THE MICROCOMPUTER MAGAZINE

commodore

AUGUST/SEPTEMBER, 1982
$2.50

Software is the solution... for making your business more profitable!

And, the most comprehensive business and professional software is Info-Solutions from Info-Designs, Inc.

Info-Designs, Inc. is a rapidly growing company with a "commitment to provide the highest quality microcomputer software anywhere!"

Our Professional Practice Software Solutions provide a comprehensive management system for attorneys, engineers, architects, consultants, accountants, and other professionals who derive income from billing their professional services and reimbursable expenses.

Our management Accounting Software Solutions provide a totally integrated accounting system for the small to medium size business—retailer, wholesaler, distributor, manufacturer, and service company.

OFFICIALLY APPROVED BY COMMODORE

Now, on one compact Commodore CBM microcomputer system, you can achieve better management control of unbilled time, billings, accounts receivable, accounts payable, checkwriting, payroll, general ledger, inventory management and on-line order entry/point-of-sale.

See your local Commodore dealer today for the **Information-Solution** to your business automation needs.

InfoDesigns

INFORMATION SOLUTIONS USING MICROCOMPUTER TECHNOLOGY

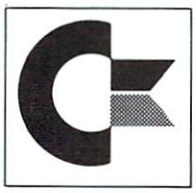
6905 Telegraph Rd. ● Birmingham, MI 48010 ● (313) 540-4010

Circle #1 on Reader Service Card

THE MICROCOMPUTER MAGAZINE

commodore

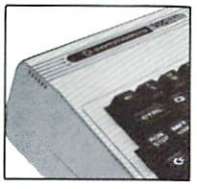
FEATURES



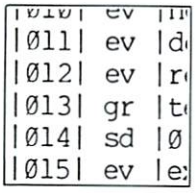
Commodore Introduces "Easy" Spreadsheet Programs 8

Commodore User Wins Prestigious Award for Educational Efforts . 29

A Case History: Moving Company Implements CBM System 45



Learn Just How Easy it is to Design Business Forms on Your VIC 20 55



Product Review: Read all about Silicon Office, a "Three-in-One" Business Software Package 86

DEPARTMENTS

Q & A Hotline 2
Editor's Notes 4
From Where I Sit 5

Commodore News
"Easy" Spreadsheet Programs 8
Telecommunications with VIC 20/ VICMODEM . . . 10
The Commodore Information Network 11
Silicon Office 13
SuperPET Update 14
Power-Play Report 16
Commodore Bibliography 17
Frequently Asked Questions and Answers 20
Users Clubs 24

Education
Software Selection 26
School Offers Instruction in Computer Literacy 28
Award Winner Uses Commodore Computers 29
Running a Completely Automated Experiment by Microcomputer 32
Update: Commodore Educational Resource Centers 36

Business
Income Tax Practitioner Praises Commodore Tax Package 42
A Case History: Moving Company Implements CBM System 45

VIC 20
The VIC Magician 49
Enhanced Business Forms on the VIC 20 55

Programmer's Tips
Line Finder 60
Introduction to the Machine Language Monitor . . . 63
BASIC Programming on the Commodore 64 65
Conversion Tutor 67
Recovering BASIC on the PET/ CBM 69
Sorting Tips 71
Commodore Disk Drive on an Old ROM PET 77

Excerpts From a Technical Notebook
Faster than a Speeding Cathode Ray 80
Some "One-Liners" 81
Deriving Mathematical Functions 81
VIC Notes 82
PET Quickie 82
Did You Know? 83

Product Review
Silicon Office 86

New Products 90

Butterfield on Commodore 98

Book Review
PET Graphics 100
Four More Reviews 102

Bitdiddling 105

Projections and Reflections 107

commodore

Editorial Manager
David A. Kaminer

Editor
Paul Fleming

Contributing Editor
Diane LeBold

Staff Writers
Ann Brickles
Bruce Downing
Neil Harris
Paul Goheen
Jeff Hand
Walt Kutz
Pat McAllister
Sue Mittnacht
John O'Brien
Joseph Siciliano
Michael Tomczyk

Contributing Writers
Jim Butterfield
Elizabeth Deal
Craig Johnson
Sue Revels
Michael Richter
Louis Sander
George Severance
Dwight Wheeler
Don White

Technical Staff
Paul Goheen
Pat McAllister
Dave Middleton
Joseph Siciliano
John Stockman
Michael Tomczyk

Circulation Manager
John O'Brien

Circulation Assistant
Diane Ottinger

Advertising Manager
Paul Fleming

Q&A HOTLINE

Q. I recently purchased a SuperPET, 8300 letter quality printer and an 8050 disk drive. At the same time, I also purchased Wordpro 5 Plus. When I tried to use this word processing program, I found the chip would not fit because the SuperPET has the UD11 socket filled. I contacted a Commodore dealer who provided me with a RAM/ROM and installation instructions. A local Commodore technician then installed this RAM/ROM. After installation, the technician turned on the SuperPET and the WordPro 5 Plus introduction immediately appeared on the screen. It seemed all was in order. However, several days later I tried to use the WordPro program and encountered a problem. I worked my way through the program and answered the prompted questions until I reached the request for DISK DRIVE DEVICE #? I pushed the RETURN key and the CRT went blank. I tried to operate the program several times and the same thing happened. Could you please advise as to what the problem might be?

H. Ringstad
Belle Vernon, PA

A. *WordPro 5 Plus will only operate with Commodore's 8096 (an 8032 with the 64K add-on board). It will NOT run on the SuperPET. However, WordPro 4 Plus will operate in the SuperPET's 6502 mode, allowing two text areas. In addition, you should be aware that the chip in the UD11 socket is a test chip installed at the factory for initial checkout. That chip must be replaced with the WordPro 4 Plus chip that should be provided with the manual and disk.*

Q. What happens when a 3K (or 8K and 16K) expander is inserted? Why don't some of my programs work any more?

J. McAllister
St. Joseph, MI

A. *The memory map of the VIC changes whenever any extra memory is added to the VIC. Key locations in RAM are shifted, including the start of the BASIC program, start of the screen, and start of color memory.*

The start of BASIC is normally at location 4097. With an extra 3K plugged in, the location moves down to 1025. With an extra 8K or 16K, the location moves up to 4608. This movement won't affect BASIC programs, but does hurt machine code instructions, which are very dependent on where in memory they go. Programmable characters must also go in the correct locations.

The start of the screen RAM is normally 7680. With an extra 3K this doesn't change, but an extra 8 or 16K causes the location to move down to 4096. Anything that POKES into the screen will be affected by this change, unless the software is written to check where the screen is located, instead of assuming it is at location 7680.

The start of color memory shifts between the normal start at 38400, and the expanded memory version at 37888. Programs using PRINT won't notice, but programs POKEing into color memory will have problems. In addition, some programs are written to be used in an unexpanded VIC only. Unplug any memory expander when using these programs.

Q. I am attempting to learn machine language, but with little success. I have purchased the Commodore Assembler program, but have been struggling with the documentation. I have a feeling that some of my problem stems from the fact that I have never had access to any type of documentation on the Monitor. From what I have been able to learn from

Continued on page 4.

Commodore—The Microcomputer Magazine *is published bi-monthly by Commodore Business Machines, Inc., 487 Devon Park Drive, Wayne, PA 19087. Phone: (215)687-9750. Domestic Subscriptions: six issues, $15.00. Application To Mail At Second-Class Postage Rates Is Pending At Wayne, PA 19087 and at additional mailing offices. POSTMASTER: Send address changes to* **Commodore—The Microcomputer Magazine,** *487 Devon Park Drive, Wayne, PA 19087.*

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

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

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

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

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

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

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

Circle #2 on Reader Service Card

Professional Software, Inc. WordPro was designed and written by Steve Punter.

Q&A HOTLINE

scant articles in magazines, etc., there is a lot of power available of which I have no knowledge. Could you please direct me to some good documentation on the Monitor that will explain its functions, capabilities, etc.

L. Holman
Reynoldsburg, OH

A. *Unfortunately, there has never been much printed about the monitor probably because it only has a few simple commands. Commodore calls its version TIM (Terminal Interface Monitor), and all but the very earliest machines have included TIM in the ROM set. TIM allows the user to type commands on the PET/CBM keyboard which will execute a program, display or modify registers and memory locations, and load or save binary data to diskette or tape. When modifying memory, TIS performs automatic read-after-write verification to ensure that the addressed memory exists, is read/write type, and responds correctly. For a thorough explanation of Commodore's Monitor, please turn to page xx in the programming tips section. Hopefully, the information presented will enable you to get some use from a very helpful tool.* C=

Editor's Notes

CLOSEUP: Commodore Magazine's user group profile.

In each issue of this magazine, we publish an updated list of the lengthening ranks of user clubs devoted to Commodore products. In the past six months we have received hundreds of new additions to that list, and the continued response indicates no signs of slowing down. This feedback is strong evidence of the grass roots support that Commodore is developing throughout the microcomputer world. And while some of these Commodore enthusiasts have just hopped aboard the bandwagon, there are other organizations that have been committed to Commodore computers since the introduction of the company's first product.

One such group is the Toronto PET Users Group which was formally established in early 1978, not long after Commodore began delivering its PET microcomputer. Now this Canadian group is 2,000 strong, with worldwide members as far away as New Zealand, and U.S. followers from Maine to Florida. In fact, according to club vice president Chris Bennett, total U.S. membership is nearly 600.

Bolstered by a library of nearly 4,000 software programs on 61 disks, the Toronto PET Users Group is a formidable source of software for all of Commodore's products. Categories in the club's extensive library include utilities, assembler, music, business, games, communications, language, and many more.

Aside from meeting twice weekly at a Toronto area high school, the club has also developed sub-groups that concentrate on more specific products and applications. Current and future groups include machine language, SuperPET, VIC, business, education, and what Bennett labels "hardware hackers." In total, more than 1000 members attend these weekly meetings.

Bennett claims that one of the biggest contributors to the group's success is the tremendous support they have received from Commodore dealers, both local and outside of Toronto. Bennett estimates that "90 to 100 dealers" have devoted their time to the user group. "With that kind of dealer support, it's very easy to establish a club," said Bennett. However, he was also quick to point out that dealers gain just as much by developing a reputation for supporting customers.

Because of its growing reputation and formidable support from dealers, the group has also become a testing ground for many new products BEFORE they are actually marketed. Bennett fondly recalls having WordPro demonstrated to his club on an old 8K PET. The group's roll call also boasts a few well-known Commodore expert, including Jim Butterfield.

Membership costs for the club are $20 per year, which includes a monthly issue of Torpet, the group's newsletter. The club is currently printing 4000 newsletters per month. Paid members also have access to the club's public domain software.

The Toronto PET Users Group is one of the oldest and most successful user groups in North America. Yet, since its inception in 1978, many other user groups throughout Canada and the United States have successfully carved their niche in supporting and educating Commodore users. In future editions of Commodore Magazine we will continue to highlight these groups, whose contributions are essential to the growing ranks of Commodore users. C=

Paul Fleming

Paul Fleming
Editor

Leon Harris is Still 'Pushing' Typewriters; He's Pushing Them Right Out of the Office!

From Where I Sit

Leon Harris
President of Commodore's Personal Computing Division

For the last 22 years, I have been involved in the dynamic growth of the office machine industry. I've traveled from coast to coast and all across the country more times than I can count, all the while ''pushing'' typewriters.

Now, to twist a phrase, I'm still pushing typewriters. The difference is I'm pushing them *out* of the office. As the president of the Personal Computer Division of Commodore, my goal is to help Commodore's expanding network of authorized dealers spread the word that the office of the future is a reality today, and that the *microcomputer* is the real key to efficiency and cost-effectiveness.

Commodore is the leader in the business micro marketplace in Europe and the leader in the home computer industry in the U.S. And we're making strides towards industry leadership in the business market here.

The reason I feel so strongly about microcomputers in general and Commodore micros in particular is simple. For 22 years I was with the Olivetti Corporation, and I got a great feel as to what a quality product was. Now I'm with another international organization that has also made its mark by offering quality products at competitive market prices.

But Commodore micros are more than just quality and price. They are truly keys to helping small businesses and middle managers get a firm grip on controlling costs and increasing productivity. The days of having specialized business machines dedicated to specific functions are gone forever, and micros are a big reason why.

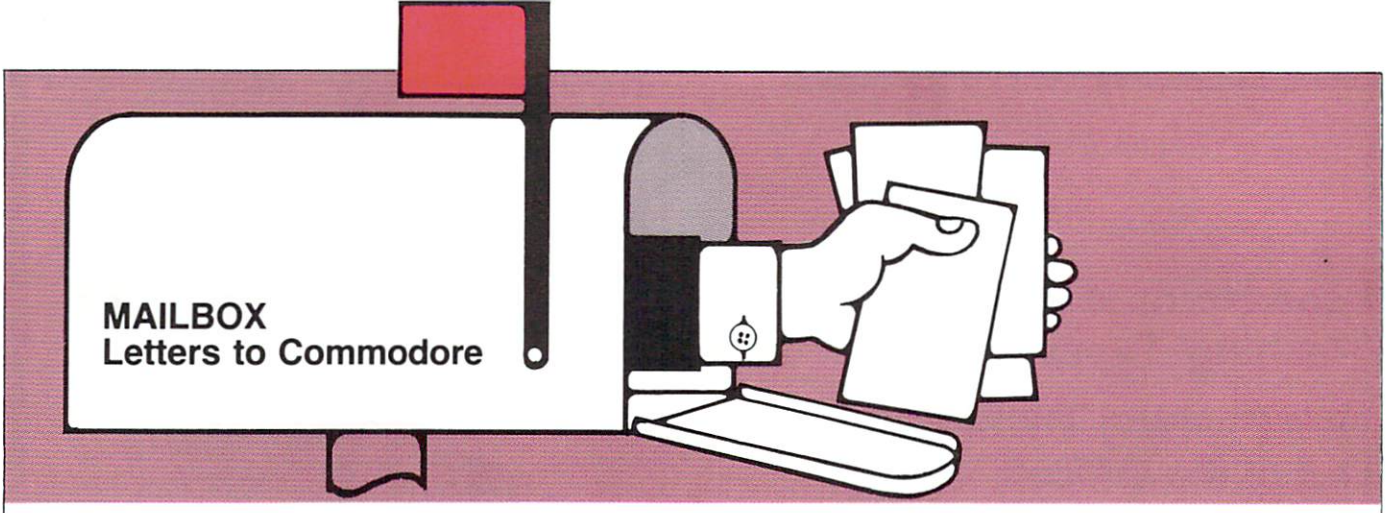
The microcomputer brings versatility and performance to the office of today as well as the office of the future. It can be used as a word processor, data processor, financial forecaster, telecommunicator and more. Why it's truly every bit as capable and cost-effective as the efficiency experts of yesteryear dreamed it would be.

Commodore is going all out to support its growing network of professional dealers because we believe that's the best way to reach small businesses and middle managers. Our dealers are computer and office machines specialists, known in their communities as leaders, and with Commodore micros in stock, they are truly carrying a product of which they can be very proud.

In order to do the job, micros need not be expensive, and need not be complicated to use. In fact, in pioneering the micro industry in Europe, Commodore has proven that quality and price do not necessarily go hand-in-hand with microcomputers. Our products have earned a worldwide reputation for dependability and performance as well as being price leaders. And the term 'user-friendly' was virtually coined for Commodore computers.

Our new Commodore 64 offers computing power never before available at even close to its price, as well as business-oriented software programs that are affordable and easy-to-use. The new Commodore B and BX lines are top-notch in what they offer in terms of price/performance ratios, and our CBM line has become a true workhorse of the industry.

Today, it is obvious that the business machine industry is keeping pace with the space age. And, not surprisingly, the full line of Commodore products is helping to keep the industry in which I've spent my professional life up-to-date with modern computer technology. C=



MAILBOX
Letters to Commodore

We run an HP3000 system for administrative purposes (it's excellent) and use DECwriters, Teleray, and Infoton (now GTC) terminals. We've had ten times as many breakdowns with the timesharing terminals (about 10 or so) compared with over 60 Commodore machines in the system. Keep up the good work!

Thanks,

R. Poland
Richmond Community Schools
Richmond, IN

Dear Sir,

Let me start this letter off by telling you how much I enjoy COMMODORE MAGAZINE! I especially like the "Programmer's Tips" and "Excerpts from a Technical Notebook." Please keep up the good work (and give these two columns more space!).

I am writing specifically to comment on the excellent article , "An EASY Cursor Positioning Routine" (April/May issue). I too was interested in this concept some months ago and spent a great deal of time trying to conquer ths problem. However I think my results might surprise you!

Your program occupies about 256 bytes in memory. Here's a way to do it much faster.

1) Load $DB with the desired row.
2) Load $C6 with the desired column.
3) Call routine $E067.

That's it! The cursor will now be on the desired row and column. The author mentioned that he tried steps 1) and 2), above, and didn't get any results. This is because step 3) is the crucial step. The way this works is that after $D8 and $C6 have been set up, routine $E067 calculates the actual location in screen memory to be written to and stores the result in $C4, $C5 and $C6. It does this by means of two look-up tables located at $E755 and $E76E. These tables provide the proper screen address to the cursor.

I realize that this is a somewhat sketchy explanation; however, a look at the disassembly for the ROM's will make it all clear.

Consider me a devout Commodore supporter.

Sincerely,

Thomas Henry
Mankato, MN

Dear Sir:

It's very seldom I undertake to write a letter of this type but I simply have to tell you of the effectiveness of your publications Commodore and the newly issued Power/Play.

I have not yet purchased my computer for personal/business applications but I customarily subscribe to magazines that will give me background on major purchases so that I have a feel for what is available and at least some superficial understanding of available products. I subscribe to BYTE, Personal Computing, Commodore and hope to continue receiving Power/Play.

My congratulations to Mr. David A. Kaminer, Editorial Manager, and Editors Diane LeBold and Paul Fleming for presenting publications that are really excellent and (mostly) understandable to neophytes like myself.

Sincerely,

Julius Brodsky

Dear Sir:

My name has been in the users club listing for a few months now, and the response and enthusiasm have been overwhelming! I am receiving many letters and phone calls from all over the world. So much so, that I've had to change my phone number . . .

HAPPY COMPUTING!
Paul V. Muffuletto

A BYTE OF WHIMSEY...

Actual Poster Size 29" wide x 24" deep

A funny thing happened here a few months ago...this fellow walked in with this very unsophisticated art he wanted us to use on the April/May cover of the COMMODORE Magazine. Gee, how could we tell him that it wasn't the "image" we wanted to project...after all, our computers might be *FUN*, but we have some very professional people using these things — doctors, lawyers, even indian chiefs. The powers that be took one look at it and said "Hey, you can't put a lot of pixies on our cover and expect people to take you seriously... there's a pixie playing frisbee with one of our disks, and another pixie floating down a waterfall with one of our business machines, there's a funny looking toad fooling around with a VIC, there's even some pixie who thinks he can access a data base from his treehouse... our readers want to see *PEOPLE* — people looking very serious, very productive, learning all sorts of great things, not a bunch of silly pixies. Well, we took a closer look, and ran it anyway...In fact, we even had it printed as a big four-color poster to hang in your *VERY* serious office, institution of learning, or, your own treehouse... So if you have just a BYTE OF WHIMSEY in you, fill out the attached coupon and send us your check or money order.

Return to:
COMMODORE MAGAZINE
COMMODORE BUSINESS MACHINES INC.
The Meadows, 487 Devon Park Drive
Wayne, Pa. 19087

Name_____

Address _____

City _____

State_____ Zip_____

Please enclose a check or money order for $5.00 for each poster ordered.

QUANTITY____ TOTAL ENCLOSED____

Dealer Inquiries Welcome

Circle #5 on Reader Service Card

commodore COMPUTER

Commodore Introduces 'Easy' Spreadsheet Programs for SuperPET and COMMODORE 64 Microcomputers

A "family" of four "Easy" spreadsheet programs, designed to provide Commodore users with lightning-fast decision-making tools to be used in a business environment, has been introduced by Commodore Business Machines, Inc.

The four new programs, EasyCalc™, EasyPlot™, EasyFinance™ and EasyScan™, will be available this summer, and will initially run on SuperPET and Commodore 64 micros. In the near future, the "Easy" programs will also be available for other Commodore micros including the recently-announced "P" and "B" series.

EasyCalc is the biggest micro worksheet available with 65 columns and 999 rows. It replaces old fashioned paper, pen and calculator and saves many hours in solving a wide range of numerical problems such as budgets, cash flow forecasting, product and resource planning, stock fluctuations, engineering or scientific data, trend analysis, agricultural yields and floor plan models.

EasyCalc has "help" functions to aid first-time users and experts alike. It has the ability to print out all formulas and assumptions, and its disk-based grid allows automatic matrix consolidations. Additional features include movement of data in a matrix, selective row reporting and printing, instant "what if" calculations, and integration with Commodore's EasyPlot spreadsheet program.

EasyCalc will be available from authorized Commodore dealers throughout the U.S. for $149.95 for the SuperPET and $99.95 for the Commodore 64.

Commodore's EasyPlot allows users to interpret and present numerical data with high quality charts and graphs, and produces bar and pie charts, scatter diagrams and line graphs. It analyzes relationships between different data sets, and charts population trends and stock price fluctuations as well as other numerical information. The user-oriented format of EasyPlot features full-page printing of all charts and graphs and integration with EasyCalc.

EasyPlot will be available for the SuperPET for $149.95 and for the Commodore 64 for $99.95.

EasyFinance from Commodore allows users to analyze alternatives and see pros and cons so vital in decision-making. Its features include automatic buy vs. lease analysis, full loan amortization functions, present and future cash flow calculations and discounted cash flow analysis to help make decisions regarding borrowing, tax deductions and the cost of inflation.

Commodore dealers will offer EasyFinance for the SuperPET for $124.95 and for the Commodore 64 for $79.95.

The fourth member of the new spreadsheet family from Commodore, EasyScan, is an instant access diary that never lets you forget. It features day, week, month or year at-a-glance, organization of data by priorities and "zoom" for in-depth scan. EasyScan allows users to schedule time and resources for optimum productivity and also serves as an appointment secretary and project planner/task coordinator.

EasyScan will be available from authorized Commodore dealers for $124.95 for the SuperPET and $79.95 for the Commodore 64.

The Commodore Information Network . . . Is Here . . . NOW!

The COMMODORE INFORMATION NETWORK is a reality! Designed for all beginners, computer telecommunications fanatics and anyone who fits in between, the network is ready to provide solutions. There are many manufacturers who say teletex is the wave of the future and promise to support it. Here at Commodore we do more than pay lip service to these comments; we feel the Commodore Information Network on CompuServe is the biggest and the best of all the manufacturer supported services! Since its June 1 premiere, Commodore's network has generated tremendous interest and user participation. At the NCC show in Chicago the network (and the VICMODEM) shared center stage with Commodore's new computer products.

And you've surprised us with your incredible response. Already, the number of people joining the Commodore Information Network has surpassed all our projections. Why? Because we are giving you what you deserve . . . service! In this issue, I'd like to give you an overview of our Network and it's future goals.

The Commodore Information Network has been created to give you the information you want as quickly and efficiently as possible. And what better medium is there than the computer itself? Working jointly with CompuServe, we are committed to bring you information, services and programs through the most sophisticated and convenient medium ever devised—telecommunications.

At the present time the network is developing in three stages: the information database, the SIG (Special Interest Group) bulletin board and public domain software. The first step for any user to access our database is to log onto the CompuServe system, (incidentally you receive a free subscription when you buy a VICMODEM). After the first CompuServe prompt, type GO CBM. This is what you will see as you enter the new and exciting world of the Commodore Information Network:

CompuServe CBM-1

Commodore Information Network Main Menu

1 Information
2 HOTLINE
3 Commodore News
4 Directories
5 Commodore Tips

1. The "Information" section offers you an explanation of the commands that are available in the database, a listing of direct access codes, and a description of the major sections of the database.
2. The "HOTLINE" gives you direct access to Commodore. You send us a question or inquiry and receive an answer in your "electronic mailbox."
3. The "Commodore News" section contains application articles from users around the world, the latest product announcements, and our bulletin board.
4. The "Directories" section includes listings of dealers and user groups across the country . . . plus educational resource centers.
5. 'Commodore Tips" is for your enjoyment. Whether you are a novice or advanced computerist, this section has something for you. Answers to questions is the name of the game, and your most frequently asked questions (and some you never thought of) are presented along with comprehensive answers.

You can choose any portion of information you want by typing the appropriate number and pressing <RETURN>. For example, if you select number 3 for Commodore News <RETURN>, the following will appear on your screen.

Commodore News

1 Public Bulletin Board
2 Magazines and Newsletters
3 Product Announcements
4 Descriptions of Publications

In the next few months, we will include a Commodore business section (including corporate reports and stock quotes), a listing of Commodore related books and reviews, a complete retail price list, a listing of manual errata on all manuals and—for the engineers in the group—semiconductor specifications.

On the SIG (Special Interest Group) bulletin board, you are invited to attend some very serious, very informative and very enjoyable Commodore user group meetings. And no matter where you live, you're as close as a trip to your computer. Sit down, get comfortable, sign onto CompuServe and at the first prompt type GO CBM310. You will also find us listed in the Home Services (HOM50) and the Personal Computer Services (PCS50) sections of CompuServe. Here you will be able to trade ideas, sell equipment, join a user group, and get answers to those hard questions. Hotline questions and answers can also be found here.

Do you have an article or program looking for a place to be published? The Commodore Information Network will serve as a forum for your work. Take advantage of the opportunities to read what others are discovering or saying about Commodore computers. To submit an article, program or newsletter, just send it to: Editor, Commodore Information Network, 487 Devon Park Drive, Wayne, Pennsylvania 19087.

The third portion of the Network, tentatively scheduled for January, is the free public domain software including the →

COMMODORE NEWS

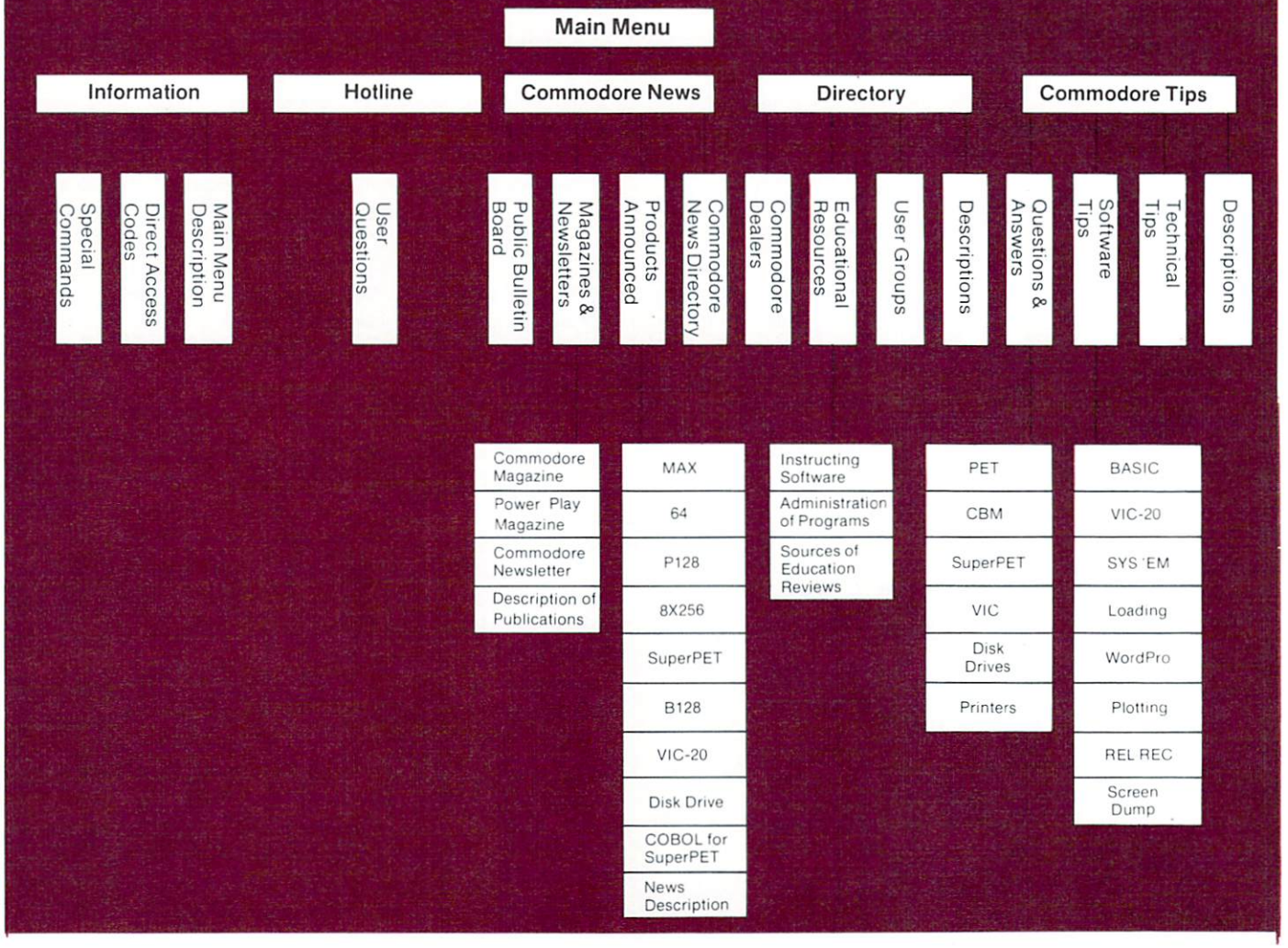
Commodore program of the month. Commodore is preparing a number of programs that users can download into their system.

Once a month we will choose an outstanding piece of software and offer it for sale on the network. We will consider user software submitted to us. The submitted software could be totally original software or an existing program translated to another Commodore computer model. The programmer, of course, will receive a royalty for all software sold.

The Commodore Information Network is co-ordinating its efforts with Commodore and Power/Play magazines so that any programs that are printed in the magazines are also available for downloading to your computer. This feature will eliminate the trouble of typing and debugging.

To use the graphics, sound, uploading, and downloading capabilities, you must have software that uses telecommunications protocols. The software that will enable you to have all these capabilities is presently under development for the VIC (and later will be available for other Commodore computers). Commodore has and will continue to support the best telecommunications network because *you* need it. And, as telecommunications services become more sophisticated we will provide our users with the most up-to-date service available. C

—Jeff Hand



'Silicon Office' on Commodore Micros Performs Variety of Business Functions

Silicon Office, a "three-in-one" software package providing cost-effective information management, word processing and communications functions, is now available to satisfy the most pressing needs of almost any business office.

Designed for use with a Commodore CBM 8032 upgraded to 96K, Silicon Office includes all the functions of an integrated data base management system as well as a word processor.

Silicon Office retails for $999.00 and is available from authorized Commodore dealers coast-to-coast.

"In today's economy, it's especially important that a business make the most of its equipment and investments," said Leon Harris, president of the Personal Computer Division of Commodore. "The Silicon Office and an upgraded CBM 8032 provides state-of-the-art capabilities and ease-of-use, and performs a wide variety of office functions at a very cost-effective price."

The flexible Silicon Office system enables users to custom-design file formats as well as easily call up stored information and individual records by name. Silicon Office can search through six different files at any one time to provide fast retrieval of information, and records can be readily added, deleted or updated.

The built-in word processing system allows for easy creation, storage, editing and printing of office documents including invoices and letters.

Designed as a "screen-based virtual memory system," the word processing segment of Silicon Office prints out letters and/or figures exactly as they appear on the screen. In addition, the text is automatically adjusted whenever additions or deletions are made. Silicon Office can handle text from 20 to 125 characters in width, and provides for creation of multi-columns of text.

Because Silicon Office is an integrated system, all calculations can be used in conjunction with the word processing facility, which can print results in any format desired. Using information from any of the system's files, Silicon Office can perform most mathematical functions including complex calculations.

The system also supports communications between computer systems in the form of file transfer, direct communication, and data sharing. With the use of a modem, two-way communications between Silicon Office systems is possible over normal telephone lines.

Another key feature of the unique three-in-one office system is programmability. Short simple programs can be written into the system's memory, and Silicon Office will then perform these functions automatically. Because of the system's extensive memory capabilities, many of these short programs can be conveniently stored.

Editor's Note: See a complete review of Silicon Office on page 86 in this issue.

