEXT I:
    REM * DELAY BETWEEN MESSAGES *
280 A=1000-A:GOTO 220
999 REM * DLI ROUTINE IN DECIMAL *
1000 DATA 120,72,152,72,138,72,173,11,212,201,79,
     240,26,16,32,109,128,6,168,105
1010 DATA 21,170,105,42,141,10,212,141,24,208,140,
     22,208,142,23,208,76,57,6,169
1020 DATA 38,141,24,208,76,57,6,169,0,141,24,208,
     169,10,141,23,208,104,170,104,168,104,88,64,
     256
1999 REM * MESSAGE #1 *
2000 DATA 32,77,69,82,82,89,32,67,72,82,73,83,84,77,
     65,83,256
2999 REM * MESSAGE #2 *
3000 DATA 32,72,65,80,80,89,32,78,69,87,32,89,69,65,
     82,256
```

MICRO

Computer Software Associates, an established software house, supplies software for COMMODORE, APPLE, and IBM home computers. CSA is currently looking to expand its product base of serious programs in the areas of home, education, and office.

Computer Software Associates offers over 30 years of market experience, worldwide distribution, innovative packaging, generous royalties, and programmer support.

Interested programmers should contact Mike Mahan, Product Development Department, Computer Software Associates, 50 Teed Drive, Randolph, Massachusetts 02368. Tel. (617) 961-5700.

COMPUTER SOFTWARE ASSOCIATES

MICRO™ Atari Reviews

Product Name: **Square Pairs**
Equip. req'd: Atari Computer, BASIC, Cassette Player
Price: ?
Manufacturer: Scholastic Inc.
906 Sylvan Ave.
P.O. Box 2010
Englewood, NJ 07632

Description: A game of matching. Square Pairs allows up to four players to take turns uncovering two boxes at a time and finding two that match.

Pluses: Even though there are three games included, the most interesting part of the program is the ability to make up your own games. After making up a game it can be saved on tape. This allows for more game variety.

Minuses: The program is only available on tape and will only interact with a Atari cassette recorder.

Documentation: The sixteen page manual is clearly written. Most of it is applied towards making up and saving games.

Skill level required: Designed for seven through twelve years of age. May not have much attraction to those over ten.

Reviewer: Richard E. DeVore

Product Name: **Turtle Tracks**
Equip. req'd: Atari Computer w/32K, BASIC, Disk Drive, Atari Printer optional
Price: ?
Manufacturer: Scholastic Inc.
906 Sylvan Ave.
P.O. Box 2010
Englewood Cliffs, NJ 07632

Description: *Turtle Tracks* is an interesting method for children to learn the fundamentals of programming. By typing in simple programs, they are shown, by a "turtle" drawing on the screen, exactly what their program does.

Pluses: The self-booting program is well error-trapped, allowing mistakes to be made without crashing. It allows loops and variables, demonstrating on the screen what they do. There is also a small segment on sound with enough information to let you compose simple music. If there is an Atari printer connected to the system, a print out of the screen may be made by just pressing the OPTION key.

Minuses: The program is slow in accepting keyboard input. The longer a program gets, the slower the cursor responds. Program execution is also slow.

Documentation: The eighty page manual is quite well done. It carries you from loading the program through saving and reloading your work. One of the clearest for children I have seen.

Skill level required: Beginner, recommended minimum starting age is nine years old.

Reviewer: Richard E. Devore

Product Name: **MMG Data Manager**
Equip. req'd: Atari Computer w/48K, BASIC, Disk Drive; printer optional
Price: $49.95
Manufacturer: MMG MICRO SOFTWARE
Manalapan 1000 Office Building
1000 Route 9
Englishtown, NJ 07726

Description: *MMG Data Manager* is a file management program for any Atari computer with sufficient memory and a disk drive. You may set up your records and fields to suit your particular needs. Although a BASIC program, it has machine language routines and is quite fast in use.

Pluses: The 26 page manual that comes with the program is quite clear and takes you step by step through the use of its features. The program is menu driven and simple enough to use that the manual may not be necessary after the first or second time a record is set up. The sort routine is extremely fast and may be used on up to three levels. The program is well error trapped, making use by the beginning computerist easy.

Minuses: There can be a maximum of only ten fields. The fields do not support computations. A record, once set up cannot be reconfigured. These minuses are features usually found in much more expensive programs and are not often available in this price range.

Documentation: The manual is easy to use and understand. It is a tutorial showing how each program function is used.

Skill level required: Beginning level user.

Reviewer: Richard E. DeVore

Product Name: **WORDRACE & WORDRACE Accessory Disk**
Equip. req'd: Atari 400/800 w/32K & BASIC Cartridge
Price: ?
Manufacturer: Don't Ask Computer Software
2265 Westwood Blvd., Suite B-150
Los Angeles, CA 90064

Description: As the name suggests, this is a word game that tests vocabulary skills. Game players, from one to four, must find the correct definition of a word from six possible choices. The clock is ticking so find the correct answer as quickly as possible. Choose your strategy: guess quickly or take more time to study the definitions. Loose points for incorrect answers or too much hesitation. There are three levels of play for everyone from pre-teens to pundits. The number of words in each round of play is selectable, also. For those more interested in sports or famous historical persons, an extra-cost accessory diskette is available. After booting the system diskette, insert the alternate data diskette and the new game data will load.

Pluses: Challenging and educational. This game would also be a valuable (and fun!) way for a student to prepare for college entrance exams.

Minuses: The graphics are simply boring, but word game lovers will overlook this aspect of the software's design.

Skill level required: Age 9 to adult.

Reviewer: Tim Kilby

Product Name: **Preparing For the SAT**
Equip. req'd: Atari Computer, Atari 410 Recorder
Price: $99.95
Manufacturer: Program Design, Inc.
11 Idar Court
Greenwich, CT 06830

Description: *Preparing For the SAT* is a cassette based series of lessons designed to help prepare for aptitude testing, especially the Scholastic Aptitude Test. The lessons cover analogies, vocabulary, quantitative comparisons, and number relationships. There is one cassette devoted to the taking of aptitude tests with the back side containing a time program for practice. There are a total of six cassettes included with the manuals.

Pluses: Two of the programs have a voice narration included on the cassette for reinforcement. This is a feature used by PDI that I feel greatly enhances the learning process. The information included in the manuals is valid and the programs are well done. After a demonstration, the adults attending spent over an hour working with the various lessons. They not only felt they were learning but found it enjoyable at the same time.