SuperPET Update

From its inception there has been an abundance of information as well as misinformation about the SuperPET. This is the first of an ongoing series of articles to help support the end user in understanding the SuperPET. The information in these articles will be as accurate as possible, drawing from both internal and external sources.

In the future, I would like to accomplish three things in each edition: present something new about the SuperPET (new software, technical update, unique application); provide a more in depth explanation of information presented in the Commodore News section concerning the SuperPET; and respond to more frequently asked questions about the use of the languages and communication capabilities.

Articles by end users for end users, would also be included. These could be articles about such things as interfacing little known peripherals to the SuperPET or a particular software application. This is an open invitation to all APL programmers to submit those bizarre, left field, single line functions.

COBOL

Having stated the charter, Yes Virginia, there really is a Waterloo MicroCOBOL interpreter for the SuperPET and it is available now. Even as this article is going to print, our Software Group is preparing diskettes which will contain the Waterloo MicroCOBOL interpreter, updates to existing Waterloo Micro language interpreters, and Waterloo MicroCOBOL program examples. The packet (P/N 970142) will contain the following: Two (2) diskettes (WCS,P/N 970144 and Tutorial,P/N 970145); documentation updates (replacement pages) for the MicroEditor, MicroBASIC, MicroPASCAL, update information for MicroAPL; and erratum sheet for MicroCOBOL and Waterloo MicroCOBOL manual (P/N E81909, cost $9.95). The cost for this packet which will be available through local dealers, is $29.95.

Using that Extra Memory

For those hardcore 6502 basic programmers who want to use the additional 64K of memory but are not sure how, you can find out by reading Paul Donato's article "Run 96K Programs on the SuperPET" which appeared on page 139 in the June, 1982 issue of COMPUTE.

APL PRINTER

APL programmers, there is a printer available. Actually, it has been here all along, we just didn't know how to access it correctly. It's our own Commodore 8300P. There are several APL print wheels available. Diablo has both a plastic (P/N 38150-01) and a metalized version (P/N 311951-01). If ordering through Xerox, use P/N 9R21135 for their metalized print wheel. To list the tutorial function 'APL.MASTERMIND' do the following:

```
)LOAD APL.MASTERMIND
'IEEE4'□CREATE 4
A←□CR'MASTERMIND'
B←A,□TC[7]
C←B,□TC[3]
(□XR,C)□PUT 4
```

Single Board Upgrade (8032 to SuperPET)

The single board upgrade to make the 8032 a SuperPET is available in kit form now from your local dealers. It will contain all the necessary hardware and instructions as well as the Waterloo Micro language disk, language tutorial disk and associated manuals. When ordering use P/N 9000035-01. Cost of the entire packet is $795.00.

ROM Protected Software

If you are having problems using ROM protected software (three board version SuperPET), your dealer has a "ROM Select Modification Kit" (P/N 9000029). It's free. The two board version SuperPET has two ROM ports, U-45 and U-46 for ROM protected software, but to use the U-46 port for other then EPROMs, cut the trace from pin 21 to grounds then jump pin 21 to pin 24. This problem will be corrected on REV 'C' boards.

Accessing the Serial Port

The 'serial' port of the SuperPET can be accessed in three different ways—(1) within a program by using the filename 'serial', (2) via the monitor in 'passthru' mode or (3) via the editor in 'talk' mode.

More Software

Software that takes advantage of the additional memory has been, up to this time, in short supply. Commodore has announced the introduction of the 'EASY' group of programs designed as aids for efficient time management, and 'what if . . .' calculations. The group consists of EasyCalc, EasyPlot, EasyScan, and EasyCash. Information on these products will be available through local dealers by mid September.

If you have questions or would like to contribute to this section of the Commodore Magazine please use the following address:

Walt Kutz
SuperPET product manager, U.S.
487 Devon Park Drive
Wayne, PA 19087 C=

CMS ACCOUNTING SYSTEM

CMS GENERAL ACCOUNTING SYSTEM

A fully interactive General Accounting System designed especially for the first time user. All input requests are fully prompted with complete verification of input data. Most reports may be printed either to the screen or the printer and started or stopped at any point. The user is led completely through each function by a series of highlighted prompts fully explaining the required input at each point. A professionally written instruction manual is included which shows sample reports generated by the system and further explains each step and prompt as it is encountered by the user. These user prompts, together with the detailed step by step manual, make the system extremely user friendly.

GENERAL LEDGER

Up to 1000 accounts on the Chart of Accounts. Fully departmentalized up to ninety-nine departments. A Cash Receipts and Disbursements Journal allows each check or deposit to be distributed to nine separate General Ledger accounts. Non cash postings may be entered through the General Journal. The Cash Journal and General Journal may be printed at any time either to the screen or to the printer. A non structured Chart of Accounts allows easy tailoring of the Financial Statements to fit varying user needs. Account balances are maintained for current month, quarter to date, and year to date. A detailed Budget Analysis Report may be printed at any time either to the screen or to the printer. The General Ledger program may be used alone or may be set up to accept postings from Accounts Receivable, Accounts Payable, Payroll, or other programs.

PAYROLL

Maintains current, quarterly, and yearly totals for 350 employees. Prints Payroll checks with full deduction and pay detail as well as year to date totals. Accommodates weekly, bi-weekly, semi-monthly, and monthly employees. Pays regular, overtime, holiday, and piecework hours. Handles federal and state deductions plus up to eight miscellaneous deductions and payments per employee. Prints Payroll Journal, Check Register, and an Absentee Report as well as state and federal 941 information and W-2 forms. The Payroll program may be used alone or set up to automatically update the General Ledger.

ACCOUNTS RECEIVABLE

Allows a total of 1800 invoices and customers per 8050 data disk. Each invoice line item may be distributed to a separate General Ledger income account. Prints invoices and monthly statements and allows for individualized messages on the statements. Finance charges may be automatically added to overdue invoices at a user determined rate and period. Complete invoice aging reports with user set aging breaks may be printed at any time either to the screen or to the printer. A special Overdue Invoice Report is also available for close monitoring of receivables. Handles credit memos as well as invoices. A Sales Tax Report is available showing total sales and sales taxes payable for up to nine separate sales tax rates. Total sales and sales commissions earned are automatically tracked for up to fifteen salesmen. The Accounts Receivable program may be used alone or set up to automatically update the General Ledger.

ACCOUNTS PAYABLE

Allows a total of 2500 invoices and vendors per 8050 data disk. Each invoice may be distributed among nine separate General Ledger expense accounts. Prints checks with complete voucher detail for up to thirteen invoices per check. Invoices may be paid by vendor range or by individual invoice numbers. Credit memos and term discounts are taken automatically. Full or partial payments may be made for each invoice. A Check Register may be printed at any time. Complete aging reports with user set aging breaks may be printed at any time either to the screen or to the printer. A special Cash Requirements Analysis Report is available to assist in determining upcoming cash requirements. The Accounts Payable program may be used alone or may be set up to automatically update the General Ledger.

OFFICIALLY APPROVED BY COMMODORE

CMS Software Systems

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

Circle #10 on Reader Service Card

POWER/PLAY TAKES OFF

POWER/PLAY, the new magazine devoted exclusively to Commodore home computerists, is getting an enthusiastic response from its readers. In fact, our circulation manager, John O'Brien, now takes a cardboard carton with him when he goes to get his mail so he can carry all the new subscriptions back to his tiny (and increasingly crowded) office. Except for occasional incoherent babbling, he seems to be handling it OK. All you new subscribers, let's hear it for John—and assistant Diane Ottinger—who are performing like real troupers, getting all those subscriptions processed.

The next issue, appearing in September, should delight all VIC 20 enthusiasts. Scott Adams' Adventure Games afficianados will be thrilled with an excerpt from Scott's *Book of Hints*. And bargain hunters will be glad to discover a new computer shopping service that offers name-brand products at substantial discounts. Freelance expert David Malmberg talks about teaching kids programming through the use of Turtle Language. And syndicated columnist DX Fenten examines the pros and cons in the war against video games. And that's just for starters.

The meat of the September issue, as usual, will be a lot of solid information on how to get the most out of your Commodore home computer. That includes, among other things, technical tips, program listings to type and save, product bulletins, and a run-down on the new Commodore Information Network, now available via the CompuServe telecommunications database. Not only that, but Game Group programmer Rick Cotton clears up a lot of questions about what happens to memory locations when you add a memory expander to your VIC.

Look for POWER/PLAY at your Commodore dealer. Or, better yet, why not subscribe. We'll just get O'Brien a bigger box.

1981 Commodore Bibliography

The following is a bibliography of Commodore related articles published in 1981 in 'Creative Computing', 'Kilobaud Microcomputing' and 'COMPUTE!'.

Creative Computing 1981

January
- 24 No PET Peeves—New Computers From Commodore
- 156 Personal Electronic Transactions

February
- 18 Music Editors for Personal Computers
- 154 Personal Electronic Transactions

March
- 26 Wordpro 1 vs CMC
- 78 A PET Lizzard (Game Listing)
- 166 Reading Level: Determination & Evaluation (Listing)

April
- 222 Personal Electronic Transactions

May
- 96 Break Even Analysis With Visicalc
- 208 Personal Electronic Transactions

June
- 26 The TNW 2000
- 30 Fantasy Games
- 36 Computer Warfare
- 88 Software Techniques of Digital Music Synthesis Pt. 1

July
- 50 The Last One
- 122 Tree (Game Listing)
- 144 PET Nuclear Power Plant (Game Listing)

September - Buyer's Guide
- 43 Commodore VIC-20

October
- 54 Educational Software and Books
- 160 Bombproofing the PET INPUT Statement

November
- 65 Dynacomp Bridge Challenger
- 192 PET Screen Line Length

December
- 76 Valdez: A Supertanker Simulation
- 240 Helping Students Think About Marriage & Education

Kilobaud Microcomputing 1981

January
- 10 PET-Pourri — New PET Monitor
- 48 Real-Time Spectrum Analyser
- 78 Scramble (Game Listing)
- 188 Second Cassette Interface (Hardware Modification)

February
- 12 PET-Pourri — Jinsam
- 56 London Computer Club a Huge Success
- 72 Portrait Of A Dynamic French Company

March
- 8 PET-Pourri — Handy Utilities
- 144 PET Shorthand Compleat

April
- 10 PET-Pourri — VIC-20 Debuts

May
- 10 PET-Pourri — Conversing in Assembly Language
- 179 A PIE Taster's Report
- 195 Soulful Software Sounds (Listing)
- 200 Find That Program! (Listing)

June
- 10 PET-Pourri - ROM Packages From Skyles
- 92 Once Upon A Time (Listing)
- 177 Expand PET Memory (Listing)

July
- 10 PET-Pourri — 8032 Data Handlers
- 104 A One-Two Punch For CBM/PET Graphics
- 167 Get On The PET Instrument Bus (Listing)

August
- 10 PET-Pourri — Commodore Colors NCC
- 152 What's The Difference (Listing)

September
- 12 PET-Pourri — CBM Utilities

October
- 10 PET-Pourri — Commodore's Big Push
- 195 PET Goes To The Polls (Listing)

November
- 14 PET-Pourri — VIC Expands Its Horizons
- 50 Popping and Pushing Permutations In BASIC

December
- 10 PET-Pourri — Word Pro Enhancement
- 66 Putting The Joy Back Into Programming (Listing)
- 114 A BASIC Assembler For The PET (Listing)
- 178 Put To The Test By A Computer

COMPUTE!

January
- 32 The Mysterious & Unpredicatable RND Pt. 1
- 38 CURSOR Classifications Revisited
- 44 ODDS & ENDS..re PET cassette tape
- 92 The Screen Squeeze Fix For CBM 8000
- 96 Horray For SYS
- 102 Machine Language: Scanning The Stack
- 108 The PET Revealed & Library Of PET Subroutines
- 110 A Visible Music Monitor
- 112 Disk-O-Prop
- 114 Detecting Loading Problems & Correcting Alignment
- 118 Spooling For PET With 2040 Disk Drive
- 118 Variable Dump For New ROM PETs
- 120 The 32K Bug
- 121 An Ideal Machine Language Save For The PET
- 122 PET Metronome
- 123 PET IEEE Bus: Standing Room Only?
- 124 PET/CBM IEEE Bus Error

February
- 16 LED—A Line-Oriented Text Editor
- 30 Simulated PRINT USING
- 34 The Mysterious & Unpredictable RND Pt. 2
- 54 Basic Math For Fun & Profit
- 60 PET Spelling Lessons Your Student Can Prepare
- 97 Contour Plotting
- 103 Relocate
- 104 Mixing & Matching Commodore Disk Systems
- 109 Memory Calendar
- 114 Crash Prevention On The PET
- 116 Machine Language Printer Command
- 118 ODDS & ENDS On PET/CBM Files
- 120 Three PET Tricks
- 124 PASCAL On The PET
- 126 The PEDISK
- 127 A Disk Operating System For the CGRS PEDISK

March
- 20 Taking The Plunge—Machine Language Programming
- 42 Getting The Most From Your PET Cassette Deck
- 48 The Mysterious & Unpredictable RND Pt. 3

→

54 A CAI Program Called LINEAR EQUATION
92 Keyprint Revisited
96 Learning About Garbage Collection
102 PET Machine Language Graphics
112 Disk File Recovery Program
124 PET Exec Hello
130 A Flexible Input Subroutine
132 Universal Tape Append for PET/CBM

April
26 Commodore VIC-20: A First Look
34 How To Be A VIC Expert
46 The Mysterious & Unpredictable RND Pt. 4
52 Micros With The Handicapped
56 Matrix Row Operations
117 Partition and Load
122 Relative File Mechanics
126 COPLOT
130 ROM Expansion For The Commodore PET
136 Working With BASIC 4.0
138 Papermate Word Processor
142 Dissecting C. W. Moser's ASSM/TED 1.0
144 PET File I/O and Machine Language
146 How To Get Started In Machine Code And Not Go Crazy With A Routine For Two Joysticks
152 Machine Language: The Wonderful Wedge

May
22 The Mysterious & Unpredictable RND Pt. 5
30 Land Of The Lost — Cassette Filing System
46 EPIDEMIC
58 Naming Compounds
96 A Fast Visible Memory Dump
112 Getting To The Machine Language Program
116 A Thirteen Line BASIC Delete
118 Calculated Bar Graph Routines On The PET
120 The Revised PET/CBM Personal Computer Guide
124 Un-Compactor
126 Using The Hardware Interrupt Vector On The PET
128 PET As An IEEE-488 Logic Analyser
130 Running 40 Column Programs On A 8032

June
4 RAM/ROMs—A New Style Of Memory?
22 Mapping & Modifying Unknown Machine Language
52 Ideal-Gas Law
94 Relocation of BASIC Programs On The PET
98 Memory Partition Of BASIC Workspace
100 Machine Language Code For Appending Disk Files
102 Quadra-PET: Multitasking On Your PET
106 PET/CBM Disk Formats
110 Interfacing With The User of Your PET Programs
116 Keeping Tabs On Your Printer
120 Assembler In BASIC For The PET
128 Uncrashing
130 Notes On The PET SAVE Command
131 Optimized Data System PH-001 2114 RAM Adapter
132 Discovering Tape File Names
132 Petbug
133 Machine Language Utility Pac

July
142 Saving ML Programs on PET Tape Headers
146 Commodore ROM Systems: Terminology
150 SCREENER:4 Screen Utility Routines
155 Machine Language: Comparison Shopping
156 Using TAB, SPC and LEN

August
30 Minimize Code And Maximize Speed
50 Add A Programmable Sound Generator
105 The CBM "Fat 40" — Boon Or Bane?
109 Digital Arrayment
120 Keyword
124 CBM/PET Loading, Chaining, and Overlaying
128 Converting PET BASIC Programs To ASCII Files

August — Home & Educational Computing
4 Exploring The Rainbow Machine
9 VIC As A Super Calculator
11 Custom Characters For The VIC
16 The Confusing Quote

September
30 The Column Calculator
36 PET, Atari, Apple: On Speaking Terms
103 The Unwedge-Tape Append And Renumber
108 STP-488 A Smart Terminal Program
118 4.0 Garbage Collection: A Small Bug
120 Using The Monitor On The PET
122 Odds And Ends: Relative Files on BASIC 3.0
124 2040 Disk Program Listing
128 All About LOADing PET Cassettes
134 Graph Plotting Routine
136 Linelist

October
28 VIC-20 News
30 Various VIC Memory Locations
30 Update Floating Color, Floating Screen
48 More Machine Language For Beginners
62 Undeletable Lines
126 Practical PET Printing Primer
132 A Fat Forty Bug
138 Train Your PET To Run VIC Programs
140 Converting To Fat-40
143 High Resolution Bar Graphs For The PET
146 Waking Up The PET Screen
149 Interfacing A BSR X-10 AC Remote Control System
156 Using Non-Pin-Feed Forms in Tractor Printer
159 Why You Should Use PEEK(155) Instead of GET

November
26 A Flower Sale Program
28 SuperPET's Super Software
38 SuperPET: A Preview
54 Bits, Bytes and Basic Boole
136 POWER
142 The PET Speaks
145 Machine Language: Monitoring Progress
147 Directory For 3.0
148 Inversion Partitioning
151 A Personal News Service
155 FOR/NEXT GOSUB/RETURN, And The Stack

December
38 Subscript Heap Sort
54 Maze Generator
130 A Look At SuperPET
134 SUPERMON
142 PET To PET Communication Over The User Port
150 Replacing The INPUT# Command
154 Typing Foreign Language Text With The CBM Printer
158 Three Reviews: Superchip, Spacemaker, Sort
160 Machine Language: Jumbo Numbers
163 File Recovery
166 Looney Line Numbers
168 Mine Maze
172 COMAL: Another Language

Back Issues

Recently we've been flooded with requests for back issues of Commodore's user publications. Two of the three editions published under the name INTERFACE can still be purchased, and all five Commodore Magazines are available. CAUTION—supplies are limited and issues will be sent on a first-come, first-serve basis. If we do not have the issue(s) you ordered we will credit your subscription for the amount sent. Outside the U.S. please add $1.00 for each magazine ordered. All orders must be prepaid with check or money order. Please send to the following address:

Commodore Magazine
The Meadows
487 Devon Park Drive
Wayne, PA 19087

Yes, please send the back issues checked.

Enclosed is a check or money order for $_____ for _____ back issues.

Make checks payable to Commodore Business Machines, Inc., The Meadows, 487 Devon Park Drive, Wayne, PA 19087.

- ☐ **February INTERFACE**
- ☐ **May INTERFACE**
- ☐ **October Commodore Magazine**
- ☐ **December Commodore Magazine**
- ☐ **February Commodore Magazine**
- ☐ **April/May Commodore Magazine**
- ☐ **June/July Commodore Magazine**

Everything You Always Wanted To Know About Commodore Computers *

*And Asked!

PET

1. Q: Is Pascal available for the 4016 computer?

A: Pascal is available for the 4016 computer from Abacus Software, which is located in Grand Rapids, Michigan at (616) 241-5510. This package is called Tiny Pascal Plus.

2. Q: Where can a video adapter for a large-screen PET be obtained?

A: A video adapter for a large-screen PET is available from Madison Computer, which is located in Madison, Wisconsin at (608) 255-5552.

3. Q: Can a joystick be interfaced with the 4032 computer?

A: It is possible to interface a joystick with the 4032 through the user port. However, the software to do this must be developed by the individual who is doing the interfacing.

4. Q: How many disk drives can be attached to the 8K PET computer?

A: If the 8K PET has the upgrade ROMs, then as many as eight disk drives can be daisy-chained together.

5. Q: Can the IEEE port on a PET be interfaced to an RS232 port?

A: Devices that convert the IEEE port of the PET to an RS232 port can be obtained from TNW (called TNW 2000) at (714) 225-1040, or from Connecticut Microcomputer (called ADA 1450 or SADI) at (203) 775-4595. Also, Madison Computer at (608) 255-5552, can provide a program called McTerm, which converts the user port to an RS232 port.

CBM

1. Q: Can the character set on the 8032 computer be changed?

A: The character set on the 8032 computer can be changed by changing the ROM in UA 3 slot. Character information is contained in the ROM and can be replaced with an EPROM or ROM from companies that sell custom character generator ROMs.

2. Q: Is the character generator on the 8032 addressable?

A: The character generator is not directly addressable, but it is connected to the CPU chip by its own special "character bus." This technique saves valuable addressable memory space.

3. Q: Is it possible to do word processing in Spanish on the CBM computer?

A: Yes. A special character set is needed in the character generator ROM. These ROMs are available through West River Electronics Research & Development, which is located in New York.

4. Q: Is a BASIC compiler available for the 8032 or the 4032? How many times faster does it execute the program?

A: Yes, a BASIC compiler is available for the 8032 and the 4032. The Commodore Integer BASIC Compiler runs about 100 times faster than BASIC, but it only handles integer math. It generates true machine code and is best for compiling subroutines. Another compiler is PETSpeed, which generates pseudocode and increases execution speed by a factor of approximately 3 to 20 times, depending on the task. It handles floating point math, and is easier to use to compile previously written programs since it handles PET/CBM BASIC syntax.

5. Q: What assembler is available for the 8032 computer?

A: The Commodore Assembler Development is available through Authorized Commodore Dealers. Assemblers are also available from several other sources.

6. Q: Can the 8032 computer be upgraded to a SuperPET?

A: The SuperPET requires an additional board, which is not currently available for the 8032 computer as a separate item.

7. Q: Which port can be used on the CBM to interface it to a digital voltage meter?

A: Either the user port or the IEEE

port can be used to interface the CBM with a digital voltage meter or similar devices.

8. Q: Is training available for customers on CBM machines?

A: Many Authorized Commodore Dealers provide training on Commodore equipment to their customers. For information regarding training, contact your local Commodore dealer.

SuperPET

1. Q: Which modem does Commodore recommend to be used with the SuperPET?

A: No specific modem is recommended for use with the SuperPET. Any good quality RS232 modem will work well with the SuperPET.

2. Q: How many characters can fit within a logical record on the SuperPET?

A: On the SuperPET, 254 characters can fit within a logical record.

3. Q: How does the 8096 computer compare with the SuperPET given that the price is the same for both computers?

A: The 8096 is an expanded 8032 computer. It does not have the 6809 chip in it, and the Waterloo micro languages will not run on it. The SuperPET has the additional 6809 microprocessor, a serial port (RS232), and Waterloo micro language software.

4. Q: Can the SuperPET be expanded beyond 96K RAM?

A: No. The maximum amount of RAM allowed for the SuperPET is 96K.

5. Q: Could a hard disk unit be interfaced with the SuperPET?

A: Yes. Hard disks can be interfaced the SuperPET and most other Commodore computers.

6. Q: How can I print to the printer in BASIC using the 6809 chip on the SuperPET?

A: To print from a BASIC program, use the OPEN statement with 'printer' as the file name (see pages 166-170 of the Waterloo microBASIC book). To LIST a program while using BASIC, enter: save 'printer'.

7. Q: How do you run an 8032 program that needs ROM slots UD 11 and UD 12 on the SuperPET?

A: A retrofit kit number 9000029 is needed on original double-board SuperPETs to run 8032 programs using ROM slots UD 11 and UD 12. This kit is available at no charge from an Authorized Commodore Dealer.

VIC 20 Personal Computer

1. Q: Can VICMON and the Super Expander be used at the same time with the expansion module?

A: No. Although some functions may operate correctly, some memory locations in each cartridge are the same, which would interfere with programming.

2. Q: Can scrolling be made to reverse itself on the VIC 20?

A: It is possible to make scrolling reverse itself with a machine language routine. However, there is no routine located in ROM that will accomplish this. The routine must be developed by individuals who wish to make scrolling reverse itself.

3. Q: Are any commands assigned to function keys in the Programmer's Aid Cartridge?

A: Yes. There are two sets of commands assigned to the function keys in the Programmer's Aid Cartridge. However, these commands can be redefined.

4. Q: Is it possible to restore the VIC back to the unexpanded VIC with the expander cartridges plugged in?

A: No, it is not possible to restore the VIC back to the unexpanded VIC with the expander cartridges plugged in. The cartridges must be un-plugged to restore the VIC back to its unexpanded mode.

5. Q: What controls the clock on the serial I/O port of the VIC 20?

A: The 6522 chip in the VIC has a crystal clock in it which controls the rate of transfer on the serial I/O port.

6. Q: Where is the start of BASIC located when the 8K and 16K Memory Expander cartridges are plugged in?

A: BASIC starts at 4096 decimal with the 8K and 16K Memory Expander cartridges plugged in.

7. Q: Is the user port of the VIC 20 analog or digital?

A: The user port of the VIC 20 is digital.

8. Q: How much usable memory is

available of the full 32K RAM in the VIC?

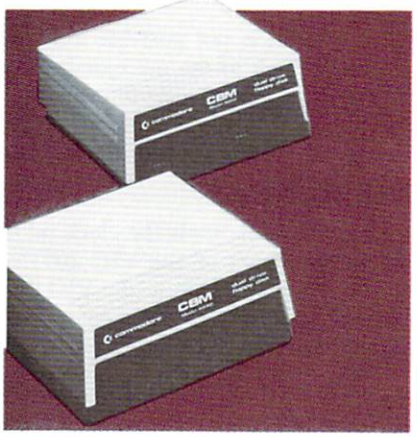
A: There is 27½ K usable memory available.

9. Q: How can screen memory be moved around on the VIC 20?

A: Screen memory can be moved around by altering the upper four bits of location 36869. This is the screen memory pointer location in the VIC chip. Also, you must tell the screen editor where the screen has been moved by POKEing the high byte of the starting screen address into location 648. Then warm start the system by either holding the run/stop and restore keys down or by typing the command: SYS 65234.

10. Q: Where does RAM start on the VIC 20?

A: RAM starts at 4097 decimal on the VIC.



Disk Drives

1. Q: Does the 8050 disk drive contain Tandon or Micropolis drives? Which drive is upgradeable to an 8250 unit?

A: The 8050 disk drive was produced with both Tandon and Micropolis drives. Tandon is the later drive. Neither drive is upgradeable to an 8250 unit.

2. Q: What is the availability of the 2040 and 4040 disk drive upgrade kits?

A: Upgrade kits may be ordered through Authorized Commodore Dealers. The part number for this kit is 321523-01.

3. Q: How can diskettes be backed up with a single disk drive?

A: Diskettes can be backed up with a single disk drive by LOADing the program into memory, and then SAVEing the program onto another diskette.

4. Q: What is the DOS for the VIC 1540 disk drive?

A: The DOS for the VIC 1540 disk drive is standard Commodore DOS version 2.6.

5. Q: Can programs stored on tape be converted to the disk drive?

A: Programs stored on tape can be converted to disk by LOADing the program into memory and then SAVEing the program onto a diskette.

6. Q: Can programs stored on diskette be copied to tape?

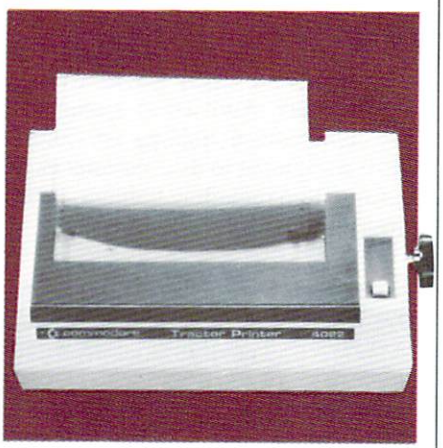
A: Programs stored on diskette can be copied to tape by LOADing the program from the diskette into memory and then SAVEing the program on tape.

7. Q: Can a Commodore Dual Disk Drive be connected to the VIC?

A: By using the IEEE-488 interface cartridge, any Commodore disk drive can be connected to the VIC 20.

8. Q: Can the VIC 1549 disk drive be connected to the PET?

A: No. The VIC 1540 disk drive connects to the VIC 20 serial port. The PET does not have a serial port like the one on the VIC. Instead, it has an IEEE port and a programmable parallel port.



Printers

1. Q: What is the difference between the 04 and 07 printer ROMs for the 2022?

A: The 07 printer ROM is the newer version of the 04 ROM. These ROMs are available from your local Authorized Commodore Dealer.

2. Q: What print wheels could be used for the 8300p printer?

A: Over 150 different print wheels are listed in the Diablo Print Wheel Book. This book is a Diablo publication (part number 90044-01).

3. Q: Are the VIC 1515 printer settings documented in any hardware manual?

A: The VIC 1515 Printer Manual has documentation on the settings. C=

In past issues, a full-page advertisement for MAG, Inc. contained an incorrect spelling of Lenox Industries. The correct spelling of this Texas-based company is Lennox Industries Inc.

THE COMPUTER CAN BE THE MOST DIFFICULT OF ALL EMPLOYEES TO TRAIN

We Train Micro Computers To Work For Business and Professional Firms

We believe we do it best as far as the folks who manage organizations are concerned. At least more business and professional firms come directly to us for management information systems than any other micro computer company in America. We produce quality products that are management oriented in a cost effective manner. We actually support what we produce and we conduct training. We even meet deadlines. Perhaps most important, we listen carefully and we do our best to meet your needs.

We have developed business and professional programs for firms with two or three employees to firms the size of Lennox Industries. Geography presents few problems for us. We have customers in all fifty states and Canada.

May We Train A Micro Computer For You!

MAG

Merging computer technology with the practical worlds of business and government

Corporate Central
Post Office Box 346
493 East Clayton Street
Athens, Georgia 30603
404/353-8090

Atlanta Division
4325 Memorial Drive
Suite H
Atlanta, Georgia 30032
404/294-8000
(Just inside I-285)

USER CLUBS
Sound Off!

We're continuing to compile a list of all Commodore Users clubs throughout the country. If you'd like to add your name to the rolls, please send your club's name, address, and other pertinent information to:

Commodore Users Clubs
c/o Editor
Commodore Magazine
681 Moore Road
King of Prussia, PA 19406

And remember, once our list is comprehensive enough, we will begin forwarding valuable information to clubs on a regular basis, including hardware and software updates, technical bulletins, new product announcements, and troubleshooting tips.

ALABAMA

Huntsville PET Users Club
9002 Berclair Road
Huntsville, AL 35802
Contact: Hal Carey
Meetings: every 2nd Thursday

ARIZONA

VIC Users Group
1206 N. Fraser Drive
Mesa, AZ 85203
Contact: Paul V. Muffuletto

ARKANSAS

Commodore/PET Users Club
Conway Middle School
Davis Street
Conway, AR 72032
Contact: Geneva Bowlin

CALIFORNIA

Lawrence Hall of Science
UC Berkeley
Computer Project, Room 254
Berkeley, CA 94720
(415) 642-3598

PALS (PETS Around Livermore Society)
886 South K
Livermore, CA 94550
Contact: John Rambo

SCPUG Southern California PET Users Group
c/o Data Equipment Supply Corp.
8315 Firestone Blvd.
Downey, CA 90241
(213) 923-9361
Meetings: First Tuesday of each month

California VIC Users Group "VIC-VILLE"
c/o Data Equipment Supply Corp.
8315 Firestone Blvd.
Downey, CA 90241
(213) 923-9361
Meetings: Second Tues. of each month

Commodore Users Club
1041 Foxenwoods Drive
Santa Maria, CA 93455
(805) 937-4106
Contact: Greg Johnson

Valley Computer Club
2006 Magnolia Blvd.
Burbank, CA
(213) 849-4094
1st Wed. 6 p.m.

Valley Computer Club
1913 Booth Road
Ceres, CA 95307

PUG of Silicon Valley
22355 Rancho Ventura Road
Cupertino, CA 95014

BAMBUG
1450 53rd Street
Emeryville, CA
(415) 523-7396

North Orange County Computer Club
3030 Topaz, Apt. A
Fullerton, CA 92361
Dave Smith

Lincoln Computer Club
750 E. Yosemite
Manteca, CA 95336
John Fung, Advisor

PET on the Air
525 Crestlake Drive
San Francisco, CA 94132
Max J. Babin, secretary

PALS (PETs Around Livermore Society)
886 South K
Livermore, CA 94550
(415) 449-1084
Every third Wednesday 7:30 p.m.
Contact: J. Johnson

SPHINX
314 10th Avenue
Oakland, CA
(415) 451-6364
Every 2nd & 4th Thurs.

San Diego PUG
c/o D. Costarakis
3562 Union Street
(714) 235-7626
7 a.m.-4 p.m.