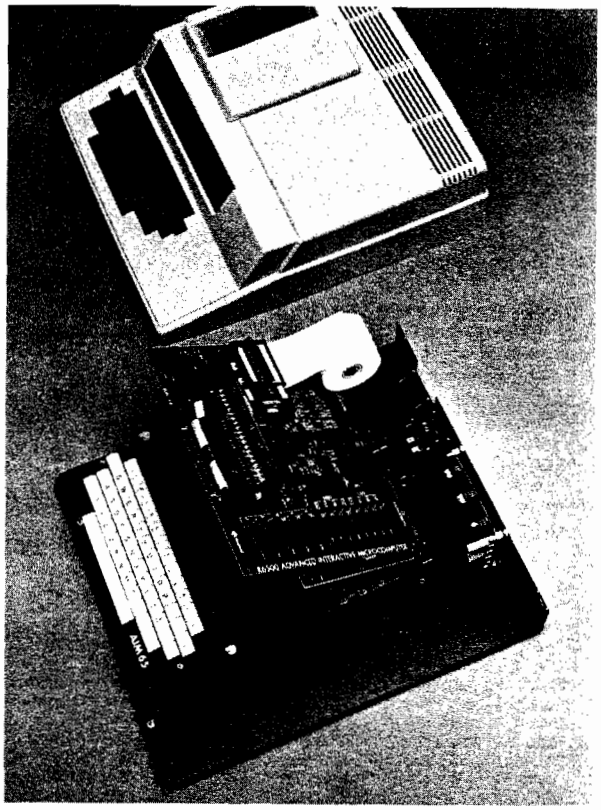
Minuses: Other than the fact that printed tests cannot be made from the programs, I found nothing to complain about.

Documentation: The programs come with two manuals. One is titled "Making The Grade, How To Take and Pass A TEST." This 34 page manual was written by the president of PDI, John Victor. He explains what the tests are, the best methods to use when taking them and shows examples of the types of questions used in the testing. The other 44 page manual explains how to use the programs and has a large section devoted to explaining the problems on the cassettes. They are well done.

Skill level required: High school student or anyone with an inquiring mind.

Reviewer: Richard E. DeVore

MICRO™



AIM HIGH

Let Unique Data Systems help you raise your sights on AIM 65 applications with our versatile family of AIM support products.

- Go for high quality with our ACE-100 Enclosure. It accommodates the AIM 65 perfectly, without modification, and features easy access two board add-on space, plus a 3″ × 5″ × 17″ and a 4″ × 5″ × 15.5″ area for power supplies and other components. $186.00.
- Get high capability with Unique Data System's add-on boards. The UDS-100 Series Memory-I/O boards add up to 16K bytes of RAM memory or up to 48K bytes ROM/PROM/EPROM to your Rockwell AIM 65. You also get 20 independently programmable parallel I/O lines with an additional user-dedicated 6522 VIA, two independent RS-232 channels with 16 switch-selectable baud rates (50 to 19.2K baud), and a large on-board prototyping area. Prices start at $259.00.
- If you need to protect against RAM data loss, the UDS-100B offers an on-board battery and charger/switchover circuit. $296.00.
- Heighten your AIM 65's communications range by adding the UDS-200 Modem board. It features full compatibility with Bell System 103 type modems and can be plugged directly into a home telephone jack via a permissive mode DAA. No need for a data jack or acoustic coupler. The UDS-200 also has software-selectable Autoanswer and Autodial capability with dial tone detector. The modem interfaces via the AIM 65 expansion bus, with the on-board UART and baud rate generator eliminating the need for an RS-232 channel. $278.00.
- The UDS-300 Wire Wrap board accepts all .300/.600/.900 IC sockets from 8 to 64 pins. Its features include an intermeshed power distribution system and dual 44-pin card edge connectors for bus and I/O signal connections. $45.00.
- Get high performance with the ACE-100-07 compact 4″ × 5″ × 1.7″ switching power supply, delivering +5V @ 6A, +12V @ 1A, and +24V for the AIM printer. $118.00.

Installation kits and other related accessories are also available to implement your AIM expansion plans. Custom hardware design, programming, and assembled systems are also available. High quality, high capability, high performance, with high reliability... all from Unique Data Systems. Call or write for additional information.

Unique Data Systems Inc.
1600 Miraloma Avenue, Placentia, CA 92670
(714) 630-1430

Products for Commodore, Atari, Apple, and others!

NEW

THE MONKEY WRENCH II
A PROGRAMMERS AID FOR ATARI 800
NEW AND IMPROVED — 18 COMMANDS
PLUGS INTO RIGHT CARTRIDGE SLOT

If you are a person who likes to monkey around with the ATARI 800, then THE MONKEY WRENCH II is for you!! Make your programming tasks easier, less time-consuming and more fun. Why spend extra hours working on a BASIC program when the MONKEY WRENCH can do it for you in seconds. It can also make backup copies of boot type cassette programs. Plugs into the right slot and works with ATARI BASIC cartridge.

$59.95

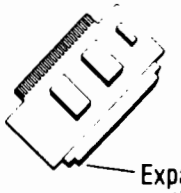
The MONKEY WRENCH provides 18 direct mode commands. They are: AUTO LINE NUMBERING — Provides new line numbers when entering BASIC program lines. RENUMBER — Renumbers BASIC's line numbers including internal references. DELETE LINE NUMBERS — Removes a range BASIC line numbers. VARIABLES — Display all BASIC variables and their current value. Scrolling — Use the START & SELECT keys to display BASIC lines automatically. Scroll up or down BASIC program. FIND STRING — Find every occurrence of a string, XCHANGE STRING — Find every occurrence of a string and replace it with another string. MOVE LINES — Move lines from one part of program to another part of program. COPY LINES — Copy lines from one part of program to another part of program. FORMATTED LIST — Print BASIC program in special line format and automatic page numbering. DISK DIRECTORY — Display Disk Directory. CHANGE MARGINS — Provides the capability to easily change the screen margins. MEMORY TEST — Provides the capability to test RAM memory. CURSOR EXCHANGE — Allows usage of the cursor keys without holding down the CTRL key. UPPER CASE LOCK — Keeps the computer in the upper case character set. HEX CONVERSION — Converts a hexadecimal number to a decimal number. DECIMAL CONVERSION — Converts a decimal number to a hexadecimal number. MONITOR — Enter the machine language monitor.

In addition to the BASIC commands, the Monkey Wrench also contains a machine language monitor with 16 commands used to interact with the powerful features of the 6502 microprocessor.

NEW

VIC RABBIT CARTRIDGE AND CBM 64 RABBIT CARTRIDGE

"High-Speed Cassette Load and Save!"



$39.95
(includes Cartridge and Manual)

Expansion Connector on the VIC Cartridge

"Don't waste your Life away waiting to LOAD and SAVE programs on Cassete Deck."

Load or Save 8K in approximately 30 seconds! Try it — your Un-Rabbitized VIC takes almost 3 minutes. It's not only Fast but VERY RELIABLE.

Almost as fast as VIC Disk Drive! Don't be foolish — Why buy the disk when you can get the VIC Rabbit for much, much less!

Easy to install — it just plugs in.
Expansion Connector on rear.
Works with or without Expansion Memory.
Works with VIC Cassette Deck.
12 Commands provide other neat features.

Also Available for 2001, 4001, and 8032

TELSTAR 64

Sophisticated Terminal Communications Cartridge for the 64.

PFO 100 000 CP<01>02 BELL 12:30:00 10:14:36
(TELSTAR's Status Line)

Don't settle for less than the best!

- Upload/Download to/from disk or tape.
- Automatic File Translation.
- Communicates in Industry Standard ASCII.
- Real-Time Clock plus Alarm Clock.
- Line editing capability allows correcting and resending long command lines.
- 9 Quick Read functions.
- Menu-driven.
- Similar to our famous STCP Terminal package.
- Works with Commodore Modems and supports auto-dialing.

The best feature is the price — **only $49.95** (Cartridge and Manual)

Machine Language Monitor Cartridge

for the CBM 64

More than 20 commands allow you to access the CBM 64's Microprocessors Registers and Memory Contents. Commands include assemble, disassemble, registers, memory, transfer, compare, plus many more.

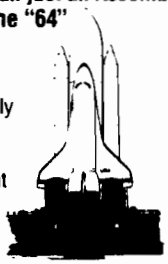
Someday every CBM 64 owner will need a monitor such as this.

Cartridge and Manual — **$24.95**

More than just an Assembler/Editor!
Now for the "64"

MAE

It's a Professionally Designed Software Development System



for
PET
APPLE
ATARI
~~$169.95~~
Only $59.95

NOW, The Best for Less!

- Designed to improve Programmer Productivity
- Similar syntax and commands — No need to relearn peculiar syntaxes and commands when you go from PET to APPLE to ATARI.
- Coresident Assembler/Editor — No need to load the Editor then the Assembler then the Editor, etc.
- Also includes Word Processor, Relocating Loader and much more
- Join the ATUG User Group for MAE formatted disks
- STILL NOT CONVINCED? Send for free spec sheet!

ATARI, PET, AND CBM 64 EPROM PROGRAMMER

Programs 2716 and 2532 EPROMs. Includes hardware and software. PET = $75.00 – ATARI and CBM 64 (both include sophisticated machine language monitor) = $119.95

CBM 64 Debugger

A more sophisticated Machine Language Monitor/Debugger. 20K of object code makes this a powerful tool. Works as a symbolic debugger for the MAE assembler. Diskette and Manual — **$49.95**

TRAP 65

TRAP 65 is a hardware device that plugs into your 6502's socket. Prevents execution of unimplemented opcodes and provides capability to extend the machines' instruction set.
For PET/APPLE/SYM.
Reduced from $149 95 to $69 95

5¼ INCH SOFT SECTORED DISKETTES

Highest quality. We use them on our PETs, APPLEs, ATARIs, and other computers. $22.50/10 or $44.50/20

Prowriter Printer - Excellent dot matrix print Parallel = $489 00
Serial = $600 00 IEEE = $589 00

DC Hayes Smart Modem = $235 00
DC Hayes Micro Modem II = $289 00

Rana Disk Drive - 375
4 Drive Controller - 114

EPROMS 2716 = $4.50 2532 = $7.50
Over 40 Commodore Programs by Baker (on 4040) = $25.00

Eastern House

3239 Linda Dr.
Winston-Salem, N.C. 27106
(919) 924-2889 (919) 748-8446
Send for free catalog!

VISA
MasterCard

Raise your Apple's IQ Twelve Times A Year!

A One Year Subscription Brings You 12 Issues With:

Over $500 of Programs for your Home, Business, Education and Entertainment. Complete Program Listings with Instructions.

Comprehensive Articles that show what each program does, how to use it and how to type it into your Apple, Franklin ACE or other Applesoft-compatible computer.

Regular Features for the Beginner and Expert.

On The Scene
The Latest New Software/Hardware Releases.

Products! Inside and Out
Comprehensive Product Reviews.

Education Corner
Programs that help make Learning Fun.

Tips 'N Techniques
Little known programming Tricks you can Use.

Disassembly Lines
An Expert reveals the mysteries of Applesoft.

Utilities
Superchargers for Basic, DOS, Printing, and More.

Games
Arcade Fun you can Type and Run.

Note
☐ Domestic U.S. First Class subscription rate is $51.95
☐ Canada Air Mail subscripton rate is $59.95
☐ Outside the U.S. and Canada Air Mail subscription rate is $89.95
All payments must be in U.S. funds drawn on a U.S. bank.

©1983 by MicroSPARC Inc. All Rights Reserved.

Apple® is a registered trademark of Apple Computer, Inc.
ACE® is a registered trademark of Franklin Computer, Inc.

Try a NIBBLE!

Here's what some of our Readers say:

☐ *"Certainly the best magazine on the Apple!"*
☐ *"Impressed with the quality and content."*
☐ *"Programs remarkably easy to enter."*
☐ *"I'll be a subscriber for life!"*
☐ *"Your service is fantastic . . . as a matter of fact, I'm amazed!"*

Try a NIBBLE!

NIBBLE is focused completely on the Apple and Applesoft-compatible computers.

Buy NIBBLE through your local Apple Dealer or subscribe now with the Coupon or Order Card in this issue.

You'll want Back Issues Too!

Here are some examples of programs you can get:

The Investor—Stock Tracking, Reporting, and Graphing.

Recipe Box—Kitchen/Menu Management made Fun.

The Librarian—Auto Logging and Retrieval of your Disks.

Designer/Illustrator—Art/Design Creation and Composition with Graphics.

Machine Language Editor—Quick and Easy Aid for Typing and Changing M/L Programs.

And Much . . . Much More!

NIBBLE will become a permanent part of your Reference Library. Discover why 95% of NIBBLE Readers save every issue!

Join more than 120,000 Apple/Ace users who say:
"NIBBLE IS TERRIFIC!"

SUBSCRIBE NOW AND SAVE $12.00 OFF THE COVER PRICE!

nibble

MasterCard VISA
We accept Master Charge & Visa

Box 325, Lincoln, MA 01773 (617) 259-9710

I'll try nibble!
Enclosed is my $26.95 (for 12 issues)
(Outside U.S., see special note on this page.)

☐ **check** ☐ **money order** ☐ **bill me (U.S. only)**

Your subscription will begin with the next issue published after receipt of your check/money order.

Card # _____ Expires _____
PLEASE PRINT CLEARLY
Signature _____
Name _____
Address _____
City _____
State _____ Zip _____

PUT THE FULL POWER OF YOUR VIC-20 AT YOUR COMMAND!

Order your copy of MICRO's newest book...

MASTERING YOUR VIC-20

with eight BASIC projects

Now you can do more with your VIC-20. This new book and the eight projects and programs it contains can teach you how to master VIC BASIC programming. Each chapter concentrates on a particular aspect of VIC BASIC, and each program is accompanied by discovery-oriented, tutorial text... clear directions that will quickly have you writing programs, modifying them and adding features all on your own. And to help you master your VIC-20 even faster, all eight programs in the book are already keyed in on the accompanying cassette.

You'll Receive:

✦ **MICRO Calc...** a miniature spread-sheet program that makes complex, repetitive calculations a breeze.

✦ **MASTER...** a guessing game that teaches programming with random numbers and flags.

✦ **VIC Clock...** to teach you ON..GOSUB function and character graphics.

✦ **BREAK-UP...** a popular game that also teaches how animation is achieved with PEEKS and POKES to screen memory.

✦ ***Plus...*** music programming, string manipulation, sorting demonstrations, and more.

Each Program Worth the Price of the Book!

Use this coupon or the postage paid card in this issue to order.