Walnut Creek PET Users Club
1815 Ygnacio Valley Road
Walnut Creek, CA 94596

Jurupa Wizards
4526 Kingsbury Pl.
Riverside, CA 92503
Contact: Walter J. Scott

PET/CBM/VIC User Club
c/o General Computer Store
22323 Sherman Way #9
Canoga Park, CA 91303
Contact: Tom Lynch

Vincent Yanniello's VIC 20 Software Exchange Club
2130 Colby Avenue
West Los Angeles, CA 90025
(213) 479-3000

COLORADO

VICKIMPET Users' Group
4 Waring Lane,
Greenwood Village
Littleton, CO 80121
Contact: Louis Roehrs

CONNECTICUT

John F. Garbarino
Skiff Lane Masons Island
Mystic, CT 06355
(203) 536-9789

Commodore User Club
Wethersfield High School
411 Wolcott Hill Road
Wethersfield, CT 06109
Contact: Daniel G. Spaneas

VIC Users Club
c/o Edward Barszczewski
22 Tunxis Road
West Hartford, CT 06107

New London County Commodore Club
Doolittle Road
Preston, CT 06360
Contact: Dr. Walter Doolittle

FLORIDA

Jacksonville Area PET Society
401 Monument Road, #177
Jacksonville, FL 32211

Richard Prestien
6278 SW 14th Street
Miami, FL 33144

South Florida PET Users Group
Dave Young
7170 S.W. 11th
West Hollywood, FL 33023
(305) 987-6982

VIC Users Club
c/o Ray Thigpen
4071 Edgewater Drive
Orlando, FL 32804

ILLINOIS

Shelly Wernikoff
2731 N. Milwaukee Avenue
Chicago, IL 60647

VIC 20/64 Users Support Group
c/o David R. Tarvin
114 S. Clark Street
Pana, IL 62557
(217) 562-4568

Central Illinois PET User Group
635 Maple
Mt. Zion, IL 62549
(217) 864-5320
Contact: Jim Oldfield

ASM/TED User Group
200 S. Century
Rantoul, IL 61866
(217) 893-4577
Contact: Brant Anderson

PET VIC Club (PVC)
40 S. Lincoln
Mundelein, IL 60060
Contact: Paul Schmidt, president

Rockford Area PET User's Group
1608 Benton Street
Rockford, IL 61107

Commodore Users Club
1707 East Main St.
Olney, IL 62450
Contact: David E. Lawless

INDIANA

PET Users
Jerry Brinson
PO Box 36014
Indianapolis, IN 46236
(317) 898-3604

GHS Computer Club
c/o Grangeville High School
910 S. D St.
Grangeville, ID 83530
Contact: Don Kissinger

Cardinal Sales
6225 Coffman Road
Indianapolis, IN 46268
(317) 298-9650
Contact: Carol Wheeler

CHUG (Commodore Hardware Users Group)
12104 Meadow Lane
Oaklandon, IN 46236
Contact: Ted Powell

IOWA

PET Users Group
c/o Don Vorhies
1321 42 St. SE
Cedar Rapids, IA 52403

KANSAS

Wichita Area PET Users Group
2231 Bullinger
Wichita, Kansas 67204
(316) 838-0518
Contact: Mel Zandler

Kansas Commodore Computer Club
101 S. Burch
Olathe, KS 66061
Contact: Paul B. Howard

VIC 20 Users Club
739 Litchfield
Wichita, KS 67203
(316) 262-4861
Contact: Walter Lounsbery

MARYLAND

Assoc. of Personal Computer Users
5014 Rodman Road
Bethesda, MD 20016

Blue TUSK
700 East Joppa Road
Baltimore, MD 21204
Contact: Jim Hauff

House of Commodore
8835 Satyr Hill Road
Baltimore, MD 21234
Contact: Ernest J. Fischer

MASSACHUSETTS

Eastern Massachusetts VIC Users Group
c/o Frank Ordway
7 Flagg Road
Marlboro, MA 02173

VIC Users Group
c/o Ilene Hoffman-Sholar
193 Garden St.
Needham, MA 02192

Commodore Users Club
Stoughton High School
Stoughton, MA 02072
Contact: Mike Lennon

Berkshire PET Lovers
CBM Users Group
Taconic High
Pittsfield, MA 01201

MICHIGAN

David Liem
14361 Warwick Street
Detroit, MI 48223

VIC Users Club
University of Michigan
School of Public Health
Ann Arbor, MI 48109
Contact: John Gannon

Toledo PETS
734 Donna Drive
Temperance, MI 48182
Contact: Gerald Carter, president

Commodore User Club
32303 Columbus Drive
Warren, MI 48093
Contact: Robert Steinbrecher

Commodore Users Group
c/o Family Computer
3947 W. 12 Mile Rd.
Berkley, MI 48072

MINNESOTA

MUPET (Minnesota Users of PET)
P.O. Box 179
Annandale, MN 55302
c/o Jon T. Minerich

Twin Cities Commodore Computer Club
6623 Ives Lane
Maple Grove, MN 55369
(612) 424-2425
Contact: Rollie Schmidt

MISSOURI

Clearwater Club
Clearwater School
Star Route
Piedmont, MO 63957
Contact: Carolyn Polk

KCPUG
5214 Blue Ridge Boulevard
Kansas City, MO 64133
Contact: Rick West
(816) 356-2382

PET SET Club of St. Louis
1501 Carman Road
Manchester, MO 63011
(314) 527-3236
Contact: Ed or Til Keil

MONTANA

Powder River Computer Club
Powder River County High School
Broadus, MT 59317
Contact: Jim Sampson

Commodore User Club
1109 West Broadway
Butte, MT 59701
Contact: Mike McCarthy

NEVADA

Las Vegas PET Users
4884 Iron Avenue
Las Vegas, NV 89110

NEW JERSEY

Amateur Computer Group of New Jersey
John Loofbourrow
UCTI, 1776 Raritan Road
Scotch Plains, NJ 07076
(201) 233-7068

Amateur Computer Group
18 Alpine Drive
Wayne, NJ 07470

Somerset Users Club
49 Marcy Street
Somerset, NJ 08873
Contact: Robert Holzer

NEW HAMPSHIRE

Northern New England Computer Society
PO Box 69
Berlin, NH 03570

NEW YORK

Capital District PET Users
Ben Green
Albany Area, NY
(518) 370-1820

Long Island PET Society
Ralph Bressler
Harborfields HS
Taylor Avenue
Greenlawn, NY 11740

PET User Club of Westchester
Box 1280
White Plains, NY 10602
Contact: Ben Meyer

LIVE (Long Island VIC Enthusiasts)
17 Picadilly Road
Great Neck, NY 11023
Contact: Arnold Friedman

Commodore Masters
25 Croton Ave.
Staten Island, NY 10301
Contact: Stephen Farkouh

VIC Users Club
76 Radford St.
Staten Island, NY 10314
Contact: Michael Frantz

VIC Users Club
c/o Christopher Kwasnicki
44 Harvey Ave.
Staten Island, NY 10314

PUG of Rockland County
c/o Ross Garber
14 Hillside Court
Suffern, NY 10901
(914) 354-7439

NORTH CAROLINA

Amateur Radio PET User's Group
PO Box 30694
Raleigh, NC 27622
Contact: Hank Roth

Commodore Users Club
4241 Castleton Road
Charlotte, NC 28211
Contact: Ed Harris

VIC Users Club
c/o David C. Fortenberry
Route 3, Box 351
Lincolnton, NC 28092

Microcomputer Users Club
Box 17142 Bethabara Sta.
Winston-Salem, NC 27116
Joel D. Brown

OHIO

Dayton Area PET User Group
933 Livingston Drive
Xenia, OH 45385
B. Worby, president
(513) 848-2065
J. Watson, secretary
(513) 372-2052

Central Ohio PET Users Group
107 S. Westmoor Avenue
Columbus, Ohio 43204
(614) 274-6451
Contact: Philip H. Lynch

OREGON

NW PET Users Group
John F. Jones
2134 N.E. 45th Avenue
Portland, OR 97213

PENNSYLVANIA

PET User Group
Gene Beals
PO Box 371
Montgomeryville, PA 18936

Penn Conference Computer Club
c/o Penn Conference of SDA
720 Museum Road
Reading, PA 19611
Contact: Dan R. Knepp

PACS PET Users Group
20th & Olney Streets
Philadelphia, PA

Glen Schwartz
807 Avon
Philadelphia, PA 19116

Gene Planchak
4820 Anne Lane
Sharpsville, PA 15150
(412) 962-9682

PPG (Pittsburgh PET Group)
c/o Joel A. Casar, DMD
2015 Garrick Drive
Pittsburgh, PA 15235
(412) 371-2882

SOUTH DAKOTA

PET User Group
515 South Duff
Mitchell, SD 57301
(605) 996-8277
Contact: Jim Dallas

TENNESSEE

River City Computer Hobbyists
Memphis, TN
1st Mon. at Main Library

Nashville VIC Users Group
American Computer Store
1004 8th Ave., S.
Nashville, TN 37203
(615) 242-8592
Jane Maggard
1st Thurs. of month, 6 pm

TEXAS

SCOPE
1020 Summit Circle
Carrolton, TX 75006

PET Users
2001 Bryan Tower
Suite 3800
Dallas, TX 75201

Larry Williams
PO Box 652
San Antonio, TX 78293

PET User Group
John Bowen
Texas A & M
Microcomputer Club
Texas A & M, TX

CHUG (Commodore Houston Users Group)
8738 Wildforest
Houston, TX 77088
(713) 999-3650
Contact: John Walker

UTAH

Utah PUG
Jack Fleck
2236 Washington Blvd.
Ogden, UT 84401

The Commodore User's Club
742 Taylor Avenue
Ogden, Utah 84404
Contact: Todd Woods Kap, president; David J. Shreeve, vice president

The VIClic
799 Ponderosa Drive
Sandy, UT 84070
Contact: Steve Graham

VIRGINIA

Northern VA PET Users
Bob Karpen
2045 Eakins Court
Reston, VA 22091
(803) 860-9116

VIC Users Group
Rt. 2, Box 180
Lynchburg, VA 24501
Contact: Dick Rossignol

VIC Users Group
c/o Donnie L. Thompson
1502 Harvard Rd.
Richmond, VA 23226

Dale City Commodore User Group
14752 Danville Rd.
Dale City, VA 22193
(703) 680-2270
James Hogler

WASHINGTON

NW PET Users Group
2565 Dexter N. 3203
Seattle, WA 98109
Contact: Richard Ball

PET Users Group
c/o Kenneth Tong
1800 Taylor Ave. N102
Seattle, WA 98102

WISCONSIN

Sewpus
c/o Theodore J. Polozynski
PO Box 21851
Milwaukee, WI 53221

Waukesha Area Commodore User Group (WACUG)
256½ W. Broadway
Waukesha, WI 53186
Contact: Walter Sadler
(414) 547-9391

CANADA

Toronto PET Users Group
381 Lawrence Ave. West
Toronto, Ontario, Canada
M5M 1B9
(416) 782-9252
Contact: Chris Bennett

PET Users Club
c/o Daniel Cayer
R.R. 6
Simcoe, Ontario
Canada N3Y 4K5

Vancouver PET Users Group
Box 91164
West Vancouver, British Columbia
Canada V7V 3N6

CCCC (Canadian Commodore Computer Club)
c/o Strictly Commodore
47 Coachwood Place
Calgary, Alberta, Canada
T3H 1E1
Contact: Roger Olanson

KOREA

Commodore Users Club
K.P.O. Box 1437
Seoul, Korea
Contact: S. K. Cha

MEXICO

Asociacion De Usuarios Commodore
c/o Alejandro Lopez Arechiga
Holbein 174-6° Piso
Mexico 18, D.F.

NORWAY

VIC Club of Norway
Nedre Bankegt 10,
1750 Halden
Norway

EDUCATION

Software Selection

Dr. Bruce Downing
Director, Industry-Marketing
Commodore

This is our third in a series of articles concerning planning for educational uses of computers. In the first article, we reviewed the elements of a good plan and the second addressed the need to set objectives and the types of applications to consider. In this issue, we will look at one of the most critical, and often misunderstood, areas—software selection.

A certain indicator that the objectives for computer use have not been carefully considered is the statement "we chose this computer because it seems to have more software for it." In reality, less than ten percent of all available educational software is of sufficient quality to be considered (as judged by "experts" in a survey). Therefore, claims by computer vendors that there are thousands of programs available have little value. What is important is whether there are programs of sufficient quality that fit with predetermined objectives and are easily implemented in a school curriculum.

The primary sources of educational software are:

Book Publishers
Independent Software Vendors
Computer Manufacturers
Public Domain Collections
Local Development

Book publishers have typically entered the software arena by producing programs intended as supplements to existing books. Few of these programs can be used without a book but the quality of programs tends to be fairly high. The programs are usually relatively expensive because the publishers development costs are high and the number of potential sales in schools is relatively low. However, new low-cost computers available for schools and home education promise to increase the available market and reduce prices.

The quality of software from independent software vendors varies greatly and there is not always an opportunity to preview programs before purchase. The quality of documentation explaining how to use the programs also varies greatly.

Computer manufacturers seldom produce educational applications programs and concentrate on general capabilities such as languages and development systems. Most of the manufacturers give assistance of various types to software producers to encourage writing programs. However, curriculum-oriented materials are seldom written by the hardware companies.

There are thousands of programs that have been put in the public domain and are, therefore, free for use. Most of these programs are limited in scope and were developed for local use. Some of them, however, have been developed with state or federal funds and some are of sufficient quality to be seriously considered. Usually, the programs available in the public domain are poorly documented and their objectives or intended grade levels are not specified. However, teachers willing to investigate and sift through many public domain programs can usually find some that are useful.

Given that publishers software is expensive and programs for particular subjects are not available, some set out to develop their own. In fact, some of the best software from independent developers comes from teachers who decided to write their own programs. Generally, local development is time consuming and expensive. The written materials and other support materials are difficult to develop, so local development is not always feasible.

Generally speaking, the programs from publishers and software vendors are the highest quality and best supported. However, the availability of good software in particular subjects may be limited. Educators preparing to investigate the availability of programs and evaluate quality need to recognize that it can be a frustrating and time consuming effort. However, a few tactics for selections can be helpful.

First of all, software which has been evaluated by other teachers and found useful is the most likely candidate for adoption. Some software producers have programs that are excellent. But, many times it is possible to find a school with similar goals that has already evaluated many programs and their experience can be valuable.

Programs that are well documented and include support materials such as teacher and student guides are usually better designed programs. However, caution must be exercised to guard against fancy looking support material that is shy of content and accompanies short, useless programs.

In all cases, one should attempt to obtain review copies of the software or a money-back guarantee. Few vendors offer this option because they fear piracy; so, again, one alternative is to obtain references of other teachers using the programs.

Software that is part of an entire curriculum and for which learning objective and grade levels are indicated is generally better software. Also, software based on curriculum recommendations of professional associations should have high priority for evaluation.

There is a tremendous amount of effort being expended to create good quality software. However, it is a slow and costly effort so that an abundance of good software will not appear soon from any particular source. Commodore has initiated several activities to identify, catalog and publicize good software. In addition, we are working with many outstanding publishers to encourage good software for our computers. Many of these efforts will produce new software to add to the substantial base of existing software.

One final point—the best source of evaluative information is *still* a referral from another teacher. C=

School Offers Instruction in Computer Literacy

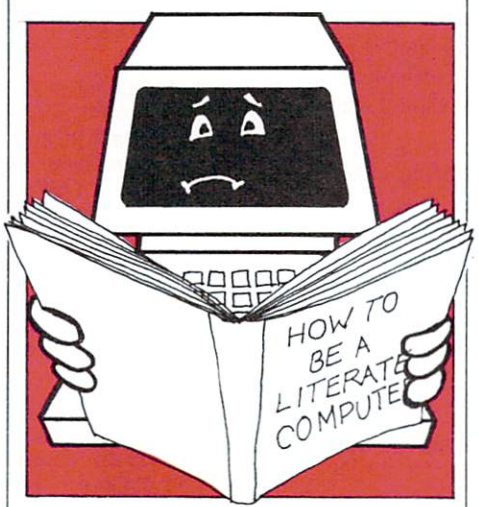
Are you a computer illiterate? Baffled and sometimes even afraid of the mystery surrounding the magical keyboard and the deep dark screen? Well, fear no more. There will soon be a course for you and your children—specially designed to demystify microcomputers.

This structured, "hands-on" learning program will be ofered by The Children's Computer School in Columbus, Ohio beginning in May. The school will offer daytime and evening classes which will focus on teaching basic computer operation and programming. The school will expand to many other cities across the country in

"We foresee personal computers being an everyday mode of operation in every business and every home," explains Linda Thoirs of CompuServe, which is lending technical support to the school. "The idea is to teach computer literacy and computer competency so that when we evolve to the stage when everyone has his own computer—he'll know how to use it.

"It's a natural fit for CompuServe," says Thoirs, "especially with our personal computing division. It just makes sense to teach people how to use computers. This is just the start of many more things of this nature that will help overcome the basic fear of computers and teach people on an individual basis how to make computers work for them."

While researching the concept of a computer school for kids, CompuServe officials heard about Dr. Eugene Galanter of Columbia University in New York. He had developed what they were looking for—a working model of a computer school geared towards teaching children, but with an enrollment that included 50 percent adults.



"We were very impressed by the simplicity of his methods of teaching at the Children's Computer School," says Thoirs. "The eight-week course is presented in everyday language without getting into high-powered computer wordage. Children can start as soon as they know how to read, usually by age seven or eight."

The key to Dr. Galanter's course is "hands-on" instruction. Each child has his own computer. With ten students, an instructor and an assistant trainer in each class—every student receives a great deal of personal attention. Encouraged to progress at their own pace, students are placed in classes, not according to age but rather by their ability.

Thoirs suggests that children could take the class as they might seek out instruction in a foreign language. "We have seen that children take to this like ducks to water. They have no fear," says Thoirs. "They don't always understand the theory behind it, but they're open to learn. It's a natural thing, like learning languages, when they're small. They comprehend so quickly and it gets to be like second nature to them—just like using the telephone."

After learning BASIC programming in two four-week sessions (Two hours each week), students can move to a variety of carefully organized computer literacy classes, including assembly and machine language, word processing and graphics.

After completing the first course, many of Dr. Galanter's students can write programs that help them with their regular school work. One student even developed a program to teach and test for Latin declensions.

In addition, notes Thoirs, learning to work with computers "teaches discipline, logic, manual dexterity and improves eye/hand coordination, reading and math skills.

"In the future," concludes Thoirs, "learning computers will be just like learning how to read and write and do math. It will be part of your everyday life. This is just the beginning."

If you'd like more information about The Children's Computer School, call 614-451-3444 or write The Children's Computer School, 1822 Fishinger Road, Columbus, Ohio 43221. C=

by Peggy Garvey

Award Winner Uses PETS

Montana Teacher Named First Technology-Using Educator of The Year

Sue Dolezal, a high school math teacher who created an extensive computers-in-education program in a conservative, low-income district, and who went on to help introduce computers to educators throughout the entire state of Montana, was awarded the top honor in Electronic Learning Magazine's first annual Technology-Using Educator of the Year Awards Program. Dolezal, 29, is a teacher in Stevensville, Montana—population 1000. She was presented the award for "outstanding leadership in the advancement of education through technology."

In announcing the award, Jack Roberts, editor of Electronic Learning, said, "I'm delighted to present this award to an educator who has demonstrated such remarkable initiative and leadership in putting technology to work in the classroom. Sue Dolezal has been an inspiration to her colleagues, not only in her own community but throughout her state as well.

"We initiated the Awards Program this year," Roberts continued, "to bring recognition to the contributions that hundreds upon hundreds of educators are making to education through their innovative use of technology for instruction." The Awards Program, according to Roberts, drew nominations of teachers, principals, and superintendents from 27 states.

Dolezal, a native of Montana, is the driving force behind the Stevensville School District's three-year-old computers-in-education program. In a community described by her nominators as "ultra conservative"—within a school district that spends the "lowest amount of dollars for education per student in the entire state of Montana"—this one teacher, working largely on her own, has created an educational computing program that compares favorably with those in districts much larger and wealthier than hers." In addition, Dolezal has acted as a computer consultant for "literally the entire state," and has been recognized as such by the Montana State Department of Education.

"It was Sue's special quality of leadership that attracted most of the judges," said Stephen Gass, associate editor of EL and coordinator of the Awards Program. "She really had to overcome tremendous odds to accomplish all that she has done. But in a larger sense, I think Sue was selected because she seemed to represent so many of the individual qualities the judges admired in other nominees."

Dolezal's Accomplishments

In beginning her computers-in-education program, Dolezal first managed to acquire a small number of PETs through an ESEA grant. Dolezal said she chose Commodore be- →

cause "what a school needs is as many computers as possible for hands-on experience, and the low price of the PETs allowed us to purchase many more computers."

Explaining how her program grew, Dolezal says: "My strategy was to bring students into my classroom, introduce them to the computer, and let them tell their other teachers they've got to have one." Soon, Dolezal was working with the teachers, conducting training workshops and selling them on the idea that computers have an important place in the classroom—"something I really believe in," Dolezal says. Before long, she was serving as a computer consultant throughout all of Montana.

In total, the Stevensville School District uses 47 Commodore computers for grades K through 12. The PETs are used for programming, Computer Assisted Instruction (CAI), word processing, and reading.

The news that she had been awarded the top honor in EL's Technology-Using Educator of the Year Awards Program came as a "total surprise." After being named the winner, she mused: "You hear and read about so many people who are doing such terrific work in the field that to be selected even as a finalist would be a great honor. To be named Educator of the Year "was a real shock."

As EL's award winner, Dolezal was featured in a special cover story about her activities in the September issue of the magazine. "I'm looking forward to being able to share my experiences with other educators throughout the country," Dolezal said. "One of the great things that the new technology—particularly the microcomputer—has done for educators at all levels is to bring them together to share new ideas and experiences. College people are talking to secondary people, who are talking to elementary people, and they're all talking about the same thing—technology and what it can mean to education. That's exciting."

Denesen's Photography

Skyles Electric Works

WORD-WRITER

Add this one small device and greatly increase the power of your 8032, 8096 or SuperPet.

This one small device can double, triple or quadruple the usefullness of your Commodore Business Machine.

Open your CBM computer and place this modern wonder chip in the rear empty socket. Close the lid and trun on your CBM with **WORD-WRITER a built-in word processing system.** Type "SYS36864", "RVS", key and "L" for letter. That is all there is to it — no diskettes or tapes to hassle with. Start writing!! You never knew a word processor could be so easy to use. Make a mistake? **No problem,** cursor to the front of the mispelling type "RVS" key "G key" and gobble up the offending word. Now strike "RVS," "I" (for insert) and type in the correct word or phrase. You only want to change a letter, or a line? **No problem,** a couple simple key strokes will do it.

WORD-WRITER

Want to move or copy a line or paragraph? Want to insert a something? **No problem,** a few simple practical keystrokes and it is done. Want to change a word because you have thought of a better word? Want to change a name everywhere it appears? Want to change some occurrences of a word? **No problem,** two (2) count them, two keystrokes set you up to do this.

Do most of your letters and reports have standard paragraphs in them? **No problem,** write them once save them on disk or cassette and use them thousands of times with only a few keystrokes of typing for each paragraph.

Have a lengthy report, memo or letter to write? **No problem, Word-Writer** holds up to 15 — count them — 15 pages of text at one time.

Want to search, scroll, skim, skip, flip or jump through the text? **No problem,** a couple of keystrokes and you are on your way.

What about typing out or printing the text? **No problem,** a few keystrokes and you can print out a rough draft or final copy on whatever printer you have attached to your PET/CBM

This modern wonder chip:

WORD-WRITER

Available Immediately from your local dealer at a modest cost of only **$85.00** complete with an easy reading manual.

Try Word-Writer for 10 days, if you aren't completely amazed that such a small item could make such a major improvement to your CBM8032, CBM8096 or SuperPet, please return it for a full refund.

INSTALLATION: Installs into socket UD11 ($A000)

PRICE: For CBM 8032, 8096 or SuperPET SP9000 WORD-WRITER $85.00
Please specify your PET/CBM model when ordering

AVAILABILITY: Immediately from your LOCAL DEALER

or

VISA, MASTERCHARGE ORDERS CALL (800) 227-998 (except California residents)
CALIFORNIA ORDERS PLEASE CALL (415) 965-1735

Skyles Electric Works | 231E South Whisman Road
Mountain View, CA 94041
(415) 965-1735

Running a Completely Automated Experiment by Microcomputer: An Example Program in BASIC

by
Craig W. Johnson

Recent advances in microcomputer technology place sophisticated control processes and data acquisition capabilities within the reach of individual investigators in small institutions. Previously, such technology was accessible only in large-scale minicomputer-based laboratories. The technological breakthrough represented in the development of the inexpensive microcomputer provides the rank and file of experimenters in the behavioral sciences an instrumentality of such potentialities that it, quite probably, will very shortly revolutionize the conduct of much cognitive and behavioral research.

This is likely to occur for a number of reasons. Among the most important are the vastly enhanced capacities for precise replication and extension, for measurement, for experimental control, and randomization, and the relative economy which microcomputer administered experimental treatments provide the researcher.

A Case in Point

The experiment. The purpose of the experiment was to compare the effectiveness of two keyword techniques of vocabulary acquisition with a "use your own method" control technique on both an immediate and a follow-up occasion.

The three treatments were totally microcomputer administered, monitored, scored, and randomly assigned in a double blind procedure without presence of an experimenter. The acquisition phase and the immediate test took place during the first week. A delayed test was administered one week later. All treatments were conducted individually.

Each participant was seated at the microcomputer terminal. An assistant then outlined very generally that the participant would be learning some new words, turned on the machines, placed the program disk into the disk drive, and loaded the program.

Participants began by pressing the "return" key of the computer. The assistant remained briefly (less than a minute) to make sure the participants had no problems in getting started and that the machines were functioning correctly. Participants were cued by the computer to proceed, with the program presenting a brief set of instructions outlining the procedure. This was followed by a prior knowledge rating task, in which the order of the words, which were subsequently to be learned, was independently re-randomized for each different participant. After the rating task, participants were randomly assigned by the computer to one of the three treatments. The instructions appropriate to treatment membership were presented, followed by four example items. Participants proceeded through all these tasks at their own pace.

At the conclusion of the examples, participants were cued by the computer to begin the focal vocabulary acquisition task. Thirty items were randomly presented in a different order for each participant, one at a time, paced by the computer. Each item was presented for 15 seconds. Each target word was accompanied by its definition, a short sentence restatement of the instructional set appropriate to the treatment, and (for the keyword treatments only) the appropriate keyword.

When participants finished the 30 items, they began the recall test, again proceeding at their own pace. Each participant responded to the 30 multiple-choice items, presented by the computer in a different random sequence, one by one. Participants returned one week after the first session and responded to the microcomputer administered delayed test.

The program. The following small and original 228 line program conducts such an experiment. The program, written in BASIC to be run on Commodore CBM or PET machines equipped with at least one disk drive, gives a more concrete idea of what micrcomputer administered experiments can do in implementing educational, psychological, and behavioral research, and illustrates some of the interactive power of the newer extended BASIC languages which have become an industry standard. It is identical to the program that actually administered the experiment, except for deletion of a very brief machine-language subroutine disabling the stop key to prevent accidental program interruption and, in the interest of space, of the subroutine presenting the four example items.

The program assesses a prior knowledge covariate; and as dependent variables, task completion times, to achieve greater control of time related effects and retention test scores. Completion times were ordered first in a MANOVA using the step-down F statistic (see Bock, 1975; Roy and Bargman, 1958).

Scores were stored on the same 5¼-inch floppy disk containing the program. Scores could be analyzed, after concatenation of the individual data files, without transcription or keypunching by SPSS, SAS, BMDP, MULTIVARIANCE, etc., through a telephone link communications modem with the main university computer. Of course, had the appropriate software been available for the microcomputer, the MANOVA with step-down F could have been performed using only the microcromputer itself.

The program, among other things: (a) collects subject identification information; (b) obtains ratings on the prior knowledge covariate; (c) randomly assigns subjects to three treatments after obtaining prior knowledge ratings;

(d) presents the instructional set for each of the three different treatment groups; (e) presents 15 abstract and 15 concrete to-be-learned items to the three different treatment groups for identically specified time intervals (15 seconds per item); (f) presents instructions for a test of retention; (g) presents the test of retention; (h) randomizes the order of presentation of all items in all phases of the procedure, and does it differently for each subject; (i) randomizes the order of alternatives within items of the test of retention; (j) monitors, scores, and stores, on disk, prior knowledge covariate and retention test performances, including rating task, acquisition task, and total treatment completion times; rating responses, and item, subscale, and total test scores for each subject (72 variables for each subject on the first occasion); and (k) administers, only, the individually randomized follow-up test on a second occasion, storing the subject identification information, the completion time and item, subscale, and total scores.

For those having some skill in BASIC, examination of the program listing will provide an understanding of how these tasks are accomplished. In particular, the "REM" statements describe functions of various subroutines within the program. Minor changes in the program allow it, generally, to be run on microcomputers from other manufacturers, if the machine's hardware includes a clock. The heart-shaped symbol, which appears occasionally in the program, is a command to clear the screen. As extensive, powerful, and flexible as the program is, it occupies only slightly more than 8K (8000) bytes in its present form. It may be easily compacted to approximately 7100 bytes. The fact that such varied, flexible, and useful control processes may be programmed in BASIC, a relatively simple to learn (yet powerful) computer language available on nearly all microcomputers, with a syntax much like that of English, should be of some interest to researchers, along with the fact that no time-sharing with a large computer, which is becoming increasingly expensive, was necessary.

Craig Johnson, who is Assistant Professor, Department of Administration and Foundations, Kansas State University, is currently developing software in Commodore BASIC to teach statistical procedures to graduate students in education and psychology.

"I have had my Commodore system for about two years," said Johnson, "and it has helped me substantially in my career." Johnson, who performs a great deal of statistical programming, expressed his pleasure with being able to run programs in SPSS, SAS, and BMDP on the university mainframe using his Commodore 8010 modem. "Of course," he added, "having a lot of statistical software of my own, I can perform many analyses using only the CBM. I live 30 miles from the university campus and this has saved me countless hours of programming and travel time."

Johnson also uses his Commodore in his grading, and for data management of consultation activities and bibliographical resources. "It turns previously dull and frustating tasks into interesting and enjoyable activities," said Johnson.

"I bought Commodore for a variety of reasons: superior word processing, disk drive capacity, easy full-screen editing capabilities, sharp integrated green-phosphor screen, real-time clock for controlling experiments, graphics printer and price. I readily agree with the latest Commodore ads that say 'Does more. Costs less.'

"I've had no problem over a two-year period, other than replacing a $15 chip after a severe electrical storm. The local Commodore representative, Bob Johnson of Midwest Computers, has provided firm, honest, and technically useful support and assistance. What a fantastically powerful, flexible, and versatile machine!

"Riding the back of this technology has already led me to the methodological forefront of educational researchers," added Johnson.

References

Atkinson, R. C. Mnemotechnics in Second Language Learning. *American Psychologist*, August 1975, 821-828.

Bock, R. D. *Multivariate Statistical Methods in Behavioral Research*. New York: McGraw-Hill, 1975.

Faw, H. W., and Waller, T. G. Mathemagenic Behaviors and Efficiency in Learning from Prose. *Review of Educational Research*, 1976, *46*, 691-720.

Flowers, J. H., and Murphy, M. H. An Economical Microcomputer Design for the Generation and Display of Timed Visual Stimulii: The TDL VDB. *Behavior Research Methods and Instrumentation*, 1979, *11*(12),115-118.

Johnson, C. W., and Adams, M. *Imagery, Mnemonics, and Level of Processing in Comprehending Concepts from Definitions*. Presented at the Annual Meeting of the American Educational Research Association, April 1980.

Johnson, C. W., and Noble, A. C. *Watch Those Keywords: Microcomputer Assessment of Keyword Effectiveness* (manuscript in preparation).

Lien, D. A. *The BASIC Handbook*. San Diego: Compsoft, 1978.

Roy, J., and Bargman, R. E. Tests of Multiple Independence and the Associated Confidence Bounds. *Annals of Mathematical Statistics*, 1958, *29*, 491-503.

In our April/May issue, we featured a story in our education section entitled "Microcomputing in the Troy, Michigan School District." In the article, it was incorrectly reported that a newsletter was available from the school district's micro committee. We regret any inconvenience this may have caused the committee or our readers.