MICRO Books P.O. Box 6502, Chelmsford, MA 01824

☐ YES, please rush _____ copies of MASTERING YOUR VIC-20 (w/cassette), at only $19.95 per copy (plus $2.00 s/h, MA res. add 5% sales tax).

My payment of $_____ is enclosed. I'm paying by ☐ Check ☐ MO ☐ VISA ☐ MC

NAME _____

ADDRESS _____

CITY _____ STATE _____ ZIP _____

CREDIT CARD # _____ EXP. DATE _____

Allow 6-8 weeks for delivery.

Order your copy of MASTERING YOUR VIC-20 Today!

MICRO™ Interface Clinic

by Ralph Tenny

A computer can be interfaced to real time events, but if a computer is to be able to react with and control real time activities, it must know when the events are happening. There are two basic ways to accomplish this — polling and *interrupts*. An interrupt is a way to signal a running computer — gaining its attention — that it must delay further execution of the running program to service another event. Most modern microprocessors have provision for three kinds of interrupts. The first, called RESET, is used on startup and causes the microprocessor's internal registers to be set to a known condition instead of the random condition which happens when power is first applied. In addition, the internal RESET algorithm initiates certain operations, including reading an external memory location for (usually) the address of the programmer's idea of a proper initialization routine. This is called *indirect addressing*, which means that the first data read from memory is not an instruction but the *address* of an instruction.

Two other interrupts are common also. The IRQ (Interrupt ReQuest) is typically a *maskable* interrupt (meaning it can be turned off via a software flag), and the NMI (Non-Maskable Interrupt) are usually available on modern microprocessors. These interrupts cause some portion of the microprocessor's status to be saved so the interrupted program can be resumed in orderly fashion. Those of you with 6809-based machines also have three software interrupts (similar to the 6502's BRK instruction) and the FIRQ (Fast Interrupt ReQuest) which responds more rapidly than IRQ by saving fewer processor registers).

Programming for interrupts requires special precautions and programming methods. Not only do you have to have special *interrupt service* programs, you must carefully manage the interrupt enable bit and the associated hardware which causes the interrupt. It is universal practice that interrupt input pins are at logic one level during normal operation, and respond (issue an interrupt) when the pin is pulled to logic zero. Usually, the NMI interrupt is edge-sensitive (a negative-going input is latched internally) so that the pin must go high and then come low again before another interrupt is accepted. IRQ inputs are usually level-sensitive; if the interrupt service routine is completed before the pin is released, another interrupt will be issued immediately. In one aspect, the microprocessor's response to either IRQ or NMI is identical — the current instruction is completed before the interrupt is honored. In most cases, the microprocessor also ignores further interrupts until the current interrupt service routine is finished. This is accomplished by using the RTI (ReTurn from Interrupt) instruction to terminate the service routine.

The program in the listing illustrates how to handle interrupts caused by the CD input of the serial port. This input drives the CA1 pin of the I/O PIA of the Color Computer, and the IRQ output from the PIA is connected to the 6809's FIRQ pin. The IRQ and the FIRQ interrupts each have their own disable flags. If either bit is set to logic one, the corresponding interrupt is inhibited or turned off. Unlike some processors (6502 for example) which directly set or clear status register bits, the 6809 uses special AND or OR instructions which set or clear selected status bit. One such example is shown in the listing one line above the label SPIN ($301D) -ANDCC $BF. If you remember how the logical AND works, any bit in the operand is set to zero if the corresponding mask bit is zero. In this case, the *bit mask* (pattern) is $BF, which has all bits except Bit 6 set to logic one. Thus, Bit 6 in the operand (Condition Code register or Status Register) is set to logic zero; this *enables* (turns on) the FIRQ interrupt. Similarly, at the label QUIT ($3055) the instruction ORCC $40 is used to turn off the FIRQ interrupt. Refer to Figure 1, which shows the PIA Control Register and briefly identifies the functions of each Control Register bit. For now, we will skip a complete description of this register's functions; instead, note bits 6 and 7, which are IRQ flags. CA1 is always an input and can be programmed as an interupt; CA2 can be either an ouput or input/interrupt. Similar relationships are true for Control Register B, CB1 and CB2. If CA1 and CA2 are programed as interrupts, Bit 7 responds when CA1 is pulled low, and Bit 6 responds to CA2. Last month's applications program polled the RS-232IN line (bit 0 of the port), but since our serial adapter also pulses the CD input, we could have polled bit 7 of the Control Register instead.

Let's examine the program flow in the listing. Beginning at the label START, the location BUFR (defined at the program's end) is cleared, then the time constant DVAL is stored in CoCo's baud rate buffer. The next command points the Y INDEX to the next location past BUFR; this can be omitted after program debugging; it simply provides a record of the input values to aid in troubleshooting. The next three instructions read the current value in the PIA Control Register, turn on Bit 0, and restore the modified value. This is the first of three steps required to completely enable the interrupt structure so this PIA can cause a processor interrupt.

The second portion of setting up an

TAYLORMADE SOFTWARE • TAYLORMADE

For the COMMODORE 64™ and VIC 20™

TOUCH TYPING TUTOR

19 lessons—Watch your TV screen to learn proper finger placement.
PRACTICE—learn your word/min. rate typing pseudo words.
TEXT—English sentence fragments for timed tests of any duration.
Includes 12-page manual.

TTT64D Diskette $24.95
TTT64 Cassette $19.95
TTT5K VIC Cassette $19.95

PINBALL MATH

Improve math skills playing PINBALL MATH! Add, subtract, multiply, divide, each with 3 levels from basic facts to two-digit operands. Sprites, sound, pinball graphics, scoreboard. For students in grades 1-6.

PM64D Diskette $24.95
PM64 Cassette $19.95
PM-VIC+8K Cassette $19.95

FUN FRACTIONS

Watch VIC show you all the steps to do addition, subtraction, multiplication, division, and reductions of fractions. Your turn; can you answer before the parachute jumper crashes? For students in grades 4-9. Includes 16-page manual.

FF-VIC+8K Cassette $19.95

Note: Product code gives computer and memory required.

U.S./Canada add $1.50 shipping/handling
COD charge $1.65
Foreign orders payable U.S. funds plus $4.00 shipping

TAYLORMADE SOFTWARE

P.O. Box 5574
Lincoln, NE 68505
(402) 464-9051

MasterCard VISA

Commodore 64 and VIC 20 are trademarks of Commodore Business Machines, Inc.

TAYLORMADE SOFTWARE • TAYLORMADE

MEMORY FOR YOUR VIC-20

PLUS A LITTLE EXTRA

32K Dynamic Ram — Same type as Commodore 64
24K Used normally, for full expansion. Blocks 1, 2, & 3.