PROGRAM LISTING

```
1 REM SET TIME INTERVAL FOR LEARNING TASK AND COLLECT SUBJECT ID INFORMATION
2 D=900:TI$="000000":PRINT"□YOUR PARTICIPATION IN THIS"
3 PRINT"RESEARCH IS APPRECIATED.  IT IS"
4 PRINT"HOPED THAT YOUR CONTRIBUTION WILL"
5 PRINT"BE REFLECTED IN THE DEVELOPMENT"
6 PRINT"OF NEW AND MORE EFFECTIVE TOOLS FOR     VOCABULARY ACQUISITION.":PRINT
7 PRINT"TYPE THE LAST 4 DIGITS OF YOUR SOCIAL   SECURITY NUMBER AND THEN ";
8 PRINT"PRESS THE      'RETURN' KEY ON THE RIGHT SIDE OF THE   KEYBOARD."
9 INPUTA$
10 PRINT:PRINT"IS THIS YOUR FIRST OR SECOND SESSION IN THIS EXPERIMENT?"
11 GETB$:IFB$=""THEN11
12 IFVAL(B$)<>1ANDVAL(B$)<>2THENPRINT"PLEASE TYPE EITHER '1' OR '2'":GOTO10
13 PRINT" ";B$:GOSUB201
14 TSUBD=TI:TI$="000000"
15 REM SET THE MAXIMUM SIZES FOR THE VARIABLES
16 DIMW$(31),WU(31),K1$(31),K2$(31),D$(31),V$(73),PK(31),PK$(31),P(4),R$(30)
17 DIMCS(30):FORI=1TO30:CS(I)=0:NEXT:GOSUB24
18 REM SUBJECT RANDOM ASSIGNMENT ROUTINE
19 G$="0":S=RND(-RND(0)):IFS<.333THENG$="E1"
20 IFS>.333ANDS<.666THENG$="E2"
21 IFS>.666THENG$="C"
22 IFG$<>"E1"ANDG$<>"E2"ANDG$<>"C"THEN19
23 TND=TI:TI$="000000":GOTO148
24 REM READ WORDS, KEYWORDS AND DEFINITIONS
25 FORI=1TO30:READW$(I):NEXT
26 FORI=1TO30:READK1$(I):NEXT:FORI=1TO30:READK2$(I):NEXT
27 FORI=1TO30:READD$(I):NEXT
28 TDI=TI:TI$="000000":IFVAL(B$)=2GOTO64
29 GOSUB169
30 TRATING=TI:TI$="000000":RETURN
31 TEX=TI:TI$="000000"
32 FORJ=1TO30:WU(J)=0:NEXT
33 FORK=1TO30
34 REM ROUTINE FOR RANDOMIZING ORDER & TIMING DURING LEARNING TASK
35 GOSUB37
36 GOTO42
37 J=INT(30*RND(1))+1
38 IFWU(J)<>0GOTO37
39 IFK=1THEN41
40 IFTI<DGOTO40
41 WU(J)=1:RETURN
42 REM PRINTS LEARNING TASK FOR UNFAMILIAR WORDS
43 PRINT"□"W$(J):IFG$="E1"THENPRINT:PRINT"(KEYWORD IS '";K1$(J)"')":GOTO45
44 PRINT"□"W$(J):IFG$="E2"THENPRINT:PRINT"(KEYWORD IS '";K2$(J)"')"
45 PRINT:PRINT"MEANS - ";D$(J):IFG$<>"C"THENGOSUB210
46 IFG$="C"THENGOSUB205
47 REM SET TIME AND TIME PRESENTATIONS
48 TLT=TI+TLT:TI$="000000":NEXTK
49 IFTI<DGOTO49
50 TLT=TI+TLT:TI$="000000"
51 REM LIST OF UNFAMILIAR WORDS
52 DATAOBJURGATION,FELLY,OXALIS,ORYX,NESCIENCE,JODHPURS,SAPROPHYTIC,CARNELIAN
53 DATAFAUGH,MANTELET,STOLON,NONAGE,EXPIRY,VITUPERATION,HOYDEN,NEPENTHE
54 DATACURRICLE,AMERCE,EXEGESIS,INTERCALATE,AMICE,RENNIN,BEWRAY,SOUTACHE,OCTROI
55 DATABUNCOMBE,POSSET,CHITON,FAIENCE,CATAFALQUE
56 REM LIST OF KEYWORDS FOR SEMANTIC METHOD
57 DATAOBJECT,BELT,FOXTAILS,RAKES,NONESENSE,JUMPERS,SAPSUCKER,GARNET,FOOEY
58 DATAMANTLE,SOILING,AGE,EXPIRE,FIT,BOYHOOD,NAP,CURRY,MERCY,JESUS,CALENDAR
59 DATAMASS,RUNNY,BETRAY,MOUSTACHE,TROY,BUNK,POSTUM,COAT,FANCY,CATACOMB
60 REM LIST OF KEYWORDS FOR STANDARD KEYWORD METHOD
61 DATAOBJECT,FILLY,OXEN,ORE,NUISANCE,JOGGERS,SAP,CARNATION,FOG,MANDOLIN,STOLE
62 DATANUN,EXPERT,OPERATION,HOARD,NEPTUNE,CUBICLE,HAMMERS,EXODUS,PERCOLATOR
63 DATAMICE,WREN,BERRY,MOUSTACHE,OCTOPUS,BUN,POSSUM,KITE,FIANCE,CATAPULT
64 REM: VOCABULARY QUIZ SUCCESSIVELY CALL SUBROUTINES
65 GOSUB73
66 GOSUB83
67 GOSUB86
68 GOSUB99
69 GOSUB104
70 GOSUB121
71 IFE=0 THEN67
72 GOTO65
73 IFEC>0THEN79
74 PRINTCHR$(147)
75 PRINT"**** VOCABULARY TEST ****"
76 PRINT:FORI=1TO30:WU(I)=0:NEXT
77 PRINT"FOR EACH ITEM TYPE THE NUMBER OF THE "
78 PRINT"ALTERNATIVE WHICH BEST REPRESENTS "
79 PRINT"THE MEANING OF THE WORD.":L=30
80 IFEC>0THEN82
81 PRINT:R=RND(-RND(0)):RETURN
82 REM HOUSEKEEPING SUBROUTINE
83 IFEC>0THEN83999
84 C=4:D=30:O=1:E=0:O1=0:RETURN
85 REM RANDOMLY SELECTS CHOICES FOR VOCABULARY QUIZ, DETERMINES CORRECT ANSWERS
86 FORJ=1TOC:P(J)=0:NEXTJ
87 A=INT(4*RND(1))+1
88 P(A)=INT(D*RND(1))+1:IFWU(P(A))<>0THEN88
89 FORJ=1TOC
90 IFJ=ATHEN96
91 P=INT(D*RND(1))+1
92 IFP=P1ORP=P2ORP=P3THEN91
93 FORK=1TOJ:IFP(K)=PORP(A)=PTHEN91:NEXTK
94 IFP(K)=PORP(A)=PTHEN91
95 P(J)=P
96 NEXTJ:RETURN
97 RETURN
98 REM ASKS QUESTIONS
99 PRINT:WU(P(A))=1:IFO>1THENPRINT"□"
100 PRINTO;"--WHAT DOES "W$(P(A));" MEAN?":TT=T+TI
101 FORJ=1TOC:PRINT:PRINTAB(5);J;"--";D$(P(J)):NEXTJ:PRINT:RETURN
102 PRINT:PRINTTAB(5);J;"--";D$(P(J))
103 REM ACCEPTS ANSWERS, SCORES QUIZ, SAVES SUBSCRIPTS OF LAST 3 ANSWERS
104 PRINT:PRINT"? ";
105 GETR$(P(A)):IFR$(P(A))=""THEN105
106 PRINTR$(P(A)):GOSUB201
107 R=VAL(R$(P(A))):R=INT(R)
108 IFR>=1ANDR<=CTHEN110
109 PRINT"I NEED A NUMBER FROM 1 TO";C:GOTO105
110 PRINT
111 IFR=ATHENCS(P(A))=1:GOTO113
112 GOTO118
113 IFP(A)=10RP(A)=50RP(A)=70RP(A)=90RP(A)=120RP(A)=13THENOA=OA+1:GOTO117
114 IFP(A)=140RP(A)=160RP(A)=180RP(A)=19THENOA=OA+1:GOTO117
115 IFP(A)=200RP(A)=220RP(A)=230RP(A)=250RP(A)=26THENOA=OA+1:GOTO117
116 OC=OC+1
117 O1=O1+1
118 IFO<30THEN119
119 P3=P2:P2=P1:P1=P(A):RETURN
120 REM COUNTS QUESTIONS AND PRINTS END MESSAGE
121 O=O+1:IFO<=LTHENRETURN
122 E=1:TCOMP=TI:TI$="000000":TT=TSUBD+TND+TDI+TRATING+TEX+TLT+TCOMP
123 PRINT:PRINT"O.K.  THAT'S IT.  YOU'RE ALL DONE!":PRINT
124 REM NAME VARIABLES FOR STORAGE, GRP 1, OCC 2, SS# 3, PK 4-33, COMP 34-63
125 V$(2)=B$:V$(3)=A$:V$(1)=RIGHT$(G$,1):IFV$(1)="C"THENV$(1)="3"
126 FORI=4TO33:V$(I)=PK$(I-3):NEXT
127 FORI=34TO63:V$(I)=STR$(CS(I-33)):NEXT
128 REM TOTAL TOTPK=64, TRATING=65, TEX=66, LT TIME=67, TOTAL COMP TIME=68
129 REM TOT TIME=69, COMPA=70, COMPC=71, COMP SCORE=72
130 V$(64)=STR$(PK):V$(65)=STR$(TRATING):V$(66)=STR$(TEX):V$(67)=STR$(TLT)
131 V$(68)=STR$(TCOMP):V$(69)=STR$(TT)
132 V$(70)=STR$(OA):V$(71)=STR$(OC):V$(72)=STR$(O1)
133 REM STORE VARIABLES MEASURING SUBJECT PERFORMANCE ON DISK
134 IFB$="2"THENV$(1)="0":FORI=4TO33:V$(I)="0":NEXT
135 OPEN2,8,2,"↑OC"+A$+B$+",S,W"
136 FORJ=1TO72:PRINT#2,V$(J);CHR$(13);:NEXT:CLOSE2:END
137 REM MEANINGS OF UNFAMILIAR WORDS
138 DATA"VEHEMENT CHIDING","SPOKED WHEEL RIM","CLOVERLIKE PLANT"
139 DATA"AFRICAN ANTELOPE",IGNORANCE,"RIDING BREECHES"
140 DATA"LIVING ON DECAYING MATTER"
141 DATA"RED QUARTZ-LIKE MINERAL","EXCLAMATION OF DISGUST","SHORT CAPE"
142 DATA"A PLANT'S RUNNER",MINOR,END,REVILING,TOMBOY,"MOOD-LIFTING DRUG"
143 DATA"TWO-WHEELED TWO-HORSE CARRIAGE","A FINE","LITERARY INTERPRETATION"
144 DATA"ADD TO THE CALENDAR"
145 DATA"PRIEST'S WHITE GARMENT","CALF'S DIGESTIVE JUICE",DIVULGE
146 DATA"BRAIDED TRIMMING","CITY IMPORT TAX",HUMBUG,"ALCOHOLIC MILK DRINK"
147 DATATUNIC,"FINE PORCELAIN","CASKET SUPPORT",XXX
148 REM DIRECTIONS FOR UNFAMILIAR VOCABULARY LEARNING METHODS
149 IFG$<>"C"THEN152
150 M$="USING WHATEVER METHOD YOU BELIEVE    WILL BE MOST SUCCESSFUL "
151 M$=M$+"DURING THE TIME YOU WILL BE ALOTTED FOR EACH WORD.":GOTO163
152 M$="READING EACH DEFINITION AND THE      KEYWORD PRESENTED ALONG WITH IT "
153 M$=M$+"AND THENFORMING A MENTAL PICTURE TO REPRESENT   THE DEFINITION AND"
154 M$=M$+" THE KEYWORD IN SOME  SORT OF INTERACTING RELATIONSHIP. "
155 M1$="YOU ARE TO CONCENTRATE ON THE IMAGE     UNTIL PRESENTED THE "
156 M1$=M1$+"NEXT WORD.  "
157 M1$=M1$+"THE     MENTAL PICTURE MAKING USE OF THE KEYWORDIS INTENDED TO"
158 M1$=M1$+" HELP YOU LINK THE      UNUSUAL WORD TO IT'S MEANING."
159 M2$="  IT CAN DOTHIS SINCE THE KEYWORD SOUNDS LIKE THE "
160 M3$=M3$+" UNUSUAL WORD OR A PART OF IT, BUT HAS A MEANING WHICH CAN "
161 M3$=M3$+"BE LINKED TO THE      DEFINITION IN THE MENTAL PICTURE YOU"
162 M3$=M3$+"    FORM."
163 PRINT:PRINT:PRINT"**** INSTRUCTIONS **** ":PRINT
164 PRINT"YOUR TASK NOW IS TO TRY TO REMEMBER THE";
165 PRINT" MEANINGS OF A SET OF UNUSUAL WORDS"
166 PRINT"BY ";M$:PRINT:PRINTM1$;M2$;M3$
167 GOSUB201
168 GOTO215
169 REM DIRECTIONS FOR PRIOR KNOWLEDGE RATING TASK AND RATING SUBROUTINE
170 PRINT"□YOU ARE GOING TO BE LEARNING SOME      RELATIVELY TECHNICAL ";
171 PRINT"VOCABULARY WORDS   DURING YOUR PARTICIPATION IN THIS ";
172 PRINT"     EXPERIMENT."
173 PRINT:PRINT"     FIRST, FOR EACH UNUSUAL WORD YOU";
174 PRINT"    WILL BE ASKED TO RATE HOW WELL YOU"
175 PRINT"ALREADY KNOW THE MEANING OF THE WORD."
176 PRINT"BRIEF INSTRUCTIONS WILL BE PRESENTED."
177 PRINT:PRINT"THE RATING TASK FOLLOWS IMMEDIATELY.":GOSUB201
178 S1=RND(-RND(0))
179 FORK=1TO30
180 J=INT(30*RND(1))+1
181 IFWU(J)<>0GOTO180
182 WU(J)=1
183 PRINT"□";W$(J):PRINT:PRINT
184 PRINT"RATE THE EXTENT TO WHICH YOU HAVE       PRIOR KNOWLEDGE OF THE "
185 PRINT"MEANING OF THE ABOVE WORD BY TYPING     THE KEY FOR THE APPROPRIATE
186 PRINT"NUMBER."
187 PRINT:PRINT"  CERTAIN I DON'T KNOW MEANING        1"
188 PRINT:PRINT"  THINK I DON'T KNOW MEANING          2"
189 PRINT:PRINT"  UNSURE WHETHER I KNOW IT OR NOT     3"
190 PRINT:PRINT"  THINK I KNOW MEANING                4"
191 PRINT:PRINT"  CERTAIN I KNOW MEANING              5"
192 PRINT:PRINT:PRINT"YOUR PRIOR KNOWLEDGE RATING?";
193 GETPK$(J):IFPK$(J)=""THEN193
194 IFVAL(PK$(J))<1ORVAL(PK$(J))>5THENPRINT:PRINT"I NEED A NUMBER FROM 1 TO 5."
195 IFVAL(PK$(J))<1ORVAL(PK$(J))>5THEN193
196 PK=VAL(PK$(J))+PK
197 PRINT" ";PK$(J):GOSUB201
198 NEXT
199 RETURN
200 REM ROUTINE TO CONTINUE
201 PRINT:PRINT"TO CONTINUE PRESS THE 'RETURN' KEY."
202 GETR$:IFR$=""THEN202
203 RETURN
204 REM ROUTINE FOR CONTROL LEARNING TASK ITEM PRESENTATIONS
205 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
206 PRINT"USE ANY METHOD YOU BELIEVE MAY BE ";
207 PRINT"      EFFECTIVE TO TRY TO REMEMBER THE        MEANING."
208 RETURN
209 REM ROUTINE FOR KEYWORD LEARNING TASK ITEM PRESENTATIONS
210 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
211 PRINT"VISUALIZE A MENTAL PICTURE REPRESENTING"
212 PRINT"THE DEFINITION AND THE KEYWORD          INTERACTING IN SOME SORT OF
213 PRINT"SPATIAL RELATIONSHIP."
214 PRINT:PRINT"CONCENTRATE ON THE MENTAL PICTURE.":RETURN
215 REM FINAL DIRECTIONS BEFORE BEGINNING LEARNING TASK
216 PRINT:PRINT"THIRTY WORDS AND THEIR MEANINGS  WILL BEPRESENTED ";
217 PRINT"AT A RATE OF ONE EVERY"D/60"     SECONDS.":PRINT
218 PRINT"STUDY EACH WORD AND ITS MEANING,        CAREFULLY USING YOUR METHOD "
219 PRINT"DURING THE PRESENTATION OF EACH ITEM,   TRYING HARD TO BE SURE "
220 PRINT"YOU'RE USING THE METHOD WELL AND WILL BEABLE TO REMEMBER THE MEANING.
221 PRINT"A SHORT TEST WILL BE ADMINISTERED AFTER YOU HAVE BEEN PRESENTED ALL
222 PRINT"THIRTY WORDS AND THEIR MEANINGS."
223 REM ADDITIONAL DIRECTION FOR SEMANTIC KEYWORD GROUP
224 IFG$="E1"THENPRINT:PRINT"NOTE THAT THE KEYWORDS HAVE MEANINGS"
225 IFG$="E1"THENPRINT"RELATED TO THE MEANINGS OF THE WORDS YOUARE LEARNING."
226 IFG$="E1"ORG$="E2"THENPRINT:PRINT"REMEMBER, YOU CREATE THE IMAGES."
227 GOSUB201
228 GOTO31
```

UPDATE:
EDUCATIONAL RESOURCE CENTERS

Early reaction to our request for educational resource centers has been beyond expectation, with responses from over 200 schools, ranging from elementary schools to colleges.

As promised in the April/May issue, we will be distributing a portion of the Public Domain software that has been catalogued to date. Also, we will be sending information on the new Commodore 64, a very powerful machine for the educational market.

We are also in the process of negotiating with a number of software publishers and book publishers for educational software. We may be asking our resource centers to evaluate this new software.

We want to hear about *success stories*—so that we can publish your stories in the Commodore Magazine, and *teacher training* programs, especially if you would be willing to have other teachers from your area attend. The resource center program is gaining ground quickly so be ready! We may be requesting, from time to time, to have other educators come into your centers and have a look around.

Commodore is very happy with the results of this exciting new program. If you have any questions or suggestions please feel free to drop us a line.

The following is a list of our current resource centers:

Ms. Virginia W. Brown
Conway Public Schools
Highway 60 West
Conway, AR 72032
501 329-5630

Mr. Jeffrey A. Lewis
Math Dept.
Westwood High School
Westwood at Eight Street
Mesa, AZ 85201
602 898-3900

Mr. Bruce Eldredge
Prescott Unified School
PO Box 1231
Prescott, AZ 86302
602 445-5400 Ext. 236

Mr. Ralph Gross, Jr.
Kingman High School
400 Grandview Ave.
Kingman, AZ 86401
602 753-6216

Mr. Fred D. Cheshire
Royal Palm Computer Center
8520 N. 19th Ave.
Phoenix, AZ 85021
602 995-6211

Ms. Deborah Wilson
Desert Sun School
PO Box 338
Idyllwild, CA 92349
714 659-2191

Mr. Ronald A. Weinert
Oxnard School District
1055 South C Street
Oxnard, CA 93030
805 487-3918 Ext. 489/490

Mr. Fred Sweet
Marie College
7202 El Cajon
San Diego, CA 92115
714 697-9585

Mr. Philip T. Repino
California Data Flow Assoc.
51 Snyder Way
Fremont, CA 94536
415 792-3322

Mr. Jerome C. Ohman
Turlock Jr. High School
1441 Colorado Ave.
Turlock, CA 95380
209 667-0881

Mr. James Rogers
Ortega Middle School
1283 Terra Nova Blvd.
Pacifica, CA 94044

Mr. Harvey Williams
Assoc. Prof. of Sociology
University of the Pacific
Stockton, CA 95211
209 946-2101

Mr. Jim French
Red Hill School
100 Shaw Dr.
San Anselmo, CA 94960
415 453-0872

Ms. Linnie Sue Washburn
Deer Creek School
805 Lindley Ave.
Nevada City, CA 95959
916 265-5883

Mr. Michael Arsulich
Dept. of Educ.
San Diego County
6401 Linda Vista Road
San Diego, CA 92111 7399
714 292-3855

Ms. Ann Lathrop
Library Coordinator
Microcomputer Center
333 Main Street
Redwood City, CA 94063
415 363-5470

Mr. Wally Chute
Estancia High School
2323 Placentia Avenue
Cost Mesa, CA 92627
714 760-3410

Mr. Paul M. Ricciardi
Principal
Los Perales School
22 Wakefield Drive
Moraga, CA 94556
415 376-7272

Mr. Jim French
Teacher Learning Coop.
1111 Los Gallinas
San Rafael, CA 94903
415 499-5811

Ms. Lonne Baldwin
San Rafael High
185 Mission Ave.
San Rafael, CA 94901
415 456-0150

Mr. James P. Jackson
CAMT San Francisco State Univ.
1600 Holloway Ave.
San Francisco, CA 94132
415 469-1601

Mr. Gordon Klein
Elrancho High School
6501 S. Passons Blvd.
Pico Rivera, CA 90660
213 942-1500

Mr. Leonard T. Meuer
Vanden High School
Markley Road & Deronde Drive
Travis Air Force Base, CA 94535
707 437-4631

Mr. Joseph Wickham
Instructional Ass't.
Sequoia School
710 Martha Street
Manteca, CA 95336
209 823-1131

Mr. Glenn Fisher
Microcomputer Center
685 A Street
Hayward, CA 94541
415 881-6083

Prof. Robert W. Monk
Boys' Club of Pasadena
3230 East Del Mar Blvd.
Pasadena, Ca 91107
213 449-1953

Mr. Arthur N. Bristol
Highcrest High School
95 Highcrest Road
Wethersfield, CT 06109

Mr. Edward M. North
Taft Educational Center
Taft School
Watertown, CT 06795
203 274-2516

Mr. Richard Latorre
Notre Dame Catholic H.S.
220 Jefferson Street
Fairfield, CT 06432
203 372-6521

Mr. Daniel G. Spaneas
Wethersfield High School
411 Woliott Hill Road
Wethersfield, CT
203 503-8181 Ext. 287

Ms. Carolyn Bottomley
Norwalk Public Schools
Main Street
Norwalk, CT 06852

Ms. Carol Helstrom
Dir. of Inst.
Regional School Dist. #1
RSSC
Falls Village, CT 06031
203 824-5469

Mr. Harold J. De Pianta
Supervisor
Windsor High School
50 Sage Park Road
Windsor, CT 06095
203 688-8334

Mr. Raymond J. Magnani
Newark Center for Creative Learning
401 Phillips Ave.
Newark, DE 19711
302 368-7772

Mr. Michael Krupp
Merrit Island High School
100 Mustang Way
Merritt Island, FL 32952
305 452-1110

Dr. Casius H. Pealer
CFCC Admissions Office
Box 1388
Ocala, FL 32678
904 237-2111 Ext. 301

Mr. Joel Weiss
Central FL Community Col
PO Box 1388
Ocala, FL 32678
904 237-2111 Ext. 248

Mr. James E. Stone
Denison Jr. High School
400 Avenue A. SE
Winter Haven, FL 33880
813 294-5451

Ms. Nanette McLain
University of Central FL
College of Education
Orlando, FL 32816
275-2436

Mr. Don Woods
Southwest Junior High
2815 S. Eden Parkway
Lakeland, FL 33803
813 683 6461

Ms. Barbara A. Miller
Valdosta High School
3101 North Forrest Street
Valdosta, GA 31601

Ms. Diane F. Goldsberry
Memorial Day School
6500 Habersham Street
Savannah, GA 31405
912 352-4535

Ms. Marjorie McKenzie
Pine Mountain Regional Library
Box 709
Manchester, GA 31816
404 846-2186

Mr. W. B. Devine
Woodbury Central School
Moville, IA 51039
712 873-3128

Mr. Mark Steinberg
Marshalltown Community College
3700 South Center
Marshalltown, IA 50158
515 752-7106

Mr. Don Kissenger
Grangeville High School
910 South D Street
Grangeville, ID 83530
983-0580

Rev. John W. Milton
St. Viator High School
1213 East Oakton Street
Arlinton Heights, IL 60004
312 392-4050

Mr. Robert E. Rigney
West Pike Comm. Unit #2
Chaney Ave.
Kinderhook, IL 62345
217 432-8324

Sister Carol Royston
Principal
St. Anthony School
410 South Park Street
Streator, IL 61364
815 672-3847

Mr. H. D. Marvel
Quincy Area Vocational
219 Baldwin Dr.
Quincy, IL 62301
217 224-3770 Ext. 334

Mr. Ron Rull
Staunton High School
801 N. Deneen Street
Staunton, IL 62088
618 635-3838

Mr. Gordon T. Hardman
Burris Elem. School
RR#2
Mitchell, IN 47446
812 849-2509

Mr. Russell D. Ingram
Union Township High School
559W. 300N.
Valparaiso, IN 46388
219 750-2561

Mr. Bill Townsend
Randolph S. School Corp.
PO Box 385
Lynn, In 47355
317 874-1181

Mr. James Mason
Griffith Sr. High
600 N. Wiggs
Griffith, IN 46319
219 924-4281

Ms. Edna Vinson
Supervisor of Math
EVSC
1 S.E. Ninth Street
Evansville, IN 47708
812 426-5027

Mr. Larry B. Copeland
Franklin Central High School
6215 S. Franklin Road
Indianapolis, IN 46259

Mr. Daniel Overton
Tri-County Middle-Senior H.S.
R.D. 1 - Box 130A
Wolcott, IN 47995
219 279-2105

Mrs. Ann Hanes
Find Materials Center
407 South B Street
Richmond, IN 47374
317 966-1521

Mr. Richard G. Carey
Principal
Midland Adventist School
6915 Maurer Road
Shawnee, KS 66217
913 268-7400

Dr. Terrence V. O'Brien
Marketing Instructional Resource Ctr.
Kansas State Univ.
Manhattan, KS 66506
913 532-6008

Mr. Irwin Noyes
Dodge City Community College
Hwy. 50 bypass at 14th
Dodge City, KS 67801
316 225-1321 Ext. 239

Mr. Carl Metzger
Hoisington High School
200 E. 7th
Hoisington, KS 67544
316 653-2141

Mr. Phil Stewart
Paola High School
405 N. Hospital Dr.
Paola, KS 66071

Mr. Tom Barnes
Spearville High School
Spearville, KS 67876
316 385-2631

Mr. Nelson M. Patton
Peabody High School
900 Walnut
Peabody, KS 66866
316 983-2196

Mr. Ed R. Heinen
Otis-Bison Senior High
Otis, KS 67565
913 387-2336

Mr. Michael J. Seiler
Computer Science Dept.
Henry Clay H.S.
2100 Fontaine Road
Lexington, KY 40502
606 269-3326

Dr. Larry E. Monk
Curriculum Director
Vernon Parish Schools
201 Belview Road
Leesville, LA
318 239-3401

Ms. Pauline M. Rankin
Louisiana State Univ.
118 Himes Hall
LSU, LA 70803
504 388-1135

Mr. Barry Emery
Valentine Road
Taconic High School
Pittsfield, MA 01201
413 743-4284

Ms. Margaret Patterson
Ayer Senior High
Washington Street
Ayer, MA 01432
617 772-3357

Ms. Daphne Johnson
Eaglebrook School
Deerfield, MA 01342
413 773-5408

Mr. Bill Green
Longmeadow High School
95 Grassy Gutter Road
Longmeadow, MA 01106
413 567-3331

Mr. Richard McKnight
Waltham Public Schools
488 Main Street
Waltham, MA 02154
617 893-8050 Ext. 238

Mr. Howard W. Smith
Principal
Williamstown Public Schools
96 School Street
Williamstown, MA 01267
413 458-5707

Mr. Howard Wailgum
Russell Elem. School
Russell Street
Hadley, MA 01035
413 584-2419/5011

Mr. Richard A. Brown
Minnechaug Regional H.S.
621 Main Street
Wilbraham, MA 01095
413 596-9011

Dr. Frank M. White
Catonsville Community College
800 S. Rolling Road
Baltimore, MD 21228
301 455-4443

Mr. Lawrence Greenleaf
Crosby Jr. High
SAD #34
Belfast, ME 04915

Mr. Rod Joyal
Searsport District High School
Church Street
Searsport, ME
207 548-2523

Mr. M. J. Winter
Educational Computer Lab
Michigan State Univ.
East Lansing, MI 48824
517 353-6337

Mr. Gary W. Dode
Whitehall Jr. High
Slocum St.
Whitehall, MI 49461
616 893-1655

Mr. David G. Britten
Muskegon Catholic Central H.S.
1851 Barclay
Muskegon, MI 49441
616 755-2201

Mr. Bill Besonen
Ontonagon Area High School
Ontonagon, MI 49953
906 884-4164

Mr. Richard Plecha
Field Elem. School
1000 S. Haggerty Rd.
Canton, MI 48188
313 397-2151

Mr. Robert Metzger
Grand Haven Junior High School
1400 S. Griffen Street
Grand Haven, MI 49417
616 842-5010

Mr. Lawrence Shaltis
Grass Lake High School
1000 Grass Lake Road
Grass Lake, MI 49240
517 522-8494

Mr. Norman J. Eisenberg
O. E. Dunckel Middle School
32800 W. 12 Mile Road
Farmington Hills, MI 48018
313 553-0310

Ms. Mary Jane Barry
Villa Duchesne
801 S. Spoede Rd.
St. Louis, MI 63131

Ms. Judith Stoeri
Linden High School
7201 Silver Lake Road
Linden, MI 48442
313 735-7821

Miss Linda Kuzmin
East Kentwood High School
6178 Campus Park Dr. SE
Kentwood, MI 49508
616 698-6700

Mr. William L. Brown
Principal
Miller Elementary School
850 Spencer Road
Brighton, MI 48116
313 229-5000 Ext. 127

Mr. James L. Clor
Assistant Director
Angus Elementary School
3180 Hein
Sterling Heights, MI 48077
313 268-7060

Ms. Marge Farrand
Consultant
Ann Arbor Public Schools
344 Gralake
Ann Arbor, MI 48103
313 994-2286

Mr. William Halbrook
City of Ladue School District
9703 Conway Road
St. Louis, MI 63124
314 994-3508/7080

Ms. Sharon Billiau
Dryden Community School
3866 Rochester Rd.
Dryden, MI 48428
313 796-2266

Ms. Caroline Garrick
Computor Tutor
502 E. Front
Traverse City, MI 49684
616 941-5320

Mr. Rollie Schmidt
Cooper Sr. High School
8230 - 47th Ave. N
Minneapolis, MN 55428

Mr. Dick Quast
SCRAM
Morgan High School
Morgan, MN 56266
507 249-3188

Mr. Bill Heck
Robbinsdale Area Schools
4148 Winnetka Ave. North
Minneapolis, MN 55427
612 533-2781

Mr. Darwin Fosse
Westwood Elementary School
Box 539
Prior Lake, MN 55372
612 447-2178

Mr. Douglas S. Hed
Edina Public Schools
5701 Normandale Road
Edina, MN 55424
612 927-9721

Mr. Richard M. Bertke
CEMREL
3120 S. 59th Street
St. Louis, MO 63139
314 781-2900

Mr. Sigmund Gorski
Queen Anne School
14111 Oak Grove Road
Upper Marlboro, MO 20772
301 249-5000

Mr. Joe Munger
Sikeston High School
200 Pine Street
Sikeston, MO 63801
314 471-5440

Mr. Mike Myrhow
Jefferson High School
Box 176
Boulder, MT 59632
406 225-3317

Mr. Ryan Overbeek
Flathead High School
Kalispell, MT 59901
406 755-7130

Ms. Sue Dolezal
Stevensville High School
300 Park Street
Stevensville, MT 59870
406 777-5481

Mr. Bill Chalgren
Libby High School
Route 3
Libby, MT 59923
406 293-6204

Mr. John Oberlitner
Polson High School
Polson, MT 59860
406 883-4381

Mr. Harvey B. Herman
Chemistry Dept.
University of NC
Greensboro, NC 27412
919 379-5714