"EXTRA" 8K maps into block 5, normally used for ROM cartridges (Games)
Access from BASIC for data storage, only through PEEK and POKE
Accesses in assembler for data, or write your own GAMES, & boot into them normally
The 3k "Block 0" ram has been intentionally left free for the VIC Superexpander
Each 8K block can be switched in or out for compatability with games, or other peripherals
Powered by your VIC-20 (175 ma typ)
Plugs directly into VIC expansion slot or motherboard
Complete, assembled, & tested
90 day "No questions asked" money back return
5 year manufacturer's warranty

$109.95

TEMPUS
Dept M101, 832 Brown Thrush
Wichita, KS 67212

We carry the entire XETEC line for VIC-20 and C64. Write for literature. Master Card and VISA accepted! Send card number & expiration date. Handline charges add $3.00. Personal checks take 3 weeks to clear, before we ship. *VIC-20 & Commodore 64 are Commodore trademarks.*

interrupt is to load the *vector*, which is the address of a routine to service (respond to) a particular interrupt. The 6809 microprocessor has provisions for eight interrupts, seven of which have been implemented and one has been reserved for future expansion. Upon receipt of an interrupt, the 6809 reads one of the eight two-byte values it finds in the memory space $FFF0 to $FFFF. When a Synchronous Address Multiplier (SAM) is in the system, as in the Color Computer, SAM forces these interrupts to be moved to $BFF0-$BFFF. If you examine memory (use a debug monitor or PEEKs), you will find some two-byte values stored as part of the BASIC ROM. Some of those addresses point to the memory area beginning at $0100. Remember, the interrupt structure expects these address vectors to be indirect addresses which point to the interrupt service routine. Since these routines can be anywhere, and of any length, a *jump table* is used. This is a series of absolute jumps, listed one after the other in memory. So, for any of the vectors beginning at $0100, you will find 7E XX XX, where XX XX represents the starting address of the interrupt service routine. The FIRQ vector is used to auto-start CoCo from a ROMPACK, so BASIC initializes the FIRQ vector at $010F. If we wish *our* FIRQ service routine to be used, then we must substitute our vector for the BASIC vector. The three program steps beginning at $3017 do just that. BASIC has already written the code $7E (JMP) in location $010F, so our program reads the address of INTSRV and writes two bytes ($302A) beginning at $0110. *That* is Step 2 needed to initialize an interrupt. Step 3 follows: read the port to clear Bit 7 in case CD has been pulsed while we were talking, and then enable the FIRQ interrupt. This cancels any pending interrupt which came in before we were ready, and allows our routine to begin with a clean slate.

In this particular case, our program simply goes into a wait loop, checking the keyboard to see if we push a key. Anytime an interrupt comes in, the eight bits captured by the serial adapter's input pin come spinning in, just as they did when we polled the RS-232IN line last month. This has been a slightly simplistic explanation, but it is accurate for any CoCo which does not have Extended BASIC. Extended BASIC runs a software timer based on IRQ, and triggered by a 60 Hz interrupt signal on CB1 of the keyboard PIA, but since FIRQ is a *higher priority* (more important) interrupt, our signal will dominate. Since our

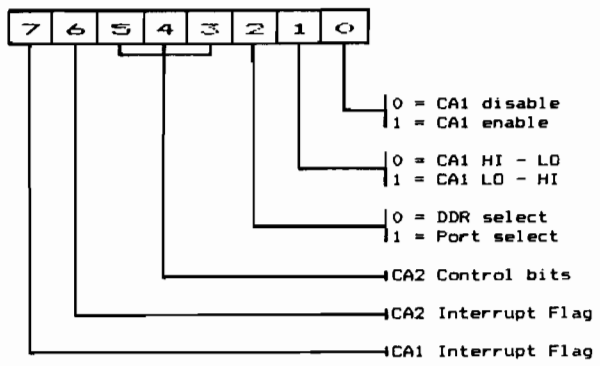


Figure 1. **Control Register of the 6821 PIA. Bits 0 and 1 control the CA1 Interrupt input pin (see text).**

routine will run longer than the 16.6 millisecond period of the 60 Hz interrupt *each time it happens*, the BASIC clock will miss a "tick" every so often. If you depend on this clock, you may wish to poll the serial adapter instead of run it under interrupt control.

Once the interrupt happens, the code at label INTSRV begins to execute. Much of this code is identical to the previous programs which we have used to exercise the hardware, so let's concentrate on the differences. Just as we had to manage the interrupt entry software carefully, certain things must be accomplished by the service routine. The processor automatically disables both interrupt bits whenever either IRQ or FIRQ are asserted, and the RTI instruction restores the original interrupt enable status upon exit from the service routine. Obviously, the service routine must perform the intended task which created the need for an interrupt, but it must also *clear* the interrupt (prevent the same interrupt from being asserted again).

If external hardware can be cleared or reset to remove the stimulus, this must be done. If this cannot be done, the service routine must continually check for the hardware status, waiting for it to clear itself. Our hardware automaticaly removes the stimulus, so we have one other thing to clear. Bit 7 of the Control Register was set by the input pulse on the CD input, and will remain set until the port is read (label EXIT). Note that although we read that port, this value is not used. The service routine is finally terminated with RTI, and (in this case) operation in the loop SPIN is resumed. Note that almost any other operation could take the place of this loop, but this is a simple example, so feel free to improvise.

We didn't get to the real world interfaces I promised last time, so we'll try again next time. This has been a learning series, preparing for "greater things", so any of you who have just joined us may have to review earlier columns for background. Let's move on and do more complex and comprehensive interfacing!

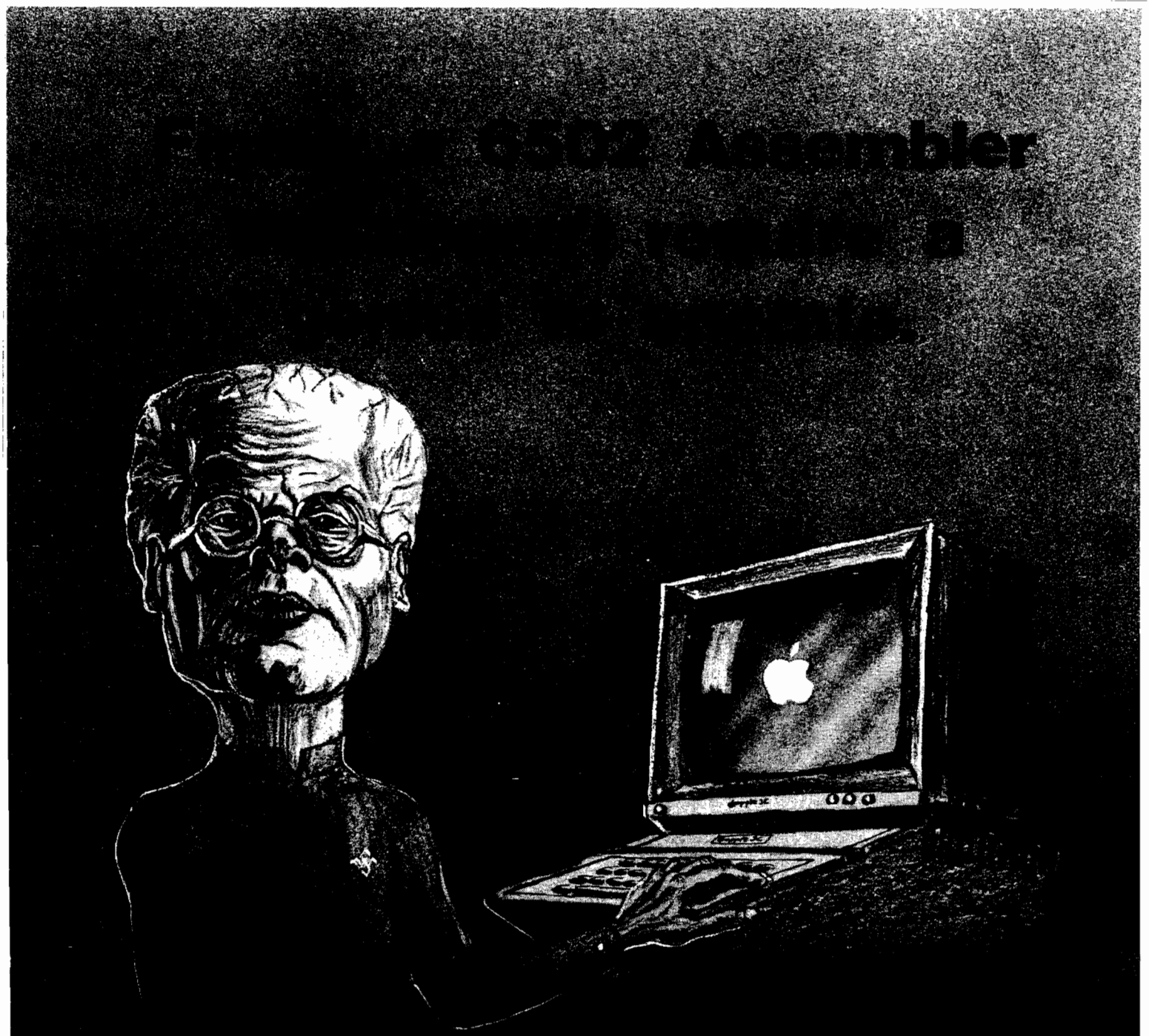
MICRO

Please forward questions and suggestions for discussion topics to Mr. Tenny at P.O. Box 545, Richardson, TX 75080.