Mr. Robert Eddy
Instructor
Wauneta Public Schools
PO Box 368
Wauneta, NE 69045
308 394-5700

Mr. Warren Neumeister
Vilas School
Alstead, NH 03602
603 835-6351

Mr. Gordon E. Smart
Coe-Brown Northwood Adademy
Route 4
Northwood, NH 03261
942-5531

Mr. Robert Proulx
Principal
Lebanon Jr. H.S.
75 Bank Street
Lebanon, NH 03766
603 448-3056

Mr. Philip Paskowitz
Canaan Elem. School
Canaan, NH 03741
603 523-4312

Mr. John R. Best
Township of Ocean Schools
West Park Ave.
Oakhurst, NJ 07755
201 531-6600

Mr. John Peraino
Levitt Adm. Bldg.
Willingboro Public Schools
Salem Road
Willingboro, NJ 08046
609 871-9000

Mr. John J. Hopton
Franklin Township Public Schools
Macafee Road School
Somerset, NJ 08873
201 249-9097

Ms. Barbara P. Caralioto
North Arlington High School
222 Ridge Road
North Arlington, NJ 07032
201 991-6800 Ext. 239

Mrs. Peg Novicki
Franklin School
May Street
Hawthorne, NJ 07506
201 427-1300 Ext. 210

Mr. Michael S. Levinson
Rutgers Prep. School
1345 Erston Ave.
Somerset, NJ 08873
201 454-5600 Ext. 63

Mrs. Anne C. Sabol
McCorristin High School
175 Leonard Ave.
Trenton, NJ 08610
609 586-3705

Mr. John C. Handfield
Electronic Service Assoc.
The Acreace Bldg.
PO Box 186
Medford, NJ 08055
609 953-08055

Mrs. Jean Rappaport
Verona
30 Gould Street
Verona, NJ 07044
201 239-2422

Mr. Henry J. Petersen
Wayne Computer Cooperative
50 Nellis Drive
Wayne, NJ 07470
201 696-3157

Mr. Stephen M. Shuller
Director
Center for Micro Education
Two Babcock Place
West Orange, NJ 07052
201 731-8400

Mr. John Melchoir
Parsippany-Troy Hills Township School
1213 S. Beverwyck Road
Parsippany, NJ 07054
201 263-7292

Dr. Phyllis Reagle
Clark County Comm. Coll.
3200 E. Cheyenne Ave.
North Las Vegas, NV 89030
702 643-6060 Ext. 280

Mr. Joel P. McKee
Charleston Heights Library
800 Brush St.
Las Vegas, NV 89107
702 878-3682

Mr. Chuck Kirst
Cold Spring Harbor H.S.
Turkey Lane
Cold Spring Harbor, NY 11724
516 692-8600

Mr. Paul S. Drago
Freeport High School
Freeport Computer Lab
South Brookside Ave.
Freeport, NY 11520
516 546-5322

Mr. Tony Asplin
Middleville Jr. H.S.
Middleville Road
Northport, NY 11768
526 261-9000

Ms. Elayne Schulman
Staten Island Academy
715 Todt Hill Road
Staten Island, NY 10304

Mr. Alan Berkowitz
Ryan Jr. High
64-20 175th St.
Flushing, NY 11365
212 353-7040

Ms. Joan Magnani
Copiaque High School
Dixon Avenue
Copiaque, NY 11726

Mr. Robert Bergevin
Ausable Valley Central School
Middle High School
Clintonville, NY 12924
518 834-7116

Ms. Sheila Dolgowich
Math Dept. Chairman
B-K-W High School
Berne, NY 12023
518 872-0251

Dr. Daniel Brovey
Queens College
Office of Micro. Use in Ed.
Box 50 Dept. Ed.
Flushing, NY 11367
212 520-7396

Mr. Arnold Friedman
Great Neck Computing School
17 Picadilly Rd.
Great Neck, NY 11023
516 482-5359

Dr. Eugene F. Stafford
Iona College
New Rochelle, NY 10801
914 636-2100

Mr. D. Paul Servey
Wellsville High School
126 W. State Street
Wellsville, NY 14895
716 593-2291

Bro. R. W. Harris
Chairman
Iona Prep. School
Wilmot Road
New Rochelle, NY 10804
914 632-0714

Dr. Dolores Shanahan
Commack Public Schools
Kings Park Road
Commack, NY 11725
516 493-3573

Mr. Wilfred J. Gray
Amityville School District
Route 110 and N. Drive
Amityville, NY 11701
516 691-7900 Ext. 212

Mr. William Kroohs
Eldred Central School Dist.
Eldred, NY 12732
914 557-6014

Mr. Ira Rosen
Middle School
Shoreham - Wading River
Randall Road
Shoreham, NY 11786
516 929-8500

Mr. Aaron Friedman
Acting Principal
Intermediate School 246
72 Veronica Place
Brooklyn, NY 11226
212 444-6436

Rabbi David Serkin
Hebrew Academy of Long Beach
530 W. Broadway
Long Beach, NY 11561
515 432-8285

Mrs. Harriet Pitkof
Computer Super.
I.S.59 Queens
132-55 Ridgedale St.
Springfield Gardens, NY 11413
212 527-3501

Dr. Eugene Galanter
The Children's Computer School
21 West 86 Street
New York, NY 10024
212 580-1335

Mr. Ted Detjen
Mid-Hudson Reg. Micro. Support Service
175 Rt. 32 N.
New Paltz, NY 12561
914 255-1450

Ms. Annette Paparello
Shoreham Wading River School District
Randall Road
Shoreham, NY 11786
516 929-8500

Mr. Robert E. Gwinn
Orchard Park Central School
S. Lincoln Ave.
Orchard Park, NY 14127
716 662-6229

Mr. Walter Edmeyer
Wellington School
1822 Fishinger Road
Columbus, OH 43221
614 457-7883

Mr. Richard L. Preston
Otsego High School
PO Box 168
Tontogany, OH 43565
419 823-7693

Mr. Jonathan Secaur
Roosevelt High School
1400 North Mantua Street
Kent, OH 44240
216 673-9595

Mr. Joseph W. Fair
United Local District
8143 S.R.9
Hanoverton, OH 44423
216 223-1521

Mr. Allen Cohen
Laurelville School
Laurelville, OH 43135
614 332-2021

Dr. James S. Meyer
St. Gregory's College
1900 W. MacArthur
Shawnee, OK 74801
405 273-9870

Mr. Charles Scott
Elliott Lake Secondary School
303 Mississauga Ave.
Elliot Lake on P5A 1E8
705 848-7162

Mr. Larry Clark
Judson Baptist College
400 E. Scenic Dr.
The Dalles, OR 97058
503 298-4455 Ext. 256

Mr. Gary W. Ferrington
Instructional Technology Studies
University of Oregon
Eugene, OR 97403
503 686-3468

Mr. William R. Loercher
Donegal High School
Box 297
Mount Joy, PA 17552
717 653-1871

Mr. Carl W. Richie
Muhlenberg High School
Sharp Ave. and Francis St.
Laureldale, PA 19605

Dr. Francis J. Wunderlich
Villanova University
Physics Dept.
Villanova, PA 19085
645-4860

Mr. Richard Seaman
Hermitage School Dist.
123 N. Hermitage Road
Hermitage, PA 16148
412 981-8750

Mr. Fred Darling
Chadds Ford Elementary School
Box 173 - Rourel
Chadds Ford, PA 19317
388-1112

Mr. Dan R. Knepp
PCCC
720 Museum Road
Reading, PA 19611
215 374-8331

Dr. Thurman R. Kremser
Department of Physics
Albright College
PO Box 516
Reading, PA 19603
215 921-2381 Ext. 388

Mr. Wayne E. Bradburn
Wilson High School
Grandview Blvd.
West Lawn, PA 19609
678-8065

Mr. Robert W. Critchfield
Ass't. Superintendent
Derry Area School District
R.D.1
Derry, PA 15627
694-2791

Mr. Bob Douglas
University of Pennsylvania
3718 Locust Walk 130
McNiel/CR
Philadelphia, PA 19104
243-6735

Dr. Bruce Burt
West Chester Schools
320 N. Church St.
West Chester, PA 19380
436-7160

Mr. Wes Fasnacht
West Chester State College
West Chester, PA 19380
436-2828

Dr. Leroy J. Tuscher
Lehigh University
524 Broadhead Ave.
Bethlehem, PA 18015
861-3241

Ms. Julie Howell
Phoenixville Area School District
Main Street
Phoenixville, PA 19460
215 933-1912

Mr. Dennis Plymette
Upper Moreland Middle School
Orangeman's Road
Hatboro, PA 19040
674-4185

Mr. Richard S. Gilpin
Octorara Int. School
R.D.1
Atglen, PA 19310
215 593-5153

Mr. Curtis McQueen
Pittsburgh New Church School
299 Le Roi Road
Pittsburgh, PA 15208
731-7421

Ms. Ellen Morrow
Main Line Computer Center
155 E. Lancaster Ave.
Wayne, PA 19087
687-8500

Mr. James P. Cooper
Bryn Athyn Church Schools
PO Box 277
Bryn Athyn, PA 19009

Ms. Donna M. Chalfant
Greensburg Inst. of Tech.
302 W. Otterman Street
Greensburg, PA 15601
412 837-3330

Father Roberto Kringel
Episcopal Cathedral School
PO Box 13305
Santurce, Puerto Rico 00908
723-5478

Mr. Daniel J. Andrews
Social Street School
706 Social Street
Woonsocket, RI 02895
401 769-4724

Mr. Jerry W. Jones
Aiken Technical College
PO Drawer 696
Aiken, SC 29801
803 593-9231

Mr. William Rivers
Midlands Tech. College
PO Box 2408
Columbia, SC 29202
803 738-1400 Ext. 4283

Mr. James L. White
West Junior High School
500 Soo San Drive
Rapid City, SD 57701-8195
605 394-4033

Dr. Jerry Willis
Educational Computing Center
Texas Tech University
Lubbock, TX 79409
806 742-2381

Ms. Katherine Mays
Bay City School District
Box 631
Bay City, TX 77414
713 245-5771

Mr. William R. Welch
Thomas J. Rusk Middle School
411 N. Mound Street
Nacogdoches, TX 75961
713 560-4608

Vernon Newsom
Mansfield Independent School Dist.
609 E. Broad St.
Mansfield, TX 76063
817 473-1178

Dr. John R. Friedrich
San Antonio College
1300 San Pedro Ave.
San Antonio, TX 78284
512 733-2855

Mr. Austin B. Cron
PSJA High School
1229 South I Road
Pharr, TX 78577
512 787-6675

Mr. Byron Craig
Abilene Ind. School Dist.
Box 981
Abilene, TX 79601
915 677-1444

Mr. Ed Franz
Live Oak Elementary School
12301 Welcome Dr.
San Antonio, TX 78233
655-6241

Mr. John P. Herndon
McCoy Elementary
2425 McCoy Road
Carrollton, TX 75006
214 323-5928

Mr. Joe E. Robinson
Junction High School
1700 College Street
Junction, TX 76849
915 446-3326

Dr. Lyndal R. Hutcherson
Carrollton/Farmers Branch I.S.D.
PO Box 186
Carrollton, TX 75006
214 323-5700

Mr. Terry Hawks
Roy High School
2150 West 4800 South
Roy, UT 84067
801 825-9766

Mr. Carlton L. Ramsey
Hargrave Military Academy
Chatham, VA 24531
804 432-2481 Ext. 28

Mr. Bruce Harding
learning Resources Center
Rappahannock Comm. College
South Campus
Glenns, VA 23149
804 333-4024

Ms. Katherine A. McGlynn
Addison Cent. Supervisory Union
Charles Ave.
Middlebury, VT 05753
802 388-4232

Mr. David D. Wilson
Headmaster
Long Trail School
Box 313
Dorset, VT 05251
802 867-5717

Mr. David Cozzens
Charlotte Central School
Hinesburg Rd.
Charlotte, VT 05445
802 425-2771

Reg Wadpoups
Kamiakin Jr. High
14111 132RD N.E.
Kirkland, WA 98033
828-3301

Mr. Kenneth W. Beasley
Waitsburg High School
Waitsburg, WA 99361

Ms. Patsy Ethridge
Cleveland High School
5511 - 15th Ave. S.
Seattle, WA 98108
587-3550

Pat McIntyre
St. Martins College
Lacy, WA 98503
206 491-4700

Mr. Jeff Tremblay
Arlington High School
135 So. French St.
Arlington, WA 98223
206 435-211-

Mr. Robert Patton
Homestead High School
5000 W. Mequon Road
Mequon, WI 53092
414 242-2400

Mr. James W. Beatty
Chemistry Dept.
Ripon College
Ripon, WI 54971
414 748-8123

Mr. Ken Sack
St. Lucy's Grade School
3035 Drexel Ave.
Racine, WI 53403
414 554-1091

APPLICATION FORM

Educational Resource Center

Your Name: _____

Computer Site Name: _____

Address: _____

_____

Telephone #: _____

Contact Person #1 _____

Contact Person #2 _____

Number of Commodore Computers at this location

Vic-20 _____

4016 _____

4032 _____

8032 _____

SuperPET _____

Number of non-Commodore Microcomputers at this location:

and Type _____

_____

Are you willing to be listed in Commodore Publications as an Educational Resource Center?

Yes_____, No_____

Do you have the staff and facilities to evaluate educational software?

Yes_____, No_____

Fold Here. Staple or Tape Bottom.

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 4 NORRISTOWN, PA 19401

POSTAGE WILL BE PAID BY ADDRESSEE

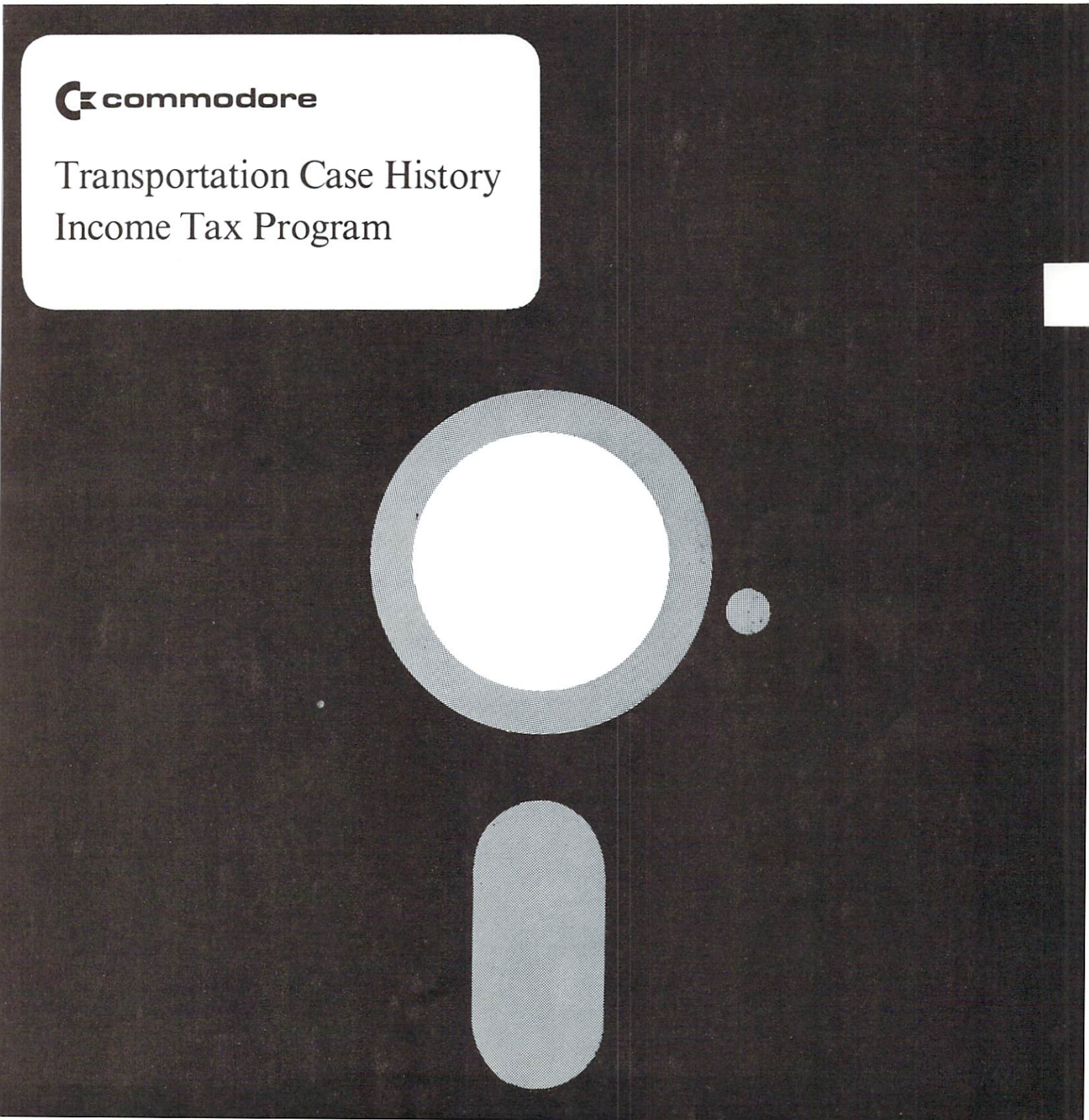
Commodore Business Machines, In.
THE MEADOWS
487 DEVON PARK DRIVE
WAYNE, PA 19087

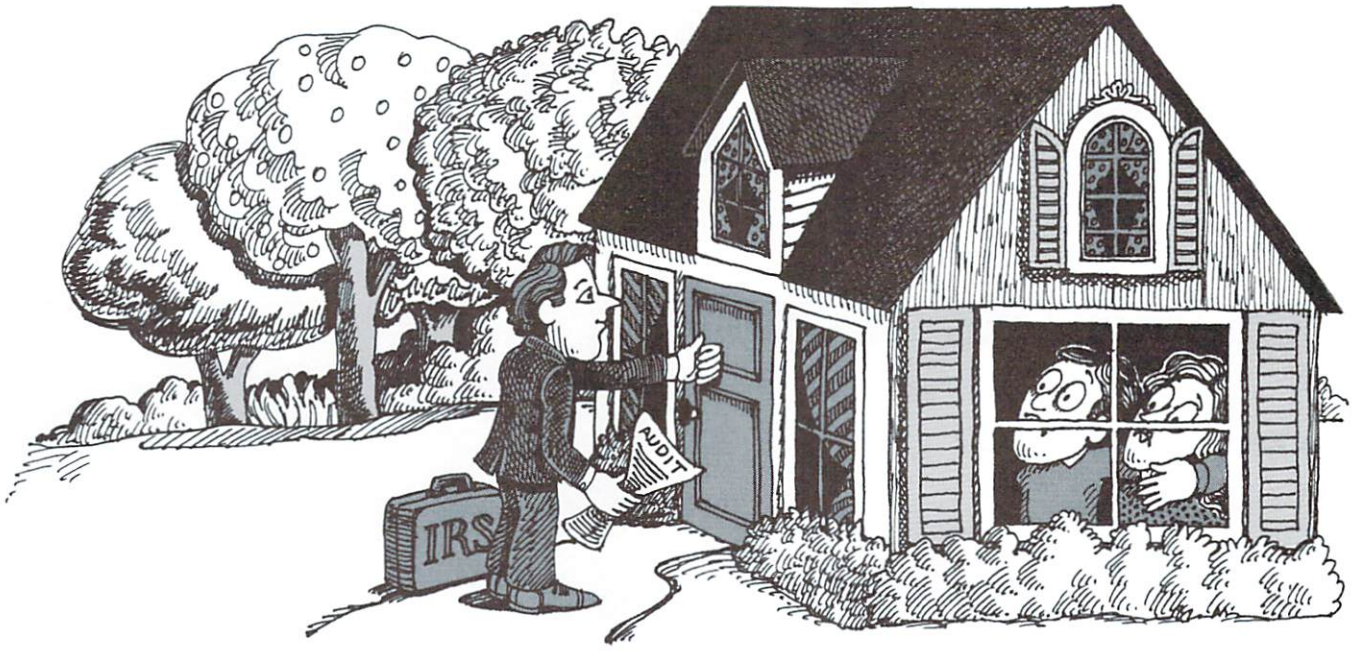
Attn: Dr. Bruce Downing
Director - Industry Marketing

BUSINESS

commodore

Transportation Case History
Income Tax Program





Commodore's Federal Income Tax Preparation System

by
George G. Severance

April 15, 1982 marked my 12th year as an income tax practitioner, averaging 500 tax returns per year for a 12-year total of 6,000 returns prepared. The first 10 years, the returns were prepared manually (5,000 manual returns—ugh!). The past two tax seasons, our office used Commodore's Federal Income Tax Preparation System. Using this system, the past tax season was easily the smoothest, most productive, and most profitable of the 12 tax seasons.

Prior to the advent of the microprocessor, there were three methods of preparing income tax returns: manually, using an in-house mini-computer, or using an outside service bureau. The manual method is relatively slow and cumbersome; the mini-computer approach is very expensive; and the service bureau method is slow, expensive, and not particularly responsive.

During the past two tax seasons, our three person office acquired and completely paid for three CBM 8032 CPUs, three CBM 4040 Dual Drive Floppy Disks, and a letter quality printer. Using Commodore's Tax Preparation Software, this equipment enabled us to substantially increase our office productivity during the tax preparation season and utilize the equipment in year-round applications in other areas. In fact, it has revolutionized our office procedures to the point that we have a headstart on the office of the future.

Commodore's Tax Preparation System handles about 95% of the tax forms and schedules that a professional tax practitioner is likely to encounter during a typical year. For the few forms not processed by the system, it is a simple matter to prepare them manually and insert the bottom line figures in the appropriate lines of the tax program.

The system enables productivity increases in every phase of the tax preparation business: actual preparation of the return, checking and editing of the return, and printing and assembly of the completed return.

The more complex the tax return, the greater the savings in preparation time. For example, if a client has dividends from 25 different companies, the preparer merely lists the company name and the amount of dividends from each company, hits the T for total button and the computer automatically computes the total and carries that amount forward to the first page of the tax return. Contrast that to using an adding machine to compute the total (at least twice), and manually carrying the total forward to page one of the return.

As another example of time savings, income averaging is a method of tax computation involving averaging the current year's taxable income with the taxable income for each of the prior four years. The computer computes the tax by income averaging in about 12 seconds. The computation includes tax for a married couple filing jointly as well as for husband and wife each filing separately. The same computation performed manually would take five to ten minutes. On average, the system reduces preparation time for the total 1040 by 50%.

In checking a completed tax return, the checker looks for two types of errors—mathematical errors and theory errors. A completed tax return may be mathematically correct, yet wrong because the preparer missed something for which the taxpayer was eligible. Commodore's system eliminates the need to check for math errors. The computer automatically

performs all addition, subtraction, calculates medical exclusions, general sales tax, and then computes the tax liability. Once you gain confidence in the system and know that the computer will not make mathematical errors, the checker need not manually add up those 25 dividends to see if the computer added correctly. The checker would verify that the company names and dividend amounts were correct and move on to the next area. The time saved thereby allows the checker more time to check for theory errors—entitlements due the taxpayer such as income averaging, installment sales treatment of capital gains, over 55 exclusions on the sale of residence, etc. On average, the system reduces the checking time by 50%.

Every tax practitioner has experienced the frustration of a phone call from the client who "forgot" to tell you that he just received notification from Micromania Corporation that the correct dividend amount was $500 rather than the $200 amount he had given you previously, or that 30% of No Nuke Utility Company dividend was non-taxable, or that he neglected to tell you that he sold his Montezuma Gold shares at a profit last year. To manually edit a completed return usually involves changing almost every page on the return as the revised numbers ripple through the return and alter the mathematical totals on most pages. This takes additional preparer time, additional checker time, and additional assembly time. To edit a completed return under Commodore's system is simplified, since all the mathematical computations are done automatically. The revised numbers are merely placed on the appropriate schedules and forms and each section of the return retotalled. Then the changed pages are reprinted and reassembled. On average, the system reduces editing time by 70%.

The tax preparation system automatically prints one page at a time on standard government tax forms. We printed on three part (four part is available) blue on white, carbonless NCR forms available from Spee-d-Tax Forms of Mundelein, Illinois. This method eliminates the need for extensive use of a copying machine and reduces the printing and assembly time by about 25%. The enhanced finished product is a beautifully packaged, typed, letter quality, error-free tax return. Our clients reacted very favorably to our professionally prepared, quality product.

Commodore's Federal Income Tax Preparation System has a well documented manual to lead the preparer step by step through the tax return. It includes a section for the preparer to organize the return in advance and determine exactly which forms and schedules are to be used and the sequence of their use. A hotline number is available to answer specific questions and solve problem areas. Last season, Brian Padol, the author of the software program, started a tax newsletter to pass on helpful hints and to provide a forum for preparer inputs. Mr. Padol wrote the initial tax software program in 1977 and each year adds some enhancements based upon user experience, changes in the tax laws, and advancements in programming techniques.

Of course, the use of a Commodore computer system is not limited to income tax preparation. We have converted our general ledger write-up accounts from a service bureau method to an in-house general ledger program. All of our client correspondence is processed on a word processing program. We are using the same program for billing purposes. We have also created our own data base management system for our tax and accounting clients. From this data base, we may generate mailing list labels, determine which clients are above a certain tax bracket, calculate the average fee charged, identify problem areas or problem clients, etc.

Besides the obvious advantages of increased productivity and reduced operating costs, Commodore's system eliminates one other major tax preparer problem: preparer stress. The knowledge that the system will process many tax returns quickly and accurately reduces the cumulative pressures associated with most tax preparation operations. Our backlog never exceeded more than a dozen tax returns. We did not work beyond 6:00 P.M. during the week and found it necessary to work no more than three to four hours on each Saturday and Sunday. Compared to most successful practitioners, that workload represents a significant reduction in tax preparation time.

Commodore's Tax Preparation System enables us to choose our direction of growth. We now have the option and the capabilities of either increasing the number of returns we process each year and/or improving the quality of our practice by concentrating on the more complex, involved returns. Commodore's computer system allows the professional practitioner to multiply himself and apply his imagination to the solution of numerous problems, tax or otherwise. C=

Advance Transportation Case History

by Sue Revels

A Company on the Move

Allied Van Lines is one of the premier names in moving. The company, through independent agents across the country, handles everything from moving families to transporting specialized cargo such as delicate computer equipment and priceless art exhibits.

Advance Transportation, one of the top movers in the country, is an Allied agent in Philadelphia. In addition to the normal residential moves, Advance Transportation maintains two warehouses for storage. They also provide special facilities and equipment to meet the needs of corporate customers like SUN Oil and TV Guide, both headquartered in Philadelphia.

Allied maintains a main frame computer system at their headquarters in the midwest to help agents establish uniform quotes for similar items hauled a set distance, but that system does nothing to help agents run their business.

Paperwork and Headaches

John Heck manages Advance's warehouses and oversees many of the company's corporate moves. A recently retired executive of a large, main frame computer manufacturer, Heck knew of the efficiencies microcomputers provide and went to work looking for a system to help cut down on large amounts of paper record used in the day-to-day operation of the warehouses.

"A customer's items are stored in large sealed containers in each warehouse, with each customer assigned a unique 'lot number,' " Heck pointed out. "The problem was that a customer could have items spread between many containers that all refer back to that one number. In the past, records were kept in a card file and we had to cross reference between all the individual records to find where things were and which containers were empty. We also used this mix of records to bill for storage.

"I knew the first step was setting up a data base system where we would have access to all the information at one time. After evaluating most of the microcomputers on the market we chose a Commodore 8096 with 8050 disk and printer. Commodore offered excellent price/performance and was one of the easiest to use. And that was important because most people using the system had little or no previous computer experience.

John Heck demonstrates the Commodore system that has dramatically reduced the warehouse's large amount of paperwork.

Another factor in the decision-making process was software. He recalled how impressed he was with range of real business applications available from third party vendors for Commodore machines. Finally, Advance's data base was established using "THE MANAGER" application from Canadian Micro Distributors.

Allied Warehouse Control System

A customer file was designed using the customer's lot number as the key. The record also contained space for the identification numbers of up to 14 storage containers. Information about the total cubic space used for storage and other billing information could also be entered.

"We experimented with the system until a workable design for the data entry screens and reports was reached," explained Heck. The important thing is to sit down beforehand and decide what information you need and how it will look. Another point is to be tolerant. Switching over to a computerized system requires some time and probably a number of changes until everything is just right.

"THE MANAGER allows you to easily modify screen formats, report criteria, and other information without starting from scratch. This feature was a real time saver. If I wanted to insert a line or move a field to a different spot the change was made in a few moments.

Heck designed the system with two data entry screens, one for basic information, and the other for billing. "Even though we didn't do billing right away we had the option to add

that information to the record later without changing existing data," said Heck.

After all the basic information was established, Advance personnel could enter a container number and immediately see who it belonged to and if there were any other containers assigned to the customer. The lot number could also be entered and the entire record displayed. "Just having this information available speeded up getting a customer's shipment when it was requested and managing warehouse space more effectively," Heck explained.

THE MANAGER's math function was then employed to calculate the amount to be billed from the basic information entered about a customer. Billing information is then automatically displayed on the second page of the data entry form, which serves as an invoice. When a payment is received it is entered in the appropriate spot, keeping the record current.

The 'accumulate' function was used to add up the number of containers in use, providing an accurate record to base decisions on whether they could service new customers.

When a customer takes delivery of items in storage the record is noted and final invoice prepared. The record is then removed from active status with a printed copy placed in a single master file that replaces many different paper forms.

Time Savings and Improved Customer Service

"The time savings using the system for record keeping and invoicing have been substantial," said Heck. "Before we created invoices every three months. Now we have the flexibility to bill on much shorter cycles, managing cash flow more closely. And the report generator function allows me to get a print-out of any information by selected customers or the entire file.

"Because it takes less time to find items in storage, we are more responsive to customer requests and this gives us a competitive advantage.

"We are also putting the computer to work in other areas such as word processing. WORDCRAFT is used for correspondence and customer mailings. I've found it to be one of the most complete systems available. I especially like the way text is displayed exactly as will be printed using the built-in control functions, and the ease of editing.

"Customized mailings are also created and sent to customers periodically reminding them that items are still in storage."

Future Plans

Advance Transportation is planning to expand the computer's role by using a modified version of FRIS (Freight Rating Invoicing System) to more accurately estimate and bill for shipments.

They are also exploring connecting systems at several locations via modems to quickly transmit important information.

Heck captured the feeling at Advance Transportation by concluding: "Our system has given us greater flexibility by providing timely, accurate information. And we see the computer expanding into other areas to give that benefit to others in the organization."

```
            ADVANCE WAREHOUSE CONTROL SYSTEM (AWCS)

COMPANY CODE: [Turner, Inc]                    START DATE: [03/01/81]

LOT# : [04563]        WAREHOUSE: [32]

LOC's: [432]  [781]  [135]  [   ]  [   ]  [   ]  [   ]  [   ]  [   ]  [   ]
       [   ]  [   ]  [   ]  [   ]

TOTAL LOCATIONS: [03]     TOTAL CUBIC FEET: [  750]

-------------------------------------------------------------------------

COSTS:

     STORAGE: $[135.00]   INS: $[15.00]     VALUE: $[  8000]

     # MONTHS BILLED: [3]      Total Monthly Costs: $  150.00
```

The system-generated customer file contains space for identifying up to 14 storage containers. Information about the total cubic space used for storage can also be entered.

STATEMENT

ADVANCE TRANSPORTATION CO., INC.
Philadelphia, PA 19121-3596

DATE 08/25/82
ACCOUNT NUMBER 04563

Turner Construction Company
1104 Bunker Hill Drive
Harleysville, PA 19478

AMOUNT ENCLOSED $

RETURN THIS PORTION WITH PAYMENT

DATE	CHARGES AND CREDITS	AMOUNT
	Opening Balance	$ 450.00
04/30/82	Payment	- $ 450.00
08/25/82	Storage and Insurance from 05/01/82 to 08/31/82	$ 450.00
08/25/82	Closing Balance	$ 450.00

PAY LAST AMOUNT IN THIS COLUMN

Thank You

Z-RAM™ Opens Up The World Of CP/M™ To The Commodore Computer!

Z-RAM, a circuit board that fits inside the Commodore 4000 and 8000 computers, adds a Z-80A microprocessor and 64K of Random Access Memory (RAM), tripling the current maximum user memory!

Also provided with Z-RAM is the CP/M operating system, adding a new dimension to the Commodore computer by finally making the vast world of CP/M software available to its users.

With the addition of the Z-RAM board, Commodore computers will operate with 96K RAM and use both the 6502 and Z-80 processors. Several modes of operation are possible:

- With Z-RAM installed, all earlier versions of Commodore computers (regardless of memory capacity) can function as 32K machines. All current programs will run using 32K memory and the standard 6502 processor.
- Z-RAM permits the 6502 to use the full 96K of memory to operate the new expanded versions of Wordcraft Ultra, WordPro -Plus™ and other expanded Commodore programs.
- The Z-80A processor will operate in 64K memory and will run the industry standard CP/M operating system. WordStar™, Super-Calc™, and Accounting II Plus™ are only a sample of the fantastic CP/M programs now available to run on your Commodore computer!

Contact Your Nearest Commodore Dealer Today . . .
You'll Be So Glad You Did!

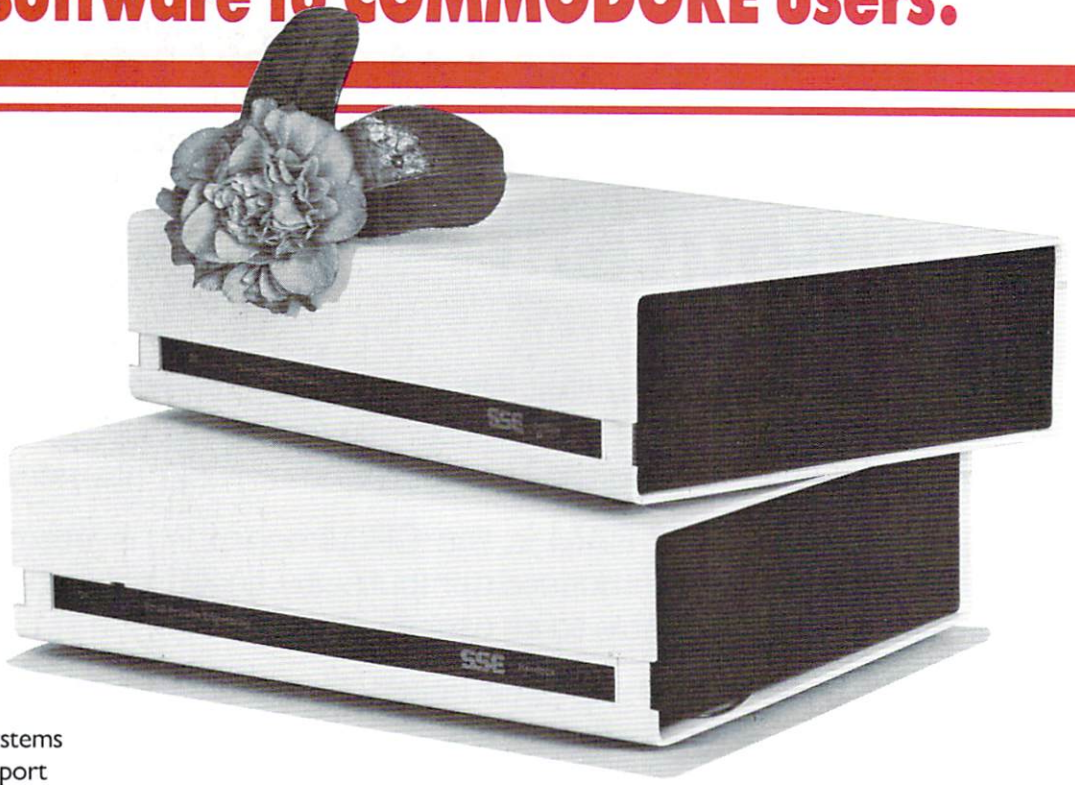
Distributed by:

COMPUTER MARKETING SERVICES INC.

(609) 795-9480
300 W. Marlton Pike, Cherry Hill, New Jersey 08034

Z-RAM is a trademark of Madison Computer
CP/M is a trademark of Digital Research

SOFTBOX® unlocks the world of proven business software to COMMODORE users.



Simply by plugging the Small Systems SOFTBOX into the IEEE-488 port and loading the CP/M disk, your Commodore PET/CBM will run under the world's most popular disk operating system. No internal connections or modifications are required.

A complete computer system in itself, the SOFTBOX uses the PET or CBM as its terminal while talking to any combination of Commodore and Corvus drives. A single command returns you to PET BASIC. The SOFTBOX utility disk gives you complete software control over your configuration.

Terminal Emulation

Applications packages designed to work with specific terminals (like ADM3A, TVI 912 or Hazeltine 1500) will need no modification for the 80-column CBM, since the SOFTBOX allows your computer to emulate any of these devices.

Add Corvus Hard Disk

Expand your CP/M file storage capacity with reliable Corvus drives. With the Corvus Mirror Option and a standard video cassette recorder, you have a fast, inexpensive disk backup. When you purchase the SOFTBOX and Corvus from Small Systems, you'll receive everything necessary to convert your Commodore to a hard disk-based CP/M system.

Features

- CP/M version 2.2 with menu driven system configuration.
- Z80 CPU running a 4MHz with no wait states.
- 64K RAM.
- Operates with any 2000, 3000, 4000 and 8000 series Commodore computer.
- Supports up to 8 Commodore 3040, 4040, or 8050 disk drives.
- Supports up to 4 Corvus hard disks of 6, 11, or 20 MBytes.
- RS232 serial interface with software-definable baud rates for use with a printer, modem or terminal.
- Diskette containing operating system with utilities.
- Comprehensive user manual.

A full range of CP/M languages and applications software on Commodore 8050 or 4040 format is available.

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

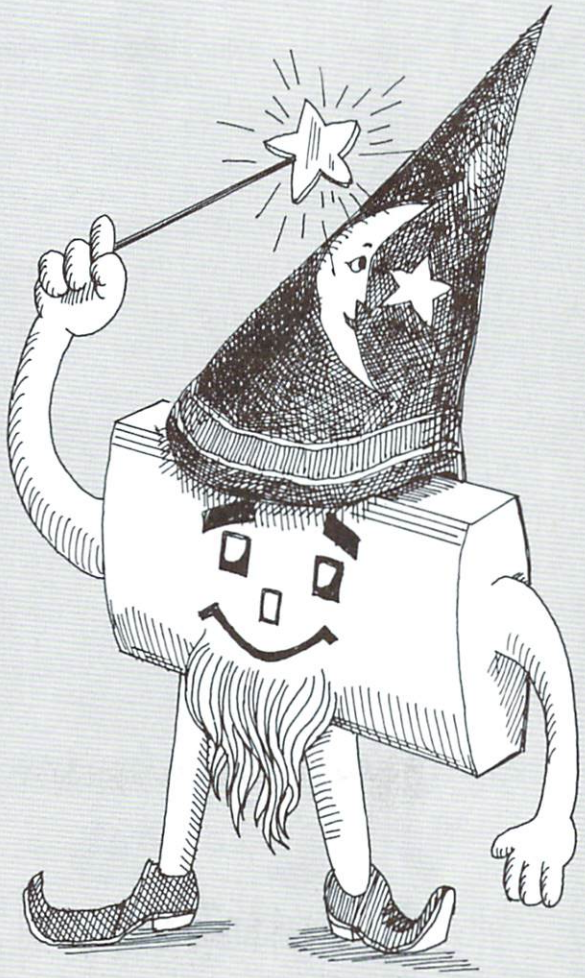
For more information on how you can use the world of proven business software on your PET and CBM, contact your local Commodore dealer. Or, for more information and the name of the dealer nearest you, call or write us today.

SSE **(415) 964-8201**

SMALL SYSTEMS ENGINEERING

222-B View Street • Mountain View, CA 94041

VIC-20
The friendly computer



The VIC Magician

Creating Attractive Screen Titles

Michael S. Tomczyk
Product Marketing Manager

Displaying an attractive screen title is always a nice way to start a BASIC program. You can even let the user choose his own title, and display it dramatically when the program starts!

One thing to remember when choosing a title is to try to keep the length of the title LESS THAN 22 CHARACTERS including spaces, because the VIC displays 22 characters on each line and your title will "spill over" to the next line if it's longer.

Centering your title

There are lots of ways to make attractive titles, but one of the easiest techniques is CENTERING your title at the top of the screen when the program begins.

To center your title in the middle of a line, the first thing to do is count the number of characters in your title. Let's say your title is "VIC MAGICIAN."

□□□□□□□□□□□□□□□□□□□□□□
□□□□□VIC MAGICIAN□□□□□

This title has 12 characters including the space. Now subtract that total from 22 (the number of characters on one line). The answer (22-12) is 10. That means you have 10 spaces "left over" after PRINTing your title on the line. To CENTER the title, you want to have an equal number of spaces on each side of the title, so next we DIVIDE the "left over" spaces by 2. Since 10/2 = 5 we know we need to put 5 spaces on both sides of our title if we want to CENTER it.

The following one-line program CENTERS the title "VIC MAGICIAN" on the screen by CLEARing the screen and then PRINTing the title 5 spaces over from the left margin (**Note:** To make each space in your program, press the long SPACE BAR once, quickly. Any space you include inside the quotation marks will appear as a space on the screen when the title is PRINTed. Notice that you don't have to worry about the spaces on the RIGHT side of the title because they're automatically left blank.):

Type this line and hit the RETURN key:

10 PRINT" [SHIFT] [CLR/HOME] [SPACE] [SPACE] [SPACE] [SPACE] [SPACE] [RVS/ON] VIC MAGICIAN"

The VIC uses "shorthand" to display the CLEAR command inside quotation marks, so when you hold down [SHIFT] and press [CLR/HOME] the VIC displays a REVERSE HEART which means "CLEAR." The SPACES you typed appear as blank spaces, like this:

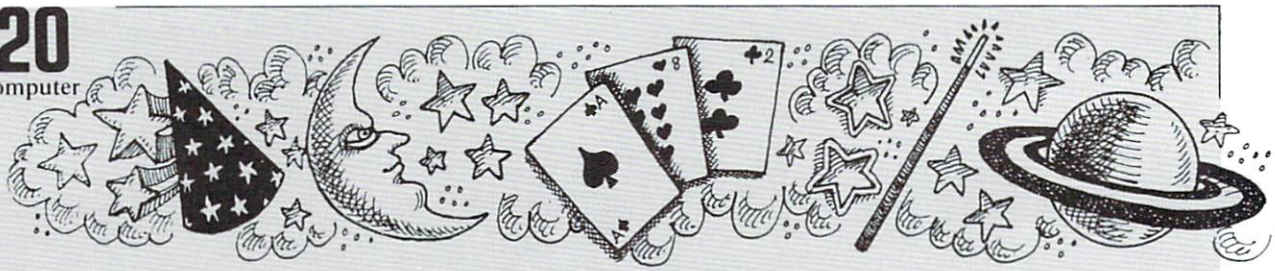
10 PRINT"♥ VIC MAGICIAN"

To display your title, type the word RUN and press the [RETURN] key.

If you have a problem or make a mistake, type LIST and hit [RETURN], then retype the line.

Using the SPC Command

There's another way to put spaces in your BASIC programs . . . by using the SPC command. If you type PRINT SPC (5) your program will insert 5 spaces. Remember the SPC command is never enclosed in quotation marks (") but is always OUTSIDE of the quotation marks . . . and you must always use the PRINT command before the SPC command. Here's how you →



can use the SPC command to PRINT the line you just typed:

10 PRINT" CLR/HOME "SPC(5) "VIC MAGICIAN"

> NOTE: From now on, CLR/HOME means "Hold Down the SHIFT key and press the CLR/HOME key.

Here's another example of the SPC command:

10 PRINTSPC (5) "MOVES 5 SPACES"SPC(3) "MOVES 3 SPACES"

Displaying Your Title in Reverse

Now that you've created a title. Let's "dress it up" a little by REVERSING the title colors. To do this, you'll have to RETYPE LINE 10. This time, just before you type the title, hold down the CTRL key and press the RVS ON key. This makes the title print in REVERSE when you RUN the program. Notice that when you hold down CTRL and press RVS ON, the VIC displays a REVERSE R. Type the following and hit the RETURN key:

TYPE THE "9" KEY

10 PRINT" SHIFT CLR/HOME SPACE SPACE SPACE SPACE SPACE CTRL RVS ON VIC MAGICIAN"

The line you typed should look like this on the screen:

10 PRINT" R VIC MAGICIAN"

Now type RUN and hit RETURN to see the new title. What if you want to REVERSE THE WHOLE LINE, including spaces? Type this line exactly as shown and hit RETURN, then type RUN and RETURN to see it:

10 PRINT" SHIFT CLR/HOME CTRL RVS ON [5 spaces] VIC MAGICIAN [5 spaces]"

This time we put the REVERSE ON command at the beginning of the first blank space and we added 5 BLANK SPACES ON THE RIGHT SIDE of our title. When you want to PRINT REVERSE SPACES (solid blocks) on the screen, you have to include all the spaces that you want to appear as solid blocks. That's why we include the right side spaces here.

Letting the User Choose a Title

One of the nicest cosmetic touches you can add to a title is letting the user choose his own title. The following program asks the user to type in a title, then centers it at the top of the screen. You can use these two lines at the beginning of any BASIC program. (Note that CLR/HOME means hold down the SHIFT key and press CLR/HOME):

10 PRINT" CLR/HOME TITLE";:INPUTT$
20 PRINT" CLR/HOME ";:FORX=1TO(22-LEN (T$))/2:PRINT" SPACE ";:NEXT:PRINTT$

After you enter this program, type the word RUN and hit the RETURN key. The screen will clear and say, "TITLE?" Type in a title and hit RETURN. The title appears automatically centered at the top of your screen.

The key here is using the INPUT statement in Line 10 to define the title typed in by the user as T$. Now, in Line 20, we can use a special BASIC command called LEN to determine the LENgth of T$. LEN (T$) counts the number of characters in the title, which was INPUT in Line 10 as "T$".

Remember the centering formula we used at the beginning of the article? Well, this whole formula is included in Line 20 like this: (22-LEN(T$))/2. If we use VIC MAGICIAN as our title, the LENgth is 12 characters,

and (22-12)/2 = 5. So this formula gives us the number of spaces on ONE SIDE of our centered title. (If you're wondering why we have 2 parentheses after T$, it's because all BASIC formulas must have the same number of left and right parentheses and the 2 right parentheses balance out the 2 left parentheses. This is an oversimplification but it's important to remember because one of the most common mistakes when creating formulas for calculation is not balancing left and right parentheses.

Next, we use a COUNTING command to tell the VIC to count 5 spaces and PRINT our title. We do this by saying, FOR X = 1 TO 5 (remember our whole formula equals 5), PRINT a SPACE. The semicolon (;) means print everything next to each other, and the NEXT command means "keep PRINTing a SPACE until the upper limit (5) is reached." After our 5 spaces are PRINTed, we PRINT T$ which is our title.

An Easy Way to Reverse a User Defined Title
The following program reverses the user-defined title and displays it at the top of the screen, flush left:

```
10 PRINT" CLR/HOME TITLE";:INPUTT$
20 PRINT" CLR/HOME ";:PRINT" CTRL
 RVS ON " T$;:FORX = LEN (T$) TO21:PRINT
" CTRL  RVS ON  SPACE ";:NEXT
```

Adding Colors to Titles

Another dimension you can add to your screen titles is COLOR. The color command works just like the RVS ON command. Hold down the CTRL key and press the color key you want. The following program PRINTs a title in red, then returns the color to blue so the rest of the program will continue in blue. CTRL RED means hold down the CTRL key and press the key marked RED on the keyboard:

TYPE THE "3" KEY

```
10 PRINT" CLR/HOME TITLE";:INPUTT$
20 PRINT" CLR/HOME ";:PRINT" CTRL
 RVS ON  CTRL  RED "
T$;:FORX = LEN (T$)TO21:PRINT" CTRL
 RVS ON  SPACE " ;:NEXT:PRINT" CTRL  BLUE "
```

TYPE THE "7" KEY

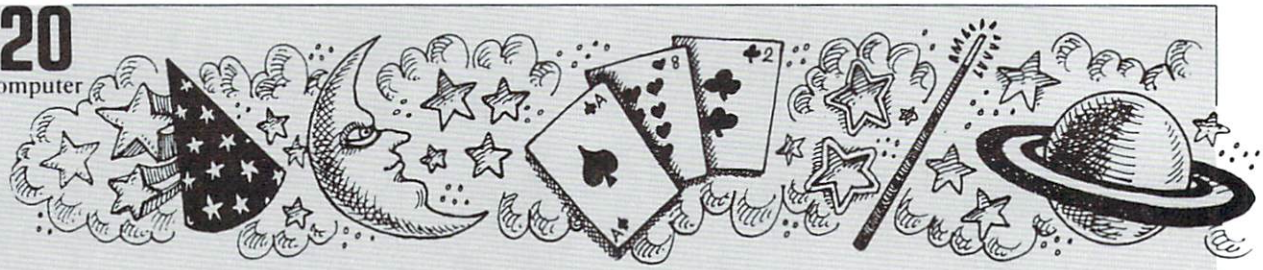
Using Titles in Long Programs
You will probably find yourself writing long programs which cause the screen to SCROLL UP as more information is added to the bottom of the screen. This scrolling action can make your screen title disappear off the top of the screen.

One way to prevent this is to repeat the two title lines in your program at the point where the title scrolls off . . . OR . . . you can make the title a GOSUB and put a GOSUB where the title will scroll off. Here's an example using the last title we used:

```
10 PRINT" CLR/HOME TITLE";:INPUTT$
15 PRINT" CLR/HOME ";:PRINT" CTRL
 RVS ON "T$;:FORX = LEN(T$)TO21:
PRINT" CTRL  RVS ON  SPACE " ;:NEXT
20 FORX = 1TO23:PRINT"YOUR PROGRAM
HERE":NEXT:GOSUB 2000:RETURN
30 PRINT"PROGRAM RUNS BEYOND 23 LINES"
100 END
2000 PRINT" HOME ";:PRINT" CTRL
 RVS ON " T$;:FORX = LEN(T$)TO21:
PRINT" CTRL  RVS ON  SPACE " ;:NEXT
```

HOLD DOWN CLR/HOME WITHOUT SHIFTING

First we centered and PRINTed the title, then we RAN a



program which went to the bottom of the screen, and added a GOSUB to Line 2000 which repeated our screen title at the point where the program would have scrolled down and "pushed" the title off the top of the screen. To see what this program accomplishes, now delete GOSUB2000:RETURN in Line 20 and delete line 2000. Now the program will scroll the title off the top of the screen when it reaches the bottom of the screen.

Experiment!
There are lots of other ways to create interesting screen titles . . . for example, try centering the title in the MIDDLE of the screen (hint: use the CRSR DOWN key in a PRINT statement, and the same centering formula we used for centering the title horizontally). You can also create borders around your titles, on both sides or on top and bottom, by using a graphic character.

Try putting graphic lines or bars above or below your titles, like this:

```
5 PRINT "SHIFT CLR/HOME";: FORG = 1TO22:
PRINT" ─ ";:NEXT
10 PRINT" CLR/HOME VIC MAGICIAN"
20 FORG = 1TO22:PRINT" ─ ";:NEXT
```

insert above title
title line(s) here
insert below title

Hold down the SHIFT key and type the "C" key to get this graphic symbol

Here's another variation to try . . . this program **centers a one-line title vertically on the screen.** Try changing the number 10 in Line 20 to move the title higher or lower on the screen:

CRSR DOWN WILL APPEAR AS A REVERSE Q on your screen.

```
10 PRINT" CLR/HOME TITLE";:INPUTT$
20 PRINT" CLR/HOME ":FORX = 1TO10:
PRINT" CRSR DOWN ";:NEXT
30 FORX = 1TO(22-LEN(T$))/2:PRINT" SPACE "
;:NEXTX:PRINTT$
```

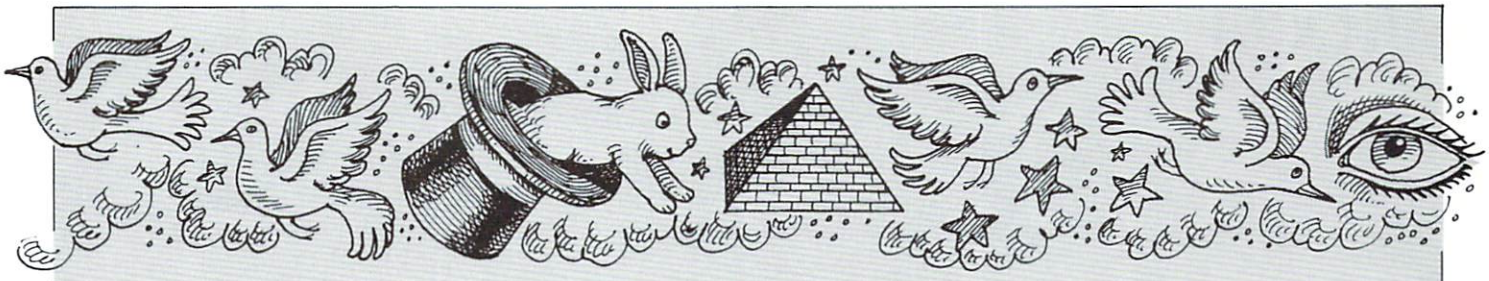
Notice that in Line 20 we told the VIC to PRINT ten CURSOR-DOWNs . . . moving the position of the cursor 10 spaces down . . . then in Line 30 we used our CENTERING formula to center the title in the middle of the screen. You can include spaces, cursor up or down movements, Control Reverse On and Off, and Control Color commands . . . as long as you put them in quotation marks. Just PRINT them like any letter, number or graphic symbol and they will appear in your program.

Michael S. Tomczyk, Product Marketing Manager

As a final variation you might want to "announce" your title with a SOUND EFFECT. Try putting this one-line program on the line IMMEDIATELY BEFORE the line that PRINTs your title:

```
25 POKE36878,15:FORM = 200TO250:POKE
36876,M:NEXTM:POKE36876,0
```

Now you're on your own. Try different colors, sound effects, screen positioning, and other variations. C=



The VIC Magician

Introducing "For . . . Next" Loops

FOR . . . NEXT loops are probably used more often than any other BASIC command, except PRINT. These loops are used for 3 purposes: 1) to insert a **time delay** in your program, and 2) to count a series of number values, in order, and 3) to repeat an action in your program a specified number of times.

Time Delay Loops
A "time delay loop" looks like this:

```
10 FORT=1TO200:NEXT
```

This line means, "count to 200 before going on the next action in the program" . . . but the word NEXT doesn't mean "next action." It means, FOR1TO200, count the NEXT NUMBER (in this case, 1,2,3, and so on up to 200). First, here's a program with NO time delay:

```
10 PRINT"VIC-20 ";
30 GOTO10
```

Type RUN/STOP to "stop" the program, then type this line and hit RETURN:

```
20 FORT=1TO900:NEXT
```

Type LIST and hit RETURN to see your program, then type RUN and RETURN to see how it works. The time delay loop has slowed down the program . . . and will slow down ANY BASIC PROGRAM. You can, in fact, insert Line 20 anywhere in your BASIC programs to create time delays. Try changing 900 in Line 20 to get a longer or shorter time delay.

Counting, Counting . . .
The only difference between a **time delay loop** and a **counting loop** is where you put the NEXT command.

In a time delay loop, you should put the NEXT immediately after the first part of the loop. But in a counting loop, the order should be: first, the FOR part of the loop; second, the program action; and third (immediately AFTER the program action), the NEXT command. Here's an example that counts from 1 to 10 and PRINTs the numbers on the screen:

```
10 FORN=1TO10:PRINTN:NEXT
```

You could count any series of numbers, like this:

```
10 FORN=150TO200:PRINTN:NEXT
```

But counting is also used for other purposes. For example, you might want to make the VIC-20 move through a progression of musical note values, in order, to see how they sound. Here's how:

```
10 POKE36878,15
20 FORM = 128TO255:POKE36876,M:
POKE 36876,M
40 NEXTM
50 POKE36876,0
```

This program POKEs the VIC speaker numbered 36876 with the musical note value of M. The note value M is actually a range of numbers from 128 to 255. We defined this range in Line 20 (FORM=128TO255). Then we POKE36876 with the value of M and at the same time we PRINT the POKE numbers that match each musical note.

Line 40 provides the NEXT command which makes the program move from 128 to 129 to 130, and so on until it reaches 255. Note that the NEXT command doesn't have to be on the same line as the FOR . . . part of the loop. Line 50 simply turns off the speaker.

There are numerous examples of FOR . . . NEXT loops being used to count values or proceed through ranges of numbers. The color program on Page 34 of the VIC-20 owner's manual is another good example.

Repeating BASIC Commands
You can make a BASIC command go back and repeat over and over again using a FOR . . . NEXT loop. Here's an example:

```
10 FORX=1TO10:PRINT"COUNTING":NEXT
```

Here, the important thing is to put the part you want to repeat BETWEEN the FOR and NEXT portions of the loop. This program PRINTs the word "COUNTING" 10 times on the screen.

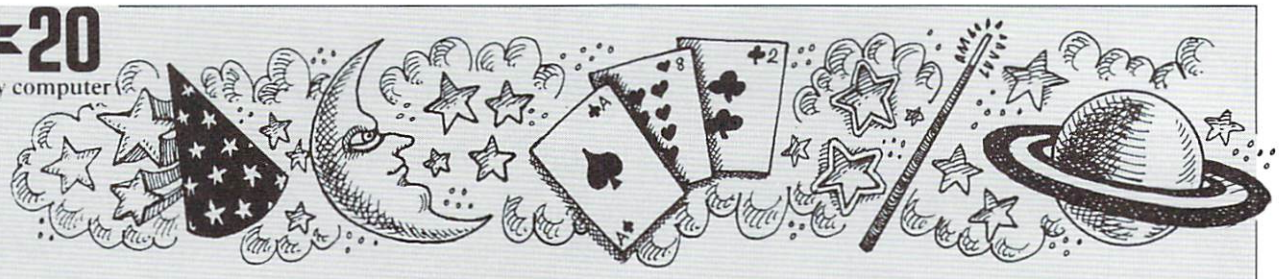
Here's another type of counting program:

```
10 FORG=1TO21:PRINTCHR$(166);:NEXT
```

This program line PRINTs a graphic symbol 21 times across the top of the screen. It tells the VIC-20 to PRINT the symbol represented by the CHR$ value (166). CHR$ values for all characters are shown on Page 146 of the owner's guide. Try substituting some other numbers. Here's another variation which gives the SAME RESULT:

```
10 FORG=1TO21:PRINT" ";:NEXT
```

(HOLD DOWN THE C= key and type " + ")



Finally, how about seeing the symbols represented by all the CHR$ values? This program will do that:

```
10 FORC=0TO191:PRINTCHR$(C);:PRINT
" [SPACE] [SPACE] " C:NEXT
```

Using More Than One Loop

You can use several FOR . . . NEXT loops in your programs, and even separate them . . . but one thing you CANNOT DO is "stagger" your loops OUTSIDE of each other. This means, if you use 2 loops, you must arrange your program so the loops are "nested" inside each other. The best way to demonstrate this is to show you a correctly "nested" program and an incorrect program:

CORRECT

```
10 FORA=1TO5:PRINT"HELLO"
  20 FORB=1TO10:PRINT"GOOD-BYE"
  30 NEXTB
40 NEXTA
```

INCORRECT

```
10 FORA=1TO5:PRINT"HELLO"
20 FORB=1TO10:PRINT"GOOD-BYE"
30 NEXTA
  40 NEXTB
```

Improper "nesting" (arranging) of loops in your program will cause the VIC-20 to give you a "NEXT WITHOUT FOR" error message. You will also get this error message if you forget to put in a NEXT command after the FOR portion.

Summary

These are only a few examples using FOR . . . NEXT loops. Try experimenting with the formats given above and you'll find that FOR . . . NEXT loops will make your programming more efficient, interesting, and creative. C=

Enhanced Business Forms on the VIC 20

Here's a modified and expanded version of last issue's VIC business form program. The original program simply printed blocks and headings, and it was up to the user to fill them in. In *this* program, the VIC asks you for all the data needed to fill in the form, then prints out your data and computes the ending inventory for each part.

The first part of the program is essentially the same as last issue's program. We'll go through it and point out the differences.

The first new line is line 4. In order to make the program easier to follow, The input section is at the end of the original program. After we do the inputting, we'll jump back to the beginning. Line 18 allows the user to output the date on the top of the form. Later on, the program asks you to input the date.

The other major change to the first part is the size of the blocks. In order to make the output easier to read, the size of each input number is limited to 7 digits, and all the blocks are the same size. Lines 30 and 40 were changed so that the headings would fit on top of the new blocks. The block size was changed in lines 150 to 170. Line 160 is the most important here. The VIC will print the data you've input (more, shortly). Let's go on to the second half of the program first.

Line 4 directs the VIC to go to line 300, the input section of the program. Lines 302 to 308 allow you to input the date using 3 string variables, A$, B$, and C$. When the program is run, the VIC will ask you for the month, day and year separately. These values go into storage spaces A$, B$, and C$. Back in line 18, we print out the contents of these three variables.

Lines 310 to 400 are the key to the input section, with 310 being the most important. The DIM statement is like making reservations at a motel. Here we're telling the VIC to reserve space in its memory. DIMZ$ (5, 7) says "Save us 5 rooms, each with 7 boxes, under the name of Z$." When we input our data, we'll put a number in the appropriate box in the appropriate room. The first 6 boxes will contain values that were input, and the seventh box will contain the value of the ending inventory, which the program will compute. In line 160, we print out the contents of each box under the correct heading.

Let's go on to line 320. The variable J runs from 1 to 5, since there were 5 items we kept track of in the original example. Line 330 tells us what item we're working on. Line 340 goes from 1 to 7, since there are 7 blocks for each item (Part #, Beginning Inventory, etc.). We read in a title for our input in line 350 from the data in line 600, and the VIC prints the title and asks for the user to input T$ in line 360. We don't want to input the ending inventory, so when K = 7, we bypass the read and input lines and go to line 375. In order to keep the output readable, we have to format each number we input. Since this is a repetitive process, we can use a subroutine. Line 363 directs the VIC to go to the subroutine beginning at line 700.

The 700 subroutine looks at the length of T$, the number we just input in line 360. We want each number to be 7 digits long when we're done. If the length is greater than 7, we'll take only the first 7 digits (line 705). Lines 710 to 770 simply add spaces to T$, making it a total of 7 characters long. Thus if T$ = "123", line 750 will execute and Z$ (J,K) will become " 123 ", 7 characters wide.

WHOA!!! What's Z$(J,K)??? That's our variable for keeping track of the input. J and K are loop variables, from lines 320 and 340. J is the item we're currently working on, and K is the category. The first time we input a number, we will be at Z$(1,1)—first item, first category (Beginning Inventory). Next we do Z$(1,2)—first item, second category (Receipts). We'll input 6 numbers for each item—lines 340 to 370 exe-

cute 6 times, each time checking T$ in the subroutine to make sure it's 7 digits long.

When we've finished inputting for an item, K will equal 7 and we jump out of the loop and go to line 375, where the VIC calculates the ending inventory. R will equal the value of beginning inventory (Z$(J,2)), plus the value of receipts (Z$(J,3)), minus the value of shipments (Z$(J,5)). Line 376 checks R to eliminate negative numbers. Line 377 turns R into a string, T$, so we can format our ending inventory just like our inputs.

Now that we've finished all 6 inputs for the first item, we want to go on to item 2. The RESTORE in line 380 allows the VIC to read in the data in line 600 more than once. Line 390 jumps us back up to line 320, and J is now 2. When we get down to line 350, D$ will be read in as "PART #". Without the RESTORE, 'READ D$' would try to read the next string after "AIRBILL #", (notching), and we'd get an OUT OF DATA ERROR. After we go through the J loop 5 times for each of our five items, we drop down to line 400, which goes back to line 5, the start of our original program.

The last thing to do is our output. Line 18 prints our previously-input date and lines 30 and 40 print the column headings. Lines 150 to 170 print the blocks. Line 161 is simply a continuation of line 160 (it was too long to fit on one line). Line 160 outputs each value of Z$(J, K) we input earlier. The loop variable A goes from 1 to 5, and line 160 will print each successive value under the correct column heading.

That's it folks!! The program may appear complicated, but it really isn't. The two main operations are inputting the data and then outputting it in the blocks. It wouldn't be difficult to adapt this program to your own needs—keep the basics the same, and change headings and spacing to suit!!