Listing

```
                    * THIS PROGRAM WILL INPUT AN 8-BIT VALUE ON THE
                    * COLOR COMPUTER SERIAL PORT IN RESPONSE TO
                    * AN INTERRUPT ON THE CD INPUT.
                    * EQUATES
              010F  FIRQ    EQU   $10F      FIRQ INTERRUPT VECTOR
              00AB  DVAL    EQU   $AB       DELAY VALUE
              0095  BAUD    EQU   $95       BUFFER FOR DELAY CONSTANT
              A000  POLCAT  EQU   $A000     KEYBOARD SCAN
              FF20  PORTOUT EQU   $FF20     RS232 OUT PORT
              FF21  CTLIN   EQU   $FF21     CONTROL PORT FOR SERIAL IN
              FF22  PORTIN  EQU   $FF22     SERIAL IN PORT
                    * MAIN PROGRAM
3000                        ORG   $3000
3000 7F   305D    START     CLR   BUFR     CLEAN SLATE
3003 8E   00AB              LDX   #DVAL    SET UP TIMER
3006 9F   95                STX   BAUD
3008 108E 305E              LDY   #BUFR+1  POINT TO RECORD BUFFER
300C B6   FF21              LDA   CTLIN    ENABLE CD INTERRUPT
300F 8A   01                ORA   #1
3011 B7   FF21              STA   CTLIN
3014 8E   302A              LDX   #INTSRV  RESET FIRQ VECTOR
3017 BF   0110              STX   FIRQ+1
301A B6   FF20              LDA   PORTOUT  RESET IRQ FLAG
301D 1C   BF                ANDCC #$BF     ENABLE FIRQ INTERRUPT
301F AD   9F A000 SPIN      JSR   [POLCAT] TEST KEYBOARD
3023 26   30                BNE   QUIT -
3025 B6   FF20              LDA   PORTOUT  RESET IRQ FLAG
3028 20   F5                BRA   SPIN     LOOP WAITING FOR INTERRUPT
302A 9E   95      INTSRV    LDX   BAUD     GET DELAY VALUE
302C 1F   10                TFR   X,D      DIVIDE BY TWO
302E 47                     ASRA
302F 56                     RORB
3030 1F   01                TFR   D,X
3032 8D   24                BSR   DELAY    AND COUNT IT DOWN
3034 B6   FF22              LDA   PORTIN   START BIT?
3037 84   01                ANDA  #1
3039 26   16                BNE   EXIT     IF NOT, SKIP IT
303B 9E   95                LDX   BAUD     OTHERWISE, GET FULL DELAY TO
303D 8D   19                BSR   DELAY    READ MIDDLE OF FIRST BIT
303F C6   08      SETUP     LDB   #8       BIT COUNT
3041 B6   FF22    INPUT     LDA   PORTIN   READ PORT
3044 A7   A0                STA   ,Y+
3046 44                     LSRA
3047 76   305D              ROR   BUFR     BIT INTO STORAGE
304A 9E   95                LDX   BAUD     SET UP TIMER
304C 8D   0A                BSR   DELAY
304E 5A                     DECB           COUNT DOWN BITS
304F 26   F0                BNE   INPUT    AND DO EIGHT PASSES
3051 B6   FF20    EXIT      LDA   PORTOUT  RESET IRQ FLAG
3054 3B                     RTI            RETURN TO WAIT LOOP
3055 1A   40      QUIT      ORCC  #$40     DISABLE INTERRUPT
3057 39                     RTS            AND THEN QUIT
3058 30   1F      DELAY     LEAX  -1,X
305A 26   FC                BNE   DELAY
305C 39                     RTS
305D              BUFR      RMB   1
                            END
```

MICRO



LISA v2.6

You don't need an I.Q. of 200 to learn assembly language on your Apple II or Apple*IIe*. All you need is the best 6502 assembler around: **Lazer's Interactive Symbolic Assembler (LISA v2.6)**. LISA is the easy-to-learn and easy-to-use system that will have even the absolute beginner up and running in a matter of minutes. In addition to LISA's clearly written 240-page manual more tutorial material is available for LISA than any other assembler *(including books by D. Fudge, R. Hyde, W. Mauer, and R. Mottola)*. LISA is the beginner's best choice.

Even if your I.Q. is 200, you'll appreciate LISA's speed, power, and ease of use. At 20,000 lines/minute it's 20 to 30 times faster than ORCA/M or Apple's own Toolkit assembler. That's why people like Bill Budge, Ken Williams, Brian Fitzgerald, Don Fudge, and Steve Wozniak use it every day. LISA is very usable. That's why it's the most often used assembler ever created for the Apple II.

LISA v2.6 is available for $79.95 at computer stores everywhere. If your local computer store doesn't have a copy you can order directly from:

Lazerware, 925 Lorna St., Corona, California 91720. (714) 735-1041

LAZERWARE

COMMODORE 64 PROGRAMS

BUSICALC 3

BUSINESS
CALCULATION
PROGRAM

$129

The "TRUE 3-D"
Electronic Spreadsheet
for the Commodore 64

BUSICALC 1

BUSINESS
CALCULATION
PROGRAM

$69

A "BEST BUY"
Electronic Spread Sheet
for all Commodore's

BUSICALC 2

BUSINESS
CALCULATION
PROGRAM

$99

The MOST CAPABLE
Electronic Spread Sheet
for the Commodore 64

WE'RE HERE!