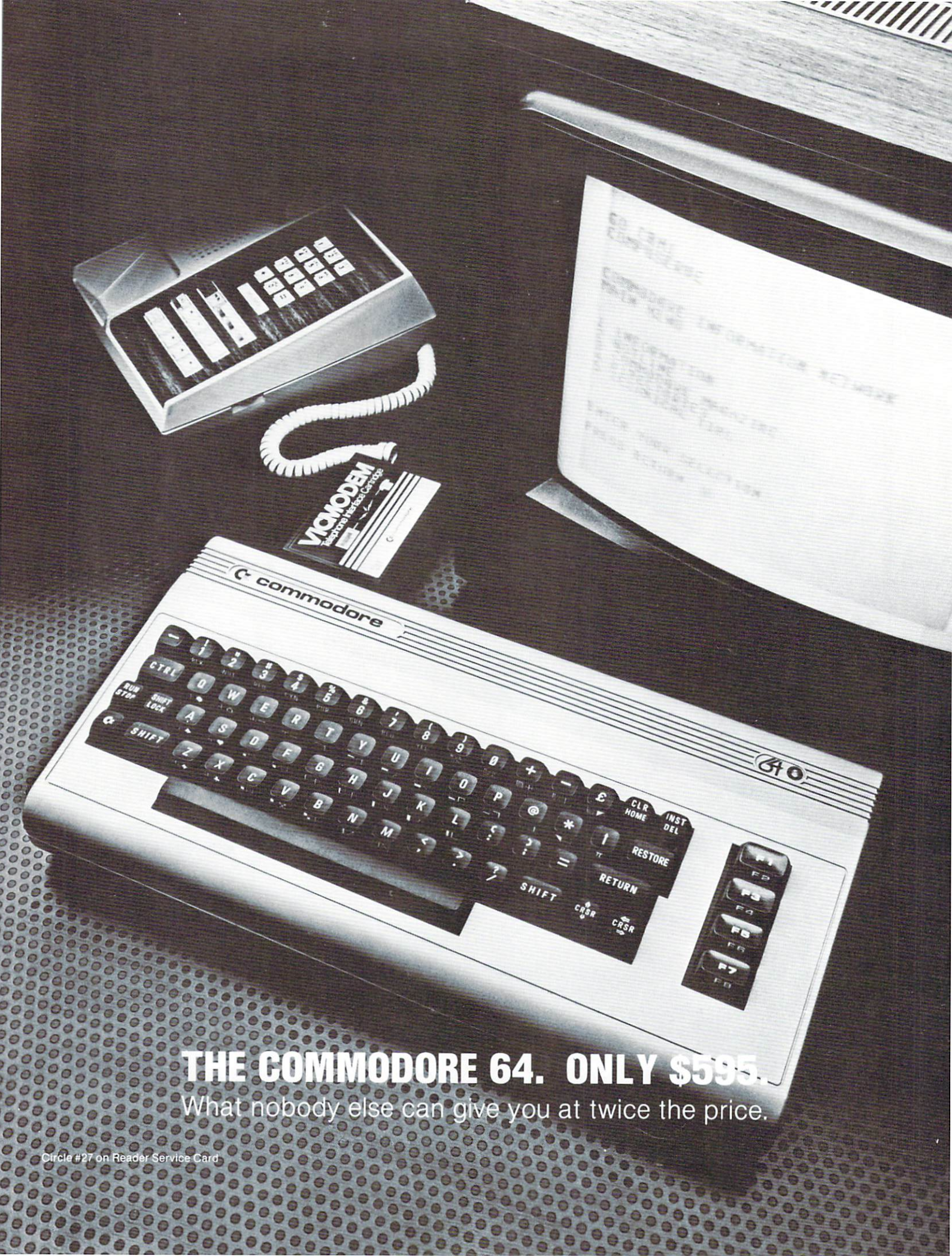
```
SHIPPING DEP'T INVENTORY

DATE:    JULY    23    1982

PART NO.   BEG. INV   REC'D   PACK LIST #   SHIP MENTS   AIR BILL #   END INV
12-A       254         54      W43F          23           243-D3       285
12-F       32          4       65-0          0            -            36
20-5A      5423        675     34-57         1622         324          4476
23         65453       3423    656-8H        25100        54-2332      43776
4F         250         25      54-0I4        120          34           155

READY.
```

```
2 REM ** VIC BUSINESS FORMS - PART 2 **
3 REM ** BY JOSEPH SICILIANO **
4 GOTO 300
5 OPEN4,4:CMD4
10 PRINTCHR$(14)"      SHIPPING DEP'T INVENTORY"CHR$(15)
15 PRINT:PRINT
18 PRINT"                        DATE:      "A$"     "B$"    "C$
20 PRINT:PRINT
30 PRINT"     PART      BEG.      REC'D      PACK      SHIP      AIR       END"
40 PRINT"      NO.      INV                 LIST #    MENTS    BILL #     INV"
50 PRINT
100 FOR A=1TO5
150 PRINT"  ┌────────┐ ┌────────┐ ┌────────┐ ┌────────┐ ┌────────┐ ┌────────┐ ┌────────┐
┐"
160 PRINT"  !"Z$(A,1)"! !"Z$(A,2)"! !"Z$(A,3)"! !"Z$(A,4)"! !"Z$(A,5)"! !"Z$(A,6
);
161 PRINT"! !"Z$(A,7)"!"
170 PRINT"  └────────┘ └────────┘ └────────┘ └────────┘ └────────┘ └────────┘ └────────┘
┘"
180 PRINT
190 NEXT A
200 END
300 REM ** INPUT ROUTINES **
302 PRINT"□       TODAY'S":PRINT
305 PRINT:PRINT:PRINT
306 PRINT"MONTH ";:INPUTA$:PRINT
307 PRINT"DAY ";:INPUTB$:PRINT
308 PRINT"YEAR ";:INPUTC$:PRINT
310 DIMZ$(5,7)
320 FORJ=1TO5
325 PRINT"□"
330 PRINT"     ITEM # "J
335 PRINT:PRINT:PRINT
340 FORK=1TO7
342 IF K=7THEN375
350 READD$
360 PRINTD$;:INPUT T$
363 GOSUB 700
365 PRINT
370 NEXT K
375 R=VAL(Z$(J,2))+VAL(Z$(J,3))-VAL(Z$(J,5))
376 IF R<0 THEN R=0
377 T$=STR$(R)
378 GOSUB700
380 RESTORE
390 NEXT J
400 GOTO 5
600 DATAPART # ,BEG INVENTORY ,RECEIPTS ,PACKING LIST # ,SHIPMENTS ,"AIRBILL # "
700 REM ** SUBROUTINE TO FORMAT OUTPUT
705 IF LEN(T$)>7THEN Z$(J,K)=LEFT$(T$,7):RETURN
710 IF LEN(T$)=7 THEN Z$(J,K)=T$:RETURN
720 IF LEN(T$)=6 THEN Z$(J,K)=" "+T$:RETURN
730 IF LEN(T$)=5 THEN Z$(J,K)=" "+T$+" ":RETURN
740 IF LEN(T$)=4 THEN Z$(J,K)="  "+T$+" ":RETURN
750 IF LEN(T$)=3 THEN Z$(J,K)="  "+T$+"  ":RETURN
760 IF LEN(T$)=2 THEN Z$(J,K)="   "+T$+"  ":RETURN
770 IF LEN(T$)=1 THEN Z$(J,K)="   "+T$+"   ":RETURN
READY.
```



THE COMMODORE 64. ONLY $595.

What nobody else can give you at twice the price.

Circle #27 on Reader Service Card

"THE COMMODORE 64 COULD BE THE MICROCOMPUTER INDUSTRY'S OUTSTANDING NEW PRODUCT INTRODUCTION SINCE THE BIRTH OF THIS INDUSTRY."

–SHEARSON/AMERICAN EXPRESS

They're speaking to a group as interested as anyone else in the future of computers: the people who buy stock in the companies that make computers.

If, on the other hand, you're a person whose livelihood depends on a personal computer—or whose leisure time revolves around one—what follows should impress you even more than it impresses investors.

MIGHT MAKES RIGHT.

The value of a computer is determined by what it can do. What it can do is largely determined by its memory.

The Commodore 64's basic RAM is 64K. This amount of power is unusual enough in a micro at any price.

At $595, it is astonishing.

Compared with the Apple II+®, for instance, the Commodore 64™ offers 33% more power at considerably less than 50% of the cost.

Compared with anything else, it's even more impressive.

PILE ON THE PERIPHERALS.

Because the basic cost of the 64 is so low, you can afford to buy more peripherals for it. Like disk drives, printers, and a telephone modem that's priced at around $100.

This means you can own the 64, disk drive, printer and modem for a little more than an Apple II+ computer alone.

HARD FACTS ABOUT SOFTWARE.

The Commodore 64 will have a broad range of custom software packages including an electronic spreadsheet; business graphics (including printout); a user-definable diary/calendar; word processing; mailing lists, and more.

With BASIC as its primary language, it is also PET BASIC compatible.

The Commodore 64 will also be programmable in UCSD PASCAL, PILOT and LOGO.

And, with the added CP/M* option, you will have access to hundreds of exciting software packages.

THE FUN SIDE OF POWER.

The Commodore 64 can become very playful at a moment's notice.

You can use Commodore's plug-in game cartridges or invent your own diversions. All will be enhanced by brilliant video quality (320 x 200 pixels, 16 available colors, 3D Sprite graphics), plus outstanding sound.

The 64's built-in music synthesizer has a programmable ADSR (attack, decay, sustain, release) envelope, 3 voices (each with a 9-octave range) and 4 waveforms. All of which you can hear through your audio system and see in full color as you compose or play back.

NOW'S YOUR CHANCE.

If you've been waiting for the "computer revolution," consider it as having arrived.

Through its 25 years of existence, Commodore has been committed to delivering better products and lower prices.

Today, the company's vertical integration has resulted in the Commodore 64's price performance breakthrough heralded by Shearson/American Express.

Visit a Commodore Computer dealer and discover the 64 soon.

It will expand your mind without deflating your wallet.

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

Commodore Business Machines/Personal Systems Division
P.O. Box 500 Conshohocken, Pennsylvania 19428

Please send me more information on the Commodore 64.™

Name_____

Company_____

Address_____

City_____ State_____

Zip_____ Phone_____

commodore COMPUTER

Line Finder

by
Elizabeth Deal

Here is a quickie routine that returns memory location of any line in a Basic program. It is written for Upgrade PET, with suggested adjustments for the 4.0 PETs. Checksums are provided.

The addresses of the 4.0 ROM routines are in the Basic loader, but are untested. I know they all work, as I have tried a similar program in a 4.0 system, but this very package is untested. (Editor's Note: 4.0 is OK)

If you type one set of data lines for your own system, as I suspect you will, omit line 504. It is there for your friendly editor, who just might not be lucky to have an Upgrade PET.

The source listing is for people who have assemblers. It uses standard MOS notation, though you may have to unpack the instructions onto separate lines. The code is completely relocateable. My loader places it in the tape buffer #2, users of 4.0 may need to put the loading address (AD) a bit higher than 826. ROM-routine addresses are from Butterfield's recovery maps.

For those who can't stand the thought of machine code I have included an old Basic line finder. You'll hear it grinding as I have not tried to optimize speed by selective placement. But still you'll get the results. No flags are shown, but a word "quit" means an in-between line number or end of program. It's good only for direct mode since it interrogates and prints.

What is Linefinder?
It's one of several routines I really miss not having built in. It can be used in direct mode and from within a Basic program. It returns a hex address as well as a decimal version. The address is that of the chain link, the actual Basic text begins at address + 4. The zero of the end of previous line is at address – 1 (minus one).

The idea of the peculiar syntax is from Brad Templeton, who brought you POWER (POWER is a registered trademark of Professional Software). We say:

```
SYS AD,GOTO1234
```

which locates line 1234 and displays results on the screen. We say:

```
SYS AD,GOTO1234,
```

with a comma at the end when we want to suppress the display as we may wish to do within a running program. Line location is also squirrelled away in the first tape buffer, low order byte first, followed by a status flag. The details are explained in the source program.

To get at the line's address within a program we can code:

```
LN=PEEK(823)+256*PEEK(824)
```

which is a sum of low byte and 256 times high byte. Once you know where a line is, you can come to grips with the details by looking at the MLM display via SYS4.

Who Cares?

Why are we doing all this? I can't think which of many reasons to list here. Linefinder seems to live in my PET at all times.

Curiosity about how it's all put together. Self modifying programs, where, horror of horrors, we poke Basic statements into an existing line, hopefully long enough. Checking if your idea where the program is the same as the PET's. What if it isn't?.

In dynamic keyboard type routines a forced GOTO target address can and should be a variable, immune to RENumbering. You can do it now.

You can ask for the end-of-program address without the monitor or formulas as in SYSAD,GOTO63999. SYSAD,GOTO0 tells you where Basic begins if you're lost. Please keep the number under 64000, don't count on PET giving a SYNTAX ERROR message at all times. You can use it in debugging machine code/Basic combinations, specifically problems of the nasty off-by-one variety. You can think up other reasons.

The cleanest example, my favorite, is self-illustrated at the beginning of the listing. Since I'm too lazy to provide a "both systems" loader, some of you may type the upgrade, others—the 4.0 version of the DATA lines.

Paul Fleming, the editor of this magazine has, I think, a Basic 4.0 PET. He must read in the second bunch of data items to check if what he prints really works, and retain the Upgrade lines for obvious reasons. That's why the silly line 504 exists. Here, everything pertaining to the Upgrade PET is bypassed, to provide 4.0 PET with the proper data.

Editor's Note: Having checked out the 4.0 version, I'm happy to report that it does work. In addition, the program can be entered as listed or you may leave out line 504 and, depending upon your ROM version (Upgrade or 4.0), enter the appropriate set of data statements:

Upgrade—lines 511-523
4.0—lines 525-537

Unless you have the Assembler by Brad Templeton, DO NOT enter any lines higher than 538

This nuisance is not too troublesome in a routine such as this. But under time constraints reading unneeded data may be undesireable. Had we a RESTORE 525 type statement, things would have been simpler.

If the Linefinder were in place, we could zoom into DATA we need in a jiffy: do SYSAD,GOTO525,: PEEK 823/824 and pack it into LN: POKE 62/63 with LN-1 split into low and high bytes, issue a READ and 4.0 system would read only what it is supposed to read.

Details

While using this inside a program, it is a good idea to check the status of the flag in 825. If it is zero, you will set the data pointer to, for instance, line 1000, which is perfectly correct, PET will find real data subsequent to 1000. If the flag is 1 (one) it means you have requested a line with an in-between number. You can quit, or use the next line as a target since that is the address you'll receive. In other applications you may wish to avoid doing anything unless the flag is zero. The choice is yours.

If you insist on writing self-modifying programs, you can use the information in this article to advantage. You can check if your destination line really exists, flag must be zero. You can check how much room you have on the target line by following the pointer chain. You can check if you've hit the end of a program as it is not very nice to start poking things into the table of variables.

In machine code, this flag can be checked by the status word; its flags will do the job. It may or may not be important, depending on what you plan to do with the information. The status of the flag is not important in direct mode, that's why I didn't bother printing it.

A Big Picture

In case you're wondering why the silly GOTO exists in the syntax, the reason is RENumbering. If you said sys ad,525 the 525 would stay 525 forever even if your new line range was in the thousands. Stick a GOTO before line number and the thing is deluxe. Yes, any RENumberable command will work in this scheme, but that would have to be coded in.

I thank Brad Templeton for suggesting the syntax, as the routine such as this is mostly useless to me when it couldn't be renumbered. Both the Palo Alto IC's TOOLKIT and Professional Software's POWER handle the syntax properly.

Incidentally, in direct mode, POWER users can have this available by a key-press, for instance M for machine code version, B for the Basic version. Just say:

```
10 REM"M=SYS826,GOTO
20 REM"B(backarrow)606
```

and you're in business. The Basic version will query you about line number. To call the machine code version, use M, PET does the typing, you only need to fill in the number and push RETURN, which isn't much trouble.

Big thanks to Jim Butterfield who has put in a big effort in trying to teach me some machine code, and congratulations to Brad Templeton whose PET-POWER-PAL invention made my PET into a dream machine.

```
500 REM------------------------------
501 REM LINE FINDER   ELIZABETH DEAL
502 REM------------------------------
503 AD=826:READ SP,CS
504 IFPEEK(50003)=160THENFORJ=0TOSP:
    READV:NEXTJ:READ SP,CS
505 FORJ=0TOSP:READV:POKEAD+J,V
506 S=S+V:NEXTJ
507 IFS<>CSTHENPRINTS"/"CS:STOP
508 PRINT"OK, LINEFINDER: 'SYS"AD",
    GOTO ##'":END
509 :
510 REM------------UPGRADE
511 DATA 89,9010
512 DATA 160,0,140,57,3,32,118,0
513 DATA 240,76,201,44,208,72,32,112
514 DATA 0,201,137,208,65,32,112,0
515 DATA 160,0,32,115,200,72,32,44
516 DATA 197,104,176,3,238,57,3,201
517 DATA 44,8,208,3,32,112,0,164
518 DATA 92,165,93,140,55,3,141,56
519 DATA 3,40,240,14,32,117,231,152
520 DATA 32,117,231,166,92,165,93,32
521 DATA 217,220,160,0,177,92,208,5
522 DATA 9,128,141,57,3,96,76,3
523 DATA 206,255
524 REM------------4.0
525 DATA 89,8913
526 DATA 160,0,140,57,3,32,118,0
527 DATA 240,76,201,44,208,72,32,112
528 DATA 0,201,137,208,65,32,112,0
529 DATA 160,0,32,246,184,72,32,163
530 DATA 181,104,176,3,238,57,3,201
531 DATA 44,8,208,3,32,112,0,164
532 DATA 92,165,93,140,55,3,141,56
533 DATA 3,40,240,14,32,34,215,152
534 DATA 32,34,215,166,92,165,93,32
535 DATA 131,207,160,0,177,92,208,5
536 DATA 9,128,141,57,3,96,76,0
537 DATA 191,255
538 REM------------------------------
539 SYS7*4096
540 .OPT OO
541 ;PAL SOURCE
542 ;
543 ;TO FIND LINE AND PRINT ADDRESS
544 ;  SYS THIS,GOTO544
545 ;TO SKIP PRINTING USE COMMA
546 ;  SYS THIS,GOTO546,
547 ;
548 ;ADDRESS IN LNAD (823/824 DEC)
549 ;  POINTS TO THE LINK BYTE ON LINE
550 ;FLAG IN LNAD+2  (825 DEC) MEANS
551 ;  $00=  0  EXACT LINE FOUND
552 ;  $01=  1  LINE# BETWEEN 2 LINES
553 ;           (ADDRESS=NEXT LINE)
554 ;  $80=128  LINE# > LAST LINE
555 ;           (ADDRESS=END OF TEXT)
556 ;
557 ;RELOCATEABLE ANY SENSIBLE PLACE
558 ;
559 *=$4000        ;CODE LENGTH $0-59
560 CHRGET =$0070
561 CHRGOT =$0076
562 BASLN  =$C52C; $B5A3 ;(UPGR;4)
563 FIXPT  =$C873; $B8F6
564 WROB   =$E775; $D722
565 PRFIX  =$DCD9; $CF83
566 SYNTAX =$CE03; $BF00
567 I1     =$5C
568 COMMA  =$2C
569 GOTO   =$89
570 LNAD   =$0337
571 FLAG   =LNAD+2
572 ;
573 ;    DO FOUND FLAG=OK
574 ↑     LDY #0:STY FLAG
575 ;    QUIT ON SYS CALL ALONE
576 ↑     JSR CHRGOT:BEQ QUIT
577 ;    QUIT ON NO ',GOTO'
578 ↑     CMP #COMMA:BNE QUIT
579 ↑     JSR CHRGET
580 ↑     CMP #GOTO:BNE QUIT
581 ;    ASCII TO INTEGER
582 ↑     JSR CHRGET
583 ↑     LDY #0:JSR FIXPT:PHA
584 ↑     JSR BASLN:PLA:BCS FOUND
585 ;    FLAG NONEXISTENT LINE
586 ↑     INC FLAG ;OVERSHOT LINE
587 FOUND CMP #COMMA:PHP:BNE FND1
588 ↑     JSR CHRGET
589 ;    SAVE LINE ADDRESS
590 FND1  LDY I1:LDA I1+1
591 ↑     STY LNAD:STA LNAD+1
592 ;    RETURN OR PRINT ADDRESS
593 ↑     PLP:BEQ DONE
594 ;    ADDRESS HEX (HHLL) AND DEC
595 ↑     JSR WROB:TYA:JSR WROB
596 ↑     LDX I1:LDA I1+1:JSR PRFIX
597 DONE  LDY #0:LDA (I1),Y:BNE DONE1
598 ↑     ORA #$80:STA FLAG ;HIT END
599 DONE1 RTS
600 QUIT  JMP SYNTAX
601 ↑     .END
602 END
603 REM------------------------------
604 REM SL00000000W BASIC VERSION
605 REM------------------------------
606 L=40:FF=1:CLOSEFF
607 M$(0)="QUIT  AT":M$(1)="FOUND AT"
608 DEFFNLL(L)=PEEK(L)+256*PEEK(L+1)
609 PRINT"L#:";
610 OPENFF,0:INPUT#FF,L$:CLOSEFF
611 IFLEN(L$)>5GOTO609
612 LF=INT(VAL(L$))
613 IFLF<00RLF>63999GOTO609
614 LA=FNLL(L):LE=0
615 J=LA-1:LC=J:IFPEEK(J+2)=0GOTO620
616 L=J+1:LA=FNLL(L):L=J+3
617 LT=FNLL(L):IFLT<LFGOTO615
618 IFLT=LFTHENLE=1
619 LC=LC+1:REM LINK ADDRESS,0 AT LC
620 PRINT"   "M$(LE)LC"$";
621 D=LC/4096:FORI=1TO4:D%=D
622 D=(D-D%)*16:IFD%>9THEND%=D%+7
623 PRINTCHR$(D%+48);:NEXT:RETURN
624 REM------------------------------
READY.
```

Introduction to the Machine Language Monitor

TIM is the Terminal Interface Monitor designed for the 65XX series of microprocessors. It allows the user to type commands on the PET/CBM keyboard which will start executing a program, display or modify registers and memory locations, and load or save binary data to diskette or tape. When modifying memory, TIM performs automatic read after write verification to insure that the addressed memory exists, is read/write type, and is responding correctly.

To enter the monitor, type SYS1024 and return. You should see a display similar to this:

```
B*
      PC   IRQ   SR AC XR YR SP
.;   0401 E455 32 04 5E 00 F8
```

When the microprocessor executed the instruction at location 1024 (a hex 0 or Break) it transfers control to TIM which displays the current contents of the Program Counter, the IRQ vector, the Status Register, the Accumulator, the X and Y registers and the Stack Pointer. The prompt character in TIM is the period. By typing an *r* at the prompt, you can display the current contents in hexadecimal of these registers whenever necessary.

One of the most useful commands is the memory display command. By specifiying a starting and ending address you will be able to examine and change a listing of sequential bytes anywhere in memory. The monitor requires addresses and values to be entered in hexadecimal and always displays in hexadecimal.

To examine the beginning of a basic program, for example, type an M, space, starting address, space, ending address, return.

.m 0400 0487

This will display the contents of 136 bytes of memory starting with hexadecimal address 400 and ending with hex 487. The bytes are displayed eight bytes per row and are each represented by 2-digit hex numbers ranging from 00 (decimal 0) to FF (decimal 255). The contents of these bytes are easily altered by cursoring up and overtyping a value on the screen. When the Return key is pressed, the values displayed on that line of the screen are entered into the corresponding memory locations. Try changing a few bytes and then type the *m* display command again. You will see that the changes you made are now the new values in memory. This is an easy way to enter a short machine language program if you "hand assemble" it because you don't have an assembler package. Try entering Dave Scott's program from the October issue. Type

.m 0500 0510

Now move the cursor back up and change the display to look like this.

```
.:  0500  a2  0d  20  d2  ff  a2  20  8a
.:  0508  20  d2  ff  e8  e0  60  d0  f7
.:  0510  00  aa  aa  aa  aa  aa  aa
```

To test the program now entered we need to use the *g* or "go" command which is similar to the "goto" command in basic. It tells the microprocessor to start executing the machine language instructions at the specified address. Type

.g 0500

The computer runs the program which displays all of the lower case characters in order and then breaks back to the monitor when it executes the 00 at location 0510. If the address is not specified, the address in the program counter is used. You can try this by typing the *r* command to display the registers and then changing the PC register to 0500. Now all that is necessary is to type the *g* and return. You will probably want to save your program onto disk or tape so you will need the *s* or "save" command next. To save this program onto disk drive 0 type

.s "0:testprogram",08,0500 ,0511

The *0* after the first quote specifies drive 0 and the *08* specifies the disk drive. For tape, leave out the *0:* and change *08* to *01*. The monitor does not save the last address specified so you must add one to the desired ending address, giving 0511 in this case. This will save all bytes from 0500 to 0510 inclusive.

To load the program back into memory you will need the **l** or "load" command.

Type

.l "0:testprogram",08

This will load testprogram from disk drive 0 back into the computer at the same memory addresses it was in when we saved it. To load from tape, leave out the *0:* and change *08* to *01*. You can test this by turning the computer off and then on again before loading testprogram. After loading, use the *m* command to make sure it loaded in addresses 0500 hex to 0510 hex. Use the *g* command to make sure that testprogram still runs.

The final command is the *x* or exit command. Simply enter an *x* at the prompt to go back to Basic. If you type *list* you will not see your program because it was not written in Basic. You can probably see now why the monitor is so useful for examining and editing small machine language programs. The monitor allows you to examine any and all memory locations in the PET/CBM. Using the memory maps that have been published in Commodore you can now start to use the monitor as a tool to learn how your computer works. C=

—Pat McAllister

PETSPEED® lets COMMODORE users do more...with less...with ease

Petspeed is not only going to change the way you think about your Pet Basic programs—it's going to change the way you use them.

It turns your painfully slow programs into fast, smooth-running, professional software.

Nothing can match the speed, compatibility and trouble-free performance of Petspeed.

Optimizes Any Program

While most compilers simply translate from one language to another, Petspeed analyzes your source program and eliminates unnecessary complexities. This speeds your processing time dramatically.

Programs are reduced to smaller components and reassembled into a far more efficient form.

For example, Petspeed:

- automatically uses faster integer arithmetic whenever possible
- automatically handles frequently occurring variables and arrays
- automatically calls all subroutines at maximum speed
- properly locates subroutines for optimum performance
- reduces the size of large programs so less memory is needed
- runs twice as fast as *any* other compiler

Totally Compatible

Petspeed will work with any combination of Commodore 4000 and 8000 series systems.

It will compile any Basic application.

And, Petspeed is already in daily use in major companies around the world, helping them work more efficiently and more profitably with their Commodore computers.

Easy-To-Use

You don't have to add compiler directives. Simply type in the program name, and in less than two minutes you'll see your program run significantly faster.

You'll see improvements no one can match.

Built-in Security

Petspeed is provided with a special security device, but you won't need a special runtime key for any compiled programs.

Instead, you can build in your own protection.

No one else can list your Petspeed code, so no one can tamper with your completed programs.

Your programs belong to you and you alone.

If you'd like to do more work in less time with your Commodore computer, contact your local Commodore dealer today. Or, call or write us for complete information and the name of the dealer nearest you.

SSE **(415) 964-8201**

SMALL SYSTEMS ENGINEERING
222 B View Street • Mountain View, CA 94041

Circle #28 on Reader Service Card

BASIC Programming on the COMMODORE 64

COMMODORE 64

By the time you read this, the Commodore 64 will be on your dealer's shelf—if he or she can keep it there! As you know by now, it is a spectacular breakthrough in personal programming, offering more computer at less cost than anything Commodore has ever produced. And, needless to say, it is even more of an advance over the competition.

Programming the 64 should be a snap for anyone with experience on PET or VIC. For the most part, our familiar BASIC 2.0 will work exactly as programming for the older machines and for the Commodore 64. As we learn more, we'll make sure that we feed the information to you. The choice of BASIC 2.0 instead of 4.0 was made with some soul-searching, not just at random. The typical user of a 64 is not expected to need the direct disk commands as much as other extensions, and the amount of memory to be committed to BASIC was to be limited. We chose to leave expansion space for color and sound extensions instead of the disk features. As a result, you will have to handle the disk in the more cumbersome manner of the "old days." And, of course, we will have to put up with noticeable garbage collection, but that doesn't appear to be quite as much of a problem as it was in the older machines.

Of all the PEEKs and POKEs for the PET, the only one we need to be concerned with on the Commodore 64 is 59468. Of course, VIC-style code for literals and graphics is the preferred route, but if you want to be interoperable on the 64 and the "Fat 40" then you may want to POKE 53272,21 along with POKE 59468,12 and POKE 53272,23 along with POKE 59468,14. Neither poke hurts any machine, and the combination will do to any machine what 59468 did to the old ones.

Operating system changes may be important to the BASIC programmer, although many will be transparent to you. Two items that are apparent are the fact that files no longer routinely close on file errors and the RESTORE key can have anomalous results, especially when sound and sprites are in use. The file change will be helpful in debugging—your error channel will still be open after, for example, a "file not open." We're trying to be more specific about the RESTORE phenomenon, and will let you know when we do.

As many programmers have already learned one can program for relative files under BASIC 2.0, although it requires use of the command channel. We find that opening a relative file of record length n requires:

OPEN lf,8,sa,"0:REL FILE NAME,L," + CHR$(n)

instead of DOPEN#lf,"REL FILE NAME",Ln. Locating a record and byte under 2.0 requires separating the record number into high and low bytes (HI = int(#/256), LO = #-256*HI). Then the command channel (#15) is used to:

PRINT#15,"P" + CHR$(SA + 96) + CHRS$(60) + CHR$(HI) + CHR-$(BYTE) where SA is the **secondary address** of the file you're using and BYTE is the byte number you want (optional for byte 1). For reasons I don't understand, this set of commands seems trickier than those of 4.0 in handling record pointers. I recommend that you point to the →

record you want explicitly before and after every operation, just to be sure.

That's a summary of everything we've learned so far that we needed to put substantial software onto the 64 in BASIC. While we still have a lot to learn, it's encouraging that most software has moved up to the new machine with nothing more than these few items of change; most of it doesn't even need that much. In most cases, a 4040 disk or a cassette of your favorite PET program will probably load and run perfectly on the 64 the first time. Then, add color, sound, sprites, and whatever else you desire to move up to a whole new world of operation. C=

—Michael Richter

COMMODORE 64

Conversion Tutor

by
Louis Sander

This program will teach you to make simple base conversions in your head. When you can do that, you'll find machine language much easier to deal with, and you'll be a better all-round computerist. Using the program gives one an easy and powerful demonstration of computer-aided instruction, using just over 2K of computer memory, and even less of the student's, to achieve a worthwhile result.

The program first asks you to choose one of the six possible conversions between hex, decimal, and binary. Once you make your choice, it drills you on converting small numbers (between zero and 15 decimal) between the two bases you have selected. If you make a mistake, the program shows you the right answer, and repeats the question later on. The drill continues until you have correctly made each conversion three times. Then it tells you how many errors you made, and how long you took to do the drill.