Skyles Electric Works
231E South Whisman Road
Mountain View, CA 94041
(415) 965-1735

AVAILABLE NOW! Call 800 227-9998**
For the name of your nearest dealer,
detailed information or a catalog of products.
**California, Canada and Alaska, please call (415) 965-1735.*

In Europe: SUPERSOFT, Winchester House, Canning Road, Harrow Wealdstone, England HA3 7SJ, Tel. 01 861 1166

Next Month in Micro

Last month in "Next Month in **MICRO**" we promised you articles detailing how to define your own character sets on the new Epson FX-80 printer for the Commodore 64, VIC-20, and Atari systems. Due to a lack of time and space, this material was not presented this month. It will, however, be presented in the next issue, and will include programs in BASIC to define the special characters on your display, to send the appropriate information to the FX-80 to define the characters, and to output BASIC listings. Some of the routines developed for the article were used to generate listings in this issue.

The main feature topic for January is Communications. One of the areas of microcomputer usage which is really expanding is that of telecommunication. There are literally hundreds of "bulletin boards", "teleservices", and so forth for you to talk to. **MICRO** is developing on its own hardware/software system, The **MICRO** Program Dump, that will allow subscribers to "download" programs that are published in **MICRO** directly into their microcomputers. The feature article Transfer Programs Over the Telephone will help your micro tie into the **MICRO** Program Dump. A second feature will be on Local Networks, a form of communication that allows you to interconnect various system components such as microcomputers, printers, storage devices and so forth. A third article will be about Using the VIC/C64 Parallel Port, providing cabling information and a program for transfering information via this useful, but not well supported, device.

MICRO™

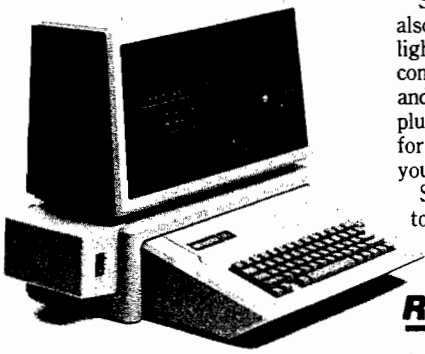
NO POWER SPIKES WITH SUPER FAN II.

Super Fan II's Zener Ray™ Transient Voltage Suppressor and Power Filter squelches spikes up to 6000 amps—even those caused by lightning—while responding up to 100 times faster than Apple II's built-in suppressor.

In addition, Super Fan II cools your Apple, removing heat buildup at a remarkable 17 cubic feet of air per minute. Yet it's the quietest fan of its kind on the market.

Super Fan II also positions a lighted on/off computer switch and two accessory plugs at your fingertips. It's warranted for two years and simply clips to your Apple II, IIe or monitor stand.

See your R.H. Electronics dealer today about Super Fan II*, or contact us at 566 Irelan Street, Buellton, CA 93427, (805) 688-2047.



RHELECTRONICS, INC.

Super Fan II, in black or tan: $109. Without Zener Ray, $74.95. Additional air flow seals, $5. Available in 240V/50 Hz.

*Dealer/OEM inquiries invited. *U.S. Patent #D268283 #4383286*

France, call B.I.P. 1-255-4463 Australia, call Imagineering (02)212-1411

This Christmas, Atari and Commodore* owners will go out of their minds.

It'll take about 30 seconds.

Because once you boot a Datamost game on your Atari home computer, or your Commodore 64, you'll come face to screen with the most mind-blasting games ever.

And what better time to go out of your mind than Christmas?

Our Music Will Have You Hearing Things.

Going out of your mind never sounded so good.

Because now our games have music. You heard right. Music. Original Music. Throughout. And scored just for our newest releases. You'll go nuts over it.

Our Graphics Will Have You Seeing Things.

You and everybody else.

Because our games are so great-how great are they?-they're so great you'll want to play them again and again and again. And then your friends will go bonkers over them.

THE TAIL OF BETA LYRAE.™ Changes as you play. Unpredictable. Impossible to master. (No one has!)

And they'll want to play. And then your family will want to play. And then total strangers off the street will want to play and...

Mind-blowing arcade-quality action like this makes it one mad world, pal.

We'll Torment You Right From The Start.

No time to settle down and get comfy.

The tension begins building the moment you boot the game.

Terrific screen titles tease you with the game's objective, scenario, characters.

And while you wait, hand twitching over the Joystick, you'll find out what planet you're on.

What the fuss is all about.

Why you're involved. And perhaps of singular importance to you personally, how to keep from being obliterated. So get ready to get the most out of your mind.

By getting the most out of ours.

DATAMOST™ INC

The most out of our minds.™

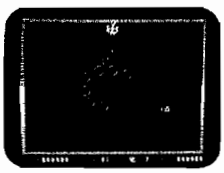
Datamost, Inc., 8943 Fullbright Ave., Chatsworth, CA 91311, (213) 709-1202

Atari is a trademark of Atari Computer. *Commodore 64 is a trademark of Commodore Business Machines, Inc. TM Registered Trademark of Datamost.

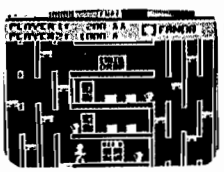
COSMIC TUNNELS.™ Four games in one. Four times the challenge! Incredible graphics.

MONSTER SMASH.™ Deathly strategy. Mash the monsters! Let the visitors live.

NIGHTRAIDERS.™ Strafe a city under siege with 3D angled selective firing!

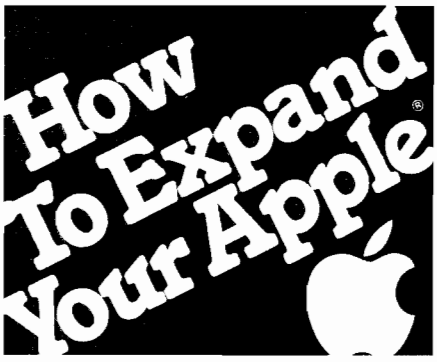


ROUNDABOUT.™ Sharpens your shoot 'em up skills. 24 different game screens. Habit-forming!



COHEN'S TOWERS.™ You're the mailboy in a big city skyscraper. Work your way to the top.

MR. ROBOT.™ Screens scream with color, action and sound! Design your own screens, too.



More speed, more memory workspace from the people who lifted the 64K limit.

In 1981, we introduced our famous Saturn RAM card to boost the Apple's memory by 32K. Now, there are 64K and 128K versions for added power.

Increased RAM lets you run bigger programs, and our VisiCalc® expansion software provides a dramatic expansion of workspace memory. Also, the extra memory can work as a pseudo disk for instant access. (No more 20-second searches!) The Saturn RAM board is a super aid for advanced word processing, data base management, spread sheet, and accounting applications. And perfect with PASCAL, CP/M®, and BASIC.

Use our easily installed cards in combination—and get a whopping 220K of VisiCalc workspace on the Apple IIe. Or sensational enhancements on Apple II, II+, Franklin, Basis and most Apple compatibles. We even have software that increases your Apple's memory in BASIC programming up to 4 megabytes!

This kind of performance requires sophisticated bank switching and supporting software that we originated, proved, and improved. Buy from the leader!

Let us help you expand your Apple's productivity. For information on our RAM boards and other Titan microcomputer products, see your computer dealer or contact: Titan Technologies, Inc., P.O. Box 8050, Ann Arbor, MI 48107; Telephone (313) 973-8422.

Sales and Marketing by The MARKETING RESOURCE GROUP, Costa Mesa, CA.

Titan
TECHNOLOGIES, INC.
FORMERLY SATURN SYSTEMS OF MICHIGAN

Apple is a registered trademark of Apple Computer, Inc.
VisiCalc is a registered trademark of VisiCorp, Inc
CP/M is a registered trademark of Digital Research, Inc.

Advertiser's Index

AB Computers 81
Alternative Energy Products 78
Apple Tree Electronics 97
Ark Computers 100
ArtSci 54
Atari Home Computers 1
Atari Program Exchange 60
Blue Sky Software ..... Cov II, Cov III
Check-Mate 28
Computer Mail Order 51,52
Computer Software Associates 91
CompuTech 48
Datamost, Inc 4, 103
Eastern House Software 94
Excert 85
Hollywood Hardware 58
Incomm 62
Interesting Software 38
John Bell Engineering 65
Leading Edge ..... Cov IV
Micro Magazine 35,80,96
Micro Motion 71
Micro Sparc 95
Micro Spec 7
Micro Ware Distributors 34
Midwest Micro 32
Momentum 2
Penguin Software 3
Percom Data 59
Performance Micro Products 34
Perry Peripherals 77
Prometheus 51
Protecto Enterprises 14,15,22
RH Electronics, Inc 102
Richvale Telecommunications 46
Safeware 74
Scientific Software 40
SGC 72
SJB Distributors 37
Skyles Electric Works 101
Star Micronics 45
Taylormade Software 98
Tempus 98
Unique Data Systems 93
Victory Software 8
Winders & Geist Inc 88
Zanim Systems 75

National Advertising Representatives

Home Office:
Sheila McDonough,
Advertising Representative
10 Northern Boulevard
P.O. Box 6502
Amherst, NH 03031

West Coast:
The R.W. Walker Co., Inc.
Gordon Carnie
2716 Ocean Park Boulevard,
Suite 1010,
Santa Monica, California 90405
(213) 450-9001

serving: Washington, Oregon, Idaho, Montana, Wyoming, Colorado, New Mexico, Arizona, Utah, Nevada, California, Alaska, and Hawaii (also British Columbia and Alberta, Canada)

Mid-West Territory:
Thomas Knorr & Associates
Thomas H. Knorr, Jr.
333 N. Michigan Avenue, Suite 400
Chicago, Illinois 60601
(312) 726-2633

serving: Ohio, Oklahoma, Arkansas, Texas, North Dakota, South Dakota, Nebraska, Kansas, Missouri, Indiana, Illinois, Iowa, Michigan, Wisconsin, and Minnesota.

SCRIPT 64™ Makes Your World Letter Perfect

SCRIPT 64 moves you into the exclusive neighborhood of sophisticated word processing for an amazing $99.95.

In addition to all the standard text editing features most higher priced systems offer, SCRIPT 64 also gives you an 80 column format and a built-in spelling dictionary...all in one program for the Commodore 64™.

To make your world letter perfect, ask your dealer for a demonstration of SCRIPT 64 today.

Developed by:

RTC

Richvale Telecommunications, LTD.

A Product of:

Blue Sky Software
Ashland Office Center
Evesham & Alpha Avenues
Voorhees, NJ 08043
609-795-4025

Distributed by:

Warehouse 1, Inc.	Computer Marketing Services, Inc.	DES-Data Equipment Supply
Eastern U.S. 800-253-5330	800-222-0585	213-923-9361
Western 800-255-0056		

Also available for the IBM-PC.

Script 64, Commodore 64 and IBM-PC are registered trademarks of Richvale Communications LTD., Commodore Electronics Limited, and International Business Machines, Corp. respectively.

THE SECRETS OF PERFECT MEMORY: ONE AND ONE HALF EARTH DOLLARS

AT LAST: THE WHOLE TRUTH ABOUT FLOPPIES.

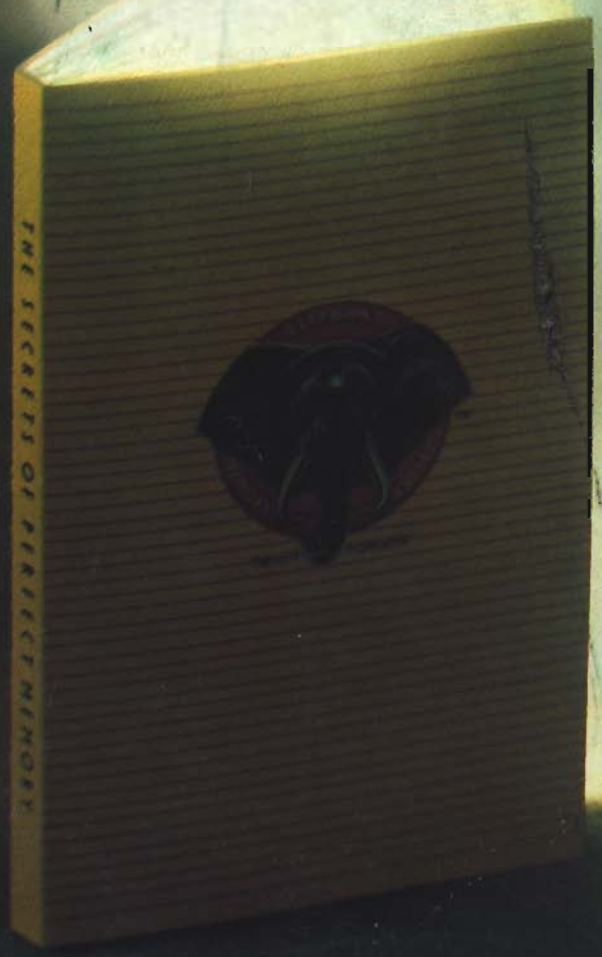
Amazing book reveals all!

How to keep from brainwashing your disk so it never loses it's memory.

How fingerprints can actually damage disks. Unretouched Kirlian photographs of UFO's (Unidentified Floppy Objects)! The incredible importance of making copies: the Department of Redundancy Department—and what goes on when it goes on! Powerful secret methods that scientists claim can actually prevent computer amnesia! All this, and much more . . .

In short, it's an 80-page plain-English, graphically stunning, pocket-sized definitive guide to the care and feeding of flexible disks.

For The Book, ask your nearest computer store that sells Elephant™ disks, and bring along one and one half earth dollars.

For the name of the store, ask us.

ELEPHANT MEMORY SYSTEMS® Marketed exclusively by Leading Edge Products, Inc., Information Systems and Supplies Division, 55 Providence Highway, Norwood, MA 02062. Call toll free 1-800-343-8413, In Massachusetts, call collect (617) 769-8150, Telex 951-624.