The program was written for a 40-column PET, but it should work with no changes on the 80-column machines, and can be easily modified to run on the VIC. It works, and it is useful—I've FINALLY learned to convert decimal into hex, myself!

```
10 REM ** BASE CONVERSION TUTOR **
20 REM
30 REM
40 REM
50 REM        LOUIS F. SANDER
60 REM        153 MAYER DRIVE
70 REM     PITTSBURGH, PA  15237
80 REM         (412)366-5414
90 REM
100 DATA  0,0000,$0
110 DATA  1,0001,$1
120 DATA  2,0010,$2
130 DATA  3,0011,$3
140 DATA  4,0100,$4
150 DATA  5,0101,$5
160 DATA  6,0110,$6
170 DATA  7,0111,$7
180 DATA  8,1000,$8
190 DATA  9,1001,$9
200 DATA 10,1010,$A
210 DATA 11,1011,$B
220 DATA 12,1100,$C
230 DATA 13,1101,$D
240 DATA 14,1110,$E
250 DATA 15,1111,$F
260 A$(1)="DEC":A$(2)="BIN":A$(3)="HEX":DIMD%(48)
270 PRINT"■■■■** WORKING **"
280 DIMN$(15,3):FORI=0TO15:FORJ=1TO3:READN$(I,J):NEXT:NEXT:REM ** D/B/H TABLE
290 I=0:FORJ=0TO15:FORK=1TO3:I=I+1:D%(I)=J:NEXT:NEXT:REM ** 'DECK' OF #'S
300 PRINT"■■■WHICH CONVERSION DO YOU WANT TO TRY?":A$="          ■"
310 PRINTA$"■1. DEC TO BIN"
320 PRINTA$"2. DEC TO HEX"
330 PRINTA$"3. BIN TO DEC"
340 PRINTA$"4. BIN TO HEX"
350 PRINTA$"5. HEX TO DEC"
360 PRINTA$"6. HEX TO BIN"
370 PRINT"■■       PRESS A NUMBER KEY..."
380 GETA$:IFA$<"1"ORA$>"6"THEN380
390 B1=1:B2=2:I=VAL(A$):IFI=2THENB2=3
400 IFI=3THENB1=2:B2=1
```

```
410 IFI=4THENB1=2:B2=3
420 IFI=5THENB1=3:B2=1
430 IFI=6THENB1=3
440 TI$="000000":N0=49
450 FORI=1TO48:N0=N0-1
460 X=INT(N0*RND(1)+1):Q=D%(X)
470 GOTO 590:REM ** QUERY ROUTINE
480 D%(X)=D%(N0)
490 NEXT
500 REM ** SCOREKEEPER
510 PRINT"■■■■■GOOD WORK!  YOU CONVERTED EVERY NUMBER"
520 PRINT"■CORRECTLY THREE TIMES.  IN DOING SO,"
530 PRINT"■YOU MADE"C-48"ERRORS."
540 PRINT"■■THE WHOLE EXERCISE TOOK YOU"INT(TI/60)"SECONDS."
550 PRINT"■YOU ■CAN■ DO BETTER!!!
560 INPUT"■■TRY AGAIN";A$
570 IFLEFT$(A$,1)="Y"THENC=0:GOTO63999
580 END
590 C=C+1:PRINT"■■■■■CONVERT THE NUMBER:■■■"
600 PRINTTAB(10)A$(B1),A$(B2)"■"
610 PRINTTAB(10)N$(0,B1)TAB(18);:INPUT"    ■■■";A$
620 IFA$=N$(0,B2)THENGOSUB670:GOTO480
630 PRINT"■■■■■■■■■■■■"TAB(18)"  ■"N$(0,B2);"■      "
640 FORJ=1TO2000:NEXT
650 GOTO460
660 REM ** CB2 SOUND
670 POKE59467,16:POKE59466,85:POKE59464,63+4*N0
680 FORJ=1TO20+N0/2:NEXT
690 POKE59467,0:POKE59466,0
700 RETURN

READY.
```

Recovering BASIC on the PET/CBM

by
Elizabeth Deal

Typing NEW permits us to begin work with an empty PET. Typing NEW when we didn't mean to forces us to retype if the program was not saved. Recovering from NEW is possible, however. If you have POWER (POWER is a registered trademark of Professional Software, Inc.) there is nothing to it, otherwise it may be tedious, but it can be done from the keyboard without loading or running any recovery programs.

The method is so simple it gives you a grand feeling of, for once, outsmarting the computer the easy way.

Understanding the Issue

Several facts are in order to understand how to fix things:

1. PET keeps the address of a program in $28/29 (dec 40/41). The address is usually $0401 (dec 1025), recovery logic being valid for any address.

2. PET keeps addresses of the beginning of variables, beginning of arrays and end of arrays in three pairs of pointers: $2A/2B, $2C/2D and $2E/2F.

3. A program line actually contains a two byte address of the next line, two byte line number, line text and a zero. Immediately after the zero is the next set of address values, the pointer in the previous line is the address of the first of those values.

4. Basic text area begins with a zero, which acts as an end of nonexistent line indicator. Basic text ends with two zeros in the address bytes shown in point 3.

5. We may, if we know what we're doing, change various pointers to suit our programming fancy. The toughest ones to change are the variables pointers: they cannot be changed in Basic using variables because as soon as we change one half of a pointer, the variables vanish into thin air. The second half gets a zero and the accidental NEWing suddenly looks mild compared to the results of messing the variables pointer.

When NEW is issued, the PET does two things:

6. It eliminates the continuity of a program by wiping out the first line's pointer to the second line. This shows up as two consecutive zeros in bytes $0401 and $0402 if that's where Basic lives. Of course, $0400 is always zero.

7. It sets the three pairs of variables pointers ($2A+) to the next available byte, $0403, and that is trouble, as any variables you use will overwrite the program.

8. Any action on our part that might cause SYNTAX ERROR will also cause the PET to overwrite the program. So the best advice I can give is LEAVE BASIC AS FAST AS YOU CAN, by typing SYS4, (MLM in POWER) correctly the first time, no keybounce, no skipping. As soon as that's done you're safe.

Recovery Procedure

Armed with this eight-point knowledge we can recover any program. You may even SAVE it if it is very important. (You can't save from Basic, because as far as Basic is concerned there is no program). Monitor save of a huge area is adviseable. It can be done by typing S"HUGE TEXT",08,0400,0400. Change 08 to 01 for tape #1, change $4000 (dec 16384) to a different value if you wish. This way the entire text is actually saved. Keep in mind it is useless in its present form. In the future, an ordinary Basic LOAD will work, and it will move the data pointers away from the program, but their values are irrelevant.

Recovery procedure consists of two steps:

STEP 1. requires nothing more than looking for a zero on the monitor display. The zero has to be an 'end of line' zero, not a zero in a line number. If you begin your search at $0405 you will not get confused, next available zero is the one we want. Go forward one more byte—next line begins here. Pick up the address and stuff it, low byte first, into $0401 and $0402 (see line 9023). Easy? A LIST command would now display the program, but it's still best not to leave the Monitor, and the program cannot be run yet anyway.

STEP 2. is tricky. We must find the end of the program by looking for three consecutive zeros and give this address to the three variables pointers. A machine code routine is, of course, a handy solution, but what if we don't have one, or what if we insist on recovery from the keyboard? We cannot set up a loop, its index is a variable, a variable will park itself smack at the beginning ($0403) of the very program we're trying to recover.

We could pretend the program is very long, by setting the three pairs of data pointers to a very high number or by Basic reLOADing what we saved. Variables could then be used, I don't know if it's safe. I also think that SAVEing/LOADing is a perfect nuisance, and programming to find the zeros is a bit difficult. One liners will not work and it's really best not to go to Basic yet.

How to do Step 2

In my opinion, there are three reasonable choices, two easy, one costly:

2.1 If the Supermon, Extramon or Micromon is in Pet, we're in luck. Hunt for three zeros:

H 0405 8000 00 00 00

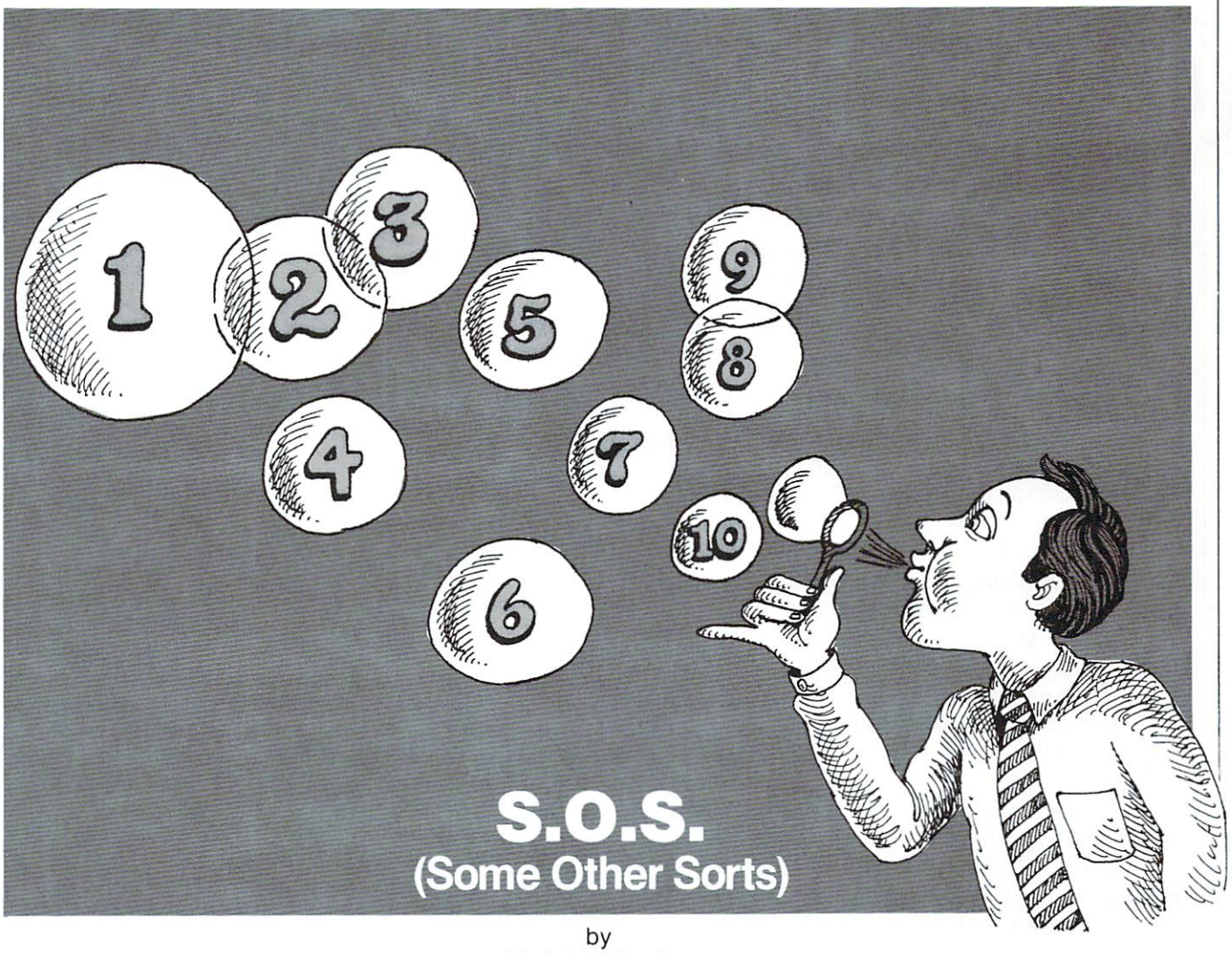
pick up the first address found on a hunt, add 3 (three) and stuff the value into the three pointers. To do this ask for display of memory: M 0028 0028. You will work on the three rightmost pairs. Suppose Hunt gave you $019. The address to use will be $0E1C, you can overtype 1C 0E three times. Push RETURN, exit the monitor via X and see the program work. This is a simple, and in my opinion, safe procedure.

2.2 Use POWER (tm, Professional Software). This is by far the fastest and the best method. POWER, theoretically, does not un-NEW, but all the works for the process exist in the FIX command. After doing step 1, which is easy, exit the monitor with X and type FIX. Unbelieveable, but that's all it takes.

This method is shown in the display attached to the article. It looks like a Basic program, but isn't. It's a screen, dump made easy to list automatically.

2.3 Find a neighborhood cash starving person under 12, and ask him to look for the three zeros by slowly displaying the entire memory from $0405 until he finds them. Pay what's due and follow the tail end of 2.1 instruction. C=

```
9000 REM--------------------------------
9001 REM RECOVER BASIC  ELIZABETH DEAL
9002 REM--------------------------------
9003 REM
9010 REM NEW
9011 REM
9012 REM READY.
9013 REM SYS4
9014 REM
9015 REM B*
9016 REM      PC   IRQ   SR AC XR YR SP
9017 REM .; 0005 9053 30 00 5E 04 F6
9018 REM .M 0400 0410
9019 REM .: 0400 00 00 00 0A 00 56 B2 31
9020 REM .: 0408 35 00 10 04 14 00 80 00
9021 REM .: 0410 00 00 AA AA AA AA AA AA
9022 REM .M 0400 0400
9023 REM .: 0400 00 0A 04 0A 00 56 B2 31
9024 REM .X
9025 REM READY.
9026 REM FIX
9027 REM (C) 1980 BRAD TEMPLETON
9028 REM READY.
9029 REM LIST
9030 REM
9031 REM 10 V=15
9032 REM 20 END
9033 REM
9034 REM READY.
9035 REM--------------------------------
READY.
```



by
Dwight Wheeler

It's been estimated (I don't remember by whom) that of all computer operations, about 25% of the time is spent on sorting. If my own experience is any indication, I would say that figure is probably accurate, or maybe too low. In any case, sorts are indispensable and the faster they run, the better.

You may already be familiar with the bubble sort, the earliest "exchange" sort. While easy to code, such a sort suffers from inefficiency. It will serve for small lists, but when the job gets tougher, we should look elsewhere. So, here's a slightly different sort which adjusts the lower boundary according to the status of the list to be sorted.

I call this the "BBSORT" which is short for "Boundary Bubble Sort." As with all bubble sorts, the heaviest items will sink to the bottom while the light ones rise to the top. What makes this sort different is that it checks the position in the list at which the *last exchange* is made and that point becomes the new lower limit of the next pass. Thus, if the list is in order at the start, no exchanges are made, the lower boundary becomes 0, and the sort stops after only one pass. Since most lists are usually in partial order at the outset, this procedure can mean a considerable saving in time.

Let's look at the code:

```
100 REM  --> PGM: BBSORT <--
```

Since this sort is done in internal memory, we must set up an array. A dimension (DIM) statement is necessary for lists greater than 10 items. The maximum of 100 here is arbitrary:

```
110 DIM A$(100)
120 REM.....LOAD ARRAY.....
```

Set the counter to 1:

```
130 N = 1
```

You could obtain your list for the array directly from the keyboard, or from data files on disc or cassette, or from DATA statements in the program, depending upon your application. For this illustration we'll use DATA statements as the source. DATA can be placed anywhere in the program, but I usually put it last so it can be expanded without conflict in line numbers. Let's read the DATA:

```
140 READ A$(N)
```

Check for the "end" of the data:

```
150 IF A$(N)="END"  THEN 180
```

Add one to the counter:

```
160 N = N + 1
```

Go back and read another data item:

```
170 GOTO 140
```

When all the data have been read, line 150 will detect it and jump down to the actual sort routine. But first we must subtract 1 from the number of items in the array to drop the "END" item which should not be included in the list.

```
180 N = N - 1
190 REM.....SORT.....
```

Here are the "working parts" of this sort. At the beginning, we set the lower boundary (B) at the number of items in the array (N) so that each entry will be looked at at least once.

```
200 B = N
```

Then we immediately check to see if there is anything in the list. If not, the procedure stops at once:

```
210 IF B = Ø THEN 33Ø
```

If B is not equal to zero, the program proceeds. At this point we must establish an intermediate variable as a position locator—call it J—and set it to the top of the list, position 0:

```
220 J = Ø
```

Now we step through the list from 1 to the end minus 1 with a FOR/NEXT loop using the index I:

```
230 FOR I = 1 TO B-1
```

Compare the first item [A$(I)] with the next [A$(I+1)].

```
240 IF A$(I) <= A$(I+1) THEN 290
```

If the first item is less than or equal to the second, it is skipped by jumping to the next I in line 290. If not, the two items are exchanged using a standard three-step exchange routine:

```
249 REM.....EXCHANGE.....
```

A temporary variable T$ is set equal to the first item:

```
250 T$ = A$(I)
```

Then the first item is assigned the value of the second (smaller) item:

```
260 A$(I) = A$(I+1)
```

The second item is then assigned the value of the temporary variable and the exchange is complete:

```
270 A$(I+1) = T$
```

So far so good, but here the plot thickens. Remember "J"? That's the position locator which keeps track of the number in the list where the exchanges are made. In this case, an exchange has been made at position I, so we take note of that:

```
280 J = I
```

Then we go to the next item:

```
290 NEXT I
```

The program goes back to line 230, the index I is incremented by one, and the next two items are compared, etc. Each time an exchange is made, J takes on the value of the index I at that point. When the whole list has been scanned, J will have the value of the index where the last exchange was made. If no exchanges were made, J will equal zero since it was initialized to zero in line 220.

When the FOR/NEXT loop has been completed (at B-1), control drops through to line 300 which sets the lower boundary B to the new value J (which was the point of the last exchange):

```
300 B = J
```

Then we hop back up to see if the new boundary is zero.

```
310 GOTO 210
```

If B is not equal to zero, the list is run through again—but only as far as the new lower boundary.

The loop is repeated until no exchanges are made J = 0 = B). At this point, line 210 will transfer control to the next segment of the program—in this case, the printout.

```
320 REM.....PRINT.....
330 FOR X = 1 TO N
340 PRINT A$(X);"  " ;
350 NEXT X
360 END
```

Finally, here is the long-lost DATA statement. As a handy "benchmark" for comparing sorts, I often use the sequence on the keyboard:

```
370 DATA Q,W,E,R,T,Y,U,I,O,P,A,S
    D,F,G,H,J,K,L,Z,X,C,V,B,N,M
```

Now, the moment of truth . . . run it, and count the number of seconds it takes to sort. Just for fun, change the DATA line to place all the letters in alphabetical order. Run it again . . . surprise? It prints almost immediately! So our "Boundary Bubble Sort" works and is an improvement on the average bubble sort.

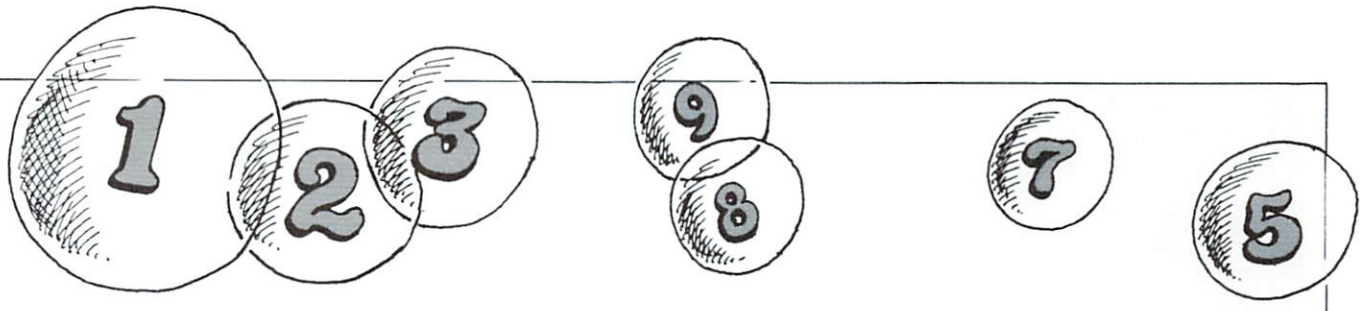
The loop is repeated until no exchanges are made and J=0=B. When B=0, line 210 will transfer control to the next segment of the program—in this case, the printout.

```
320 REM....PRINT....
330 FOR X = 1 TO N
340 PRINT A$(X);"  " ;
350 NEXT X
360 END
```

Finally, here is the long-lost DATA statement. As a handy "Benchmark" for comparing sorts, I often use the sequence found on the typewriter keyboard:

```
370 DATA Q,W,E,R,T,Y,U,I,O,P,A,S
D,F,G,H,J,K,L,Z,X,C,V,B,N,M,END
```

Finally . . . now's the time . . . run it, and count the seconds it takes to sort (about 8 seconds on an Apple II Plus).



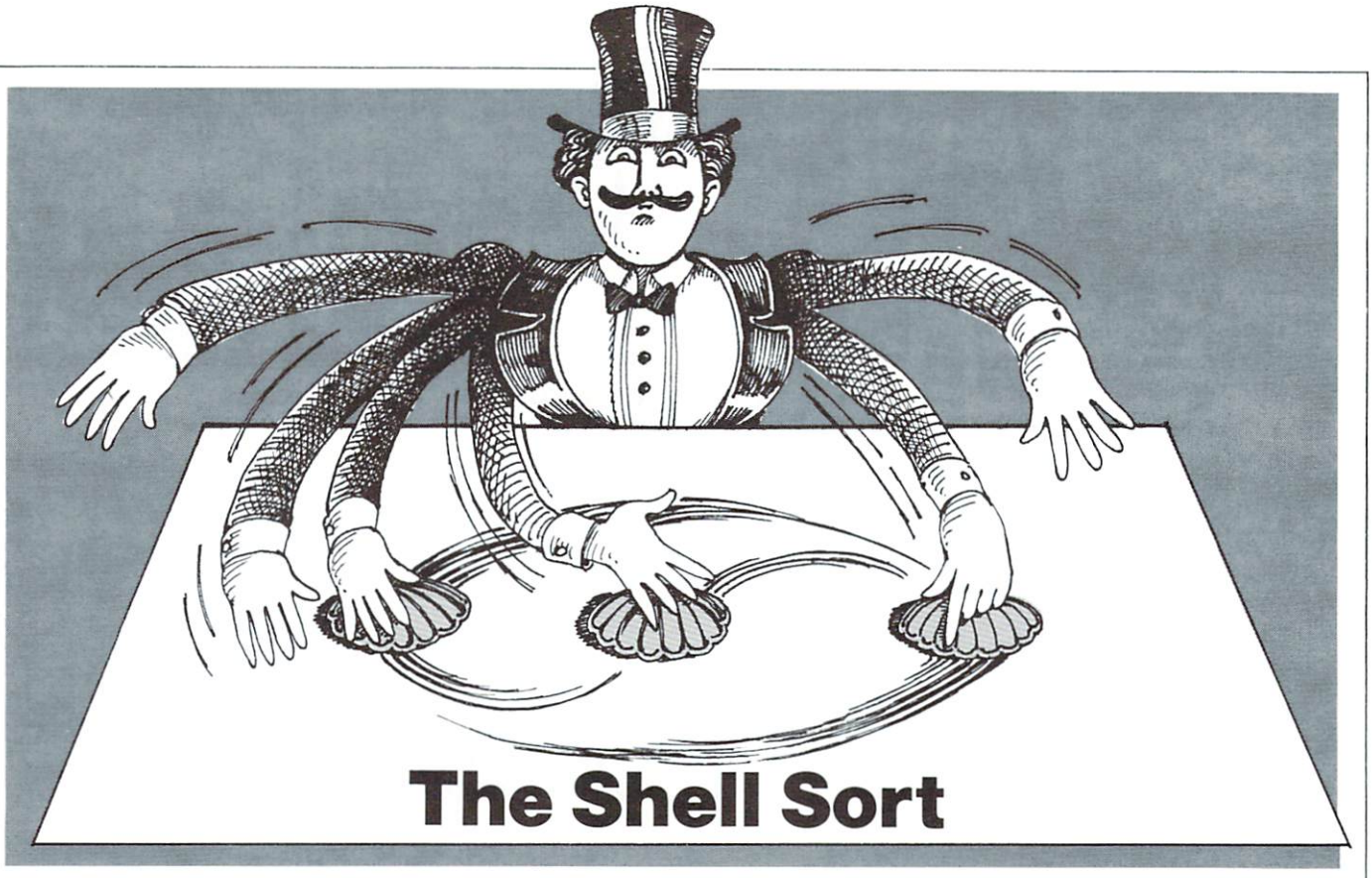
For kicks, change the DATA line to alphabetical order:

```
370 DATA A,B,C,D,E,F,G,H,I,J,K,L
    M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z
```

Now run it again . . . surprise? It prints almost immediately! So our "Boundary Bubble Sort" does work, and it is an improvement over a plain bubble sort. C=

COMING UP: S.O.S., Part II—"Shellsort"

```
100 REM  >> PGM: BBSORT <<
110 DIM A$(100)
120 REM ....LOAD ARRAY....
130 N=1
140 READ A$(N)
150 IF A$(N)="END" THEN 180
160 N=N+1
170 GOTO 140
180 N=N-1
190 REM ....SORT....
200 B=N
210 IF B=0 THEN 330
220 J=0
230 FOR I=1 TO B-1
240 IF A$(I) <= A$(I+1) THEN 290
249 REM ....EXCHANGE....
250 T$ = A$(I)
260 A$(I) = A$(I+1)
270 A$(I+1) = T$
280 J=I
290 NEXT I
300 B=J
310 GOTO 210
320 REM ....PRINT....
330 FOR X=1 TO N
340 PRINT A$(X);" ";
350 NEXT X
360 END
370 DATA Q,W,E,R,T,Y,U,I,O,P,A,S,
    D,F,G,H,J,K,L,Z,X,C,V,B,N,M,
    END
READY.
```



The Shell Sort

The computer world is indebted to Mr. D.L. Shell for this sort which appropriately bears his name. I don't know what language he used initially but using his theory I came up with this BASIC version. It should perform as prescribed on practically any computer.

While essentially another bubble sort, the Shell sort improves its efficiency by making some radical moves early in the game. Instead of comparing adjacent items in the list (as with a standard bubble sort), this sort compares items across a gap. This gap (call it G) starts out at 1/2 the length of the list. Therefore, in the first pass, members of the list can be moved from the middle all the way up to the top, or, from the end, up to the middle. This provides a very fast "rough sort" and accounts for the overall greater speed.

After all the exchanges are made that can be with G equal to 1/2 the length, the gap is halved again, and so on until the gap ends up at 1. The final passes (with a gap of 1) are actually a regular bubble sort.

Our demo program will use a data array for testing, so let's jump right in . . .

```
100 REM  --> PGM: 88SORT <--
110 DIM A$(100)(Dimension the array)
120 REM.....LOAD ARRAY.....
130 N = 1          (Set the counter)
140 READ A$(N)  Read the data
150 IF A$(N)="END"  THEN 180
(Check for end of data)
160 N = N + 1    (Increment the counter)
170 GOTO 140     (Go back for next data item)
180 N = N - 1    (Adjust array to exclude
"end" item)
```

Now the data has been read into the array with N items and we're ready to get into the actual sort. As we said, we're calling the gap G and to start it is set to 1/2 the total length of the list. In our test case, there are 26 items in the list so N = 26 and G = 13.

```
190 REM.....SORT.....
200 G = INT(N/2)
```

The integer function is used to clip all the fraction if N is an odd number. It's always a good idea to check right away to be sure that there are actually items in the list. Also, this line will be used later to signal the end of the sort.

```
210 IF G = 0 THEN 350
```

Time to step through the array list:

```
220 FOR I = 1 TO N-G
```

I will vary from 1 to (26 – 13) = 13. It is in the comparisons coming up that the Shell sort takes on its personality. The first item [A$(I)] is compared—*not* with the adjacent item, A$(I + 1)—but with A$(I + *G*) where *G* is the gap. In this case, I + G = 14. So, if the first item is less than or equal to the 14th item, nothing is done. But if it is greater, the two are exchanged:

```
230 IF A$(I) <= A$(I+G) THEN 290
240 REM.....EXCHANGE.....
250 T$ = A$(I)
260 A$(I) = A$(I+G)
270 A$(I+G) = T$
```

We must have some way to know when the list is ordered as well as it can be for each gap setting, so we establish a flag E which records and exchanges. When an exchange is made, E is set to the value 1.

```
280 E = 1
```

Then I is incremented and the next comparison is made using the same gap setting.

```
290 NEXT I
```

What happens during the comparisons in the first complete pass is illustrated in Figure 1. The comparisons

In our list, *Q* is the first item and *F* is the 14th and they are exchanged. The standard three-step routine is used

with the temporary variable T$:
which result in an exchange are indicated with an "x". You can see that some items have really moved up the list: *F, G, J, K,* and *L* are near the top while *Q, W, R, T,* and *Y* have been moved over halfway down the list. These are pretty good moves for just one pass and are proof that the "gap" comparisons accomplish their purpose. To see if exchanges have been made we check the flag:

```
300 IF E = 0 THEN 330
```

If exchanges *have* been made, E will equal 1 (line 280) and we must reset the flag to 0 so that we can check again after the next pass:

```
310 E = 0
```

Then back for another pass (using the same gap):

```
320 GOTO 210
```

If exchanges *have not* been made during the pass, E will equal 0 which will be detected in line 300. Control will pass to line 330 where we will cut the size of the gap in half. Once again, the integer function will chop off the remainder, if any: *(see footnote)*

```
330 G = INT(G/2)
```

And so back up to make another pass, but this time with the gap cut from 13 to 6:

```
340 GOTO 210
```

The passes will continue with a gap of six until no more exchanges can be made. The gap will then be cut to 3; more passes; then to 1 and a clean-up as a regular bubble sort.

When no more passes are made, E will equal zero, G will equal INT(1/2) which is zero and line 210 will pass the program to the output print routine.

```
350 REM.....PRINT.....
360 FOR X = 1 TO N
370 PRINT A$(X);"  " ;
380 NEXT X
390 END
```

Of course, we must have data:

```
400 DATA Q,W,E,R,T,Y,U,I,O,P,A,
    S,D,F,G,H,J,K,L,Z,X,C,V,B,N
    M,END
```

So you see that the Shell sort is not very difficult to code. To prove that it is worthwhile to use, try some comparison runs with other sorts and other benchmark data. The added speech can be signficant in long lists where every little bit helps. And that's our R_X for disorders. C=

Footnote:
For you Theoreticians: If the integer function were not used in line 330, the program would go into an infinite loop when the gap became less than 1. G = (G/2) would never reach zero and line 210 would never be true and the program would never stop. I know—because I fell into that trap on the very first trial run of this program!

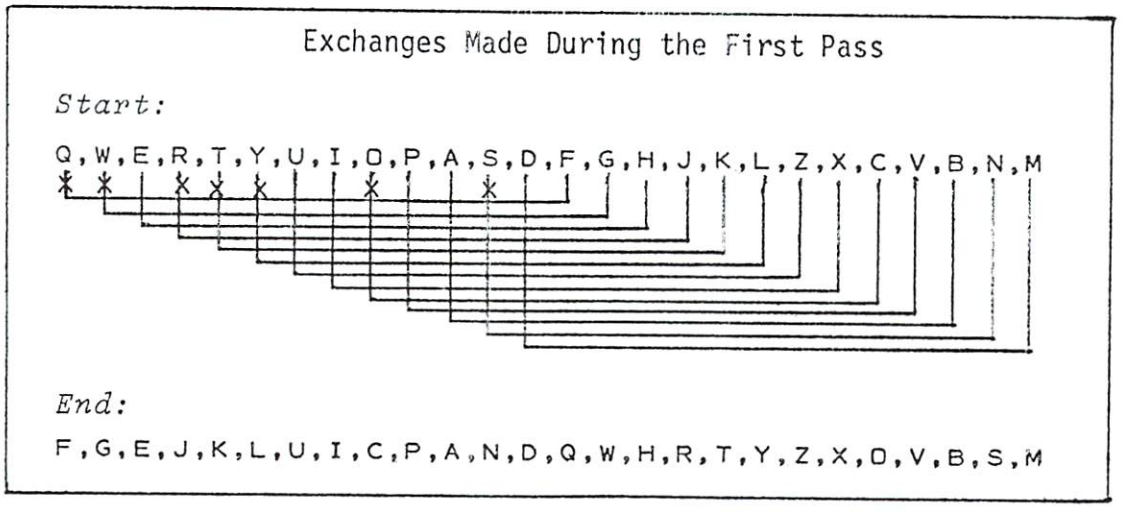


Figure 1.

```
100 REM   >> PGM: SHELL SORT <<
110 DIM A$(100)
120 REM ....LOAD ARRAY....
130 N=1
140 READ A$(N)
150 IF A$(N)= "END" THEN 180
160 N=N+1
170 GOTO 140
180 N=N-1
190 REM ....SORT....
200 G=N/2
210 IF G=0 THEN 350
220 FOR I=1 TO N-G
230 IF A$(I) <= A$(I+G) THEN 290
240 REM ....EXCHANGE....
250 T$ = A$(I)
260 A$(I) = A$(I+G)
270 A$(I+G) = T$
280 E=1
290 NEXT I
300 IF E=0 THEN 330
310 E=0
320 GOTO 210
330 G=INT(G/2)
340 GOTO 210
350 REM ....PRINT....
360 FOR X=1 TO N
370 PRINT A$(X);" ";
380 NEXT X
390 END
400 DATA Q,W,E,R,T,Y,U,I,O,P,A,S,D,F,G,H,J,K,L,Z,X,C,V,B,N,M,END
```

Commodore Disk Drive on a Old ROM PET

by
F. J. Townsend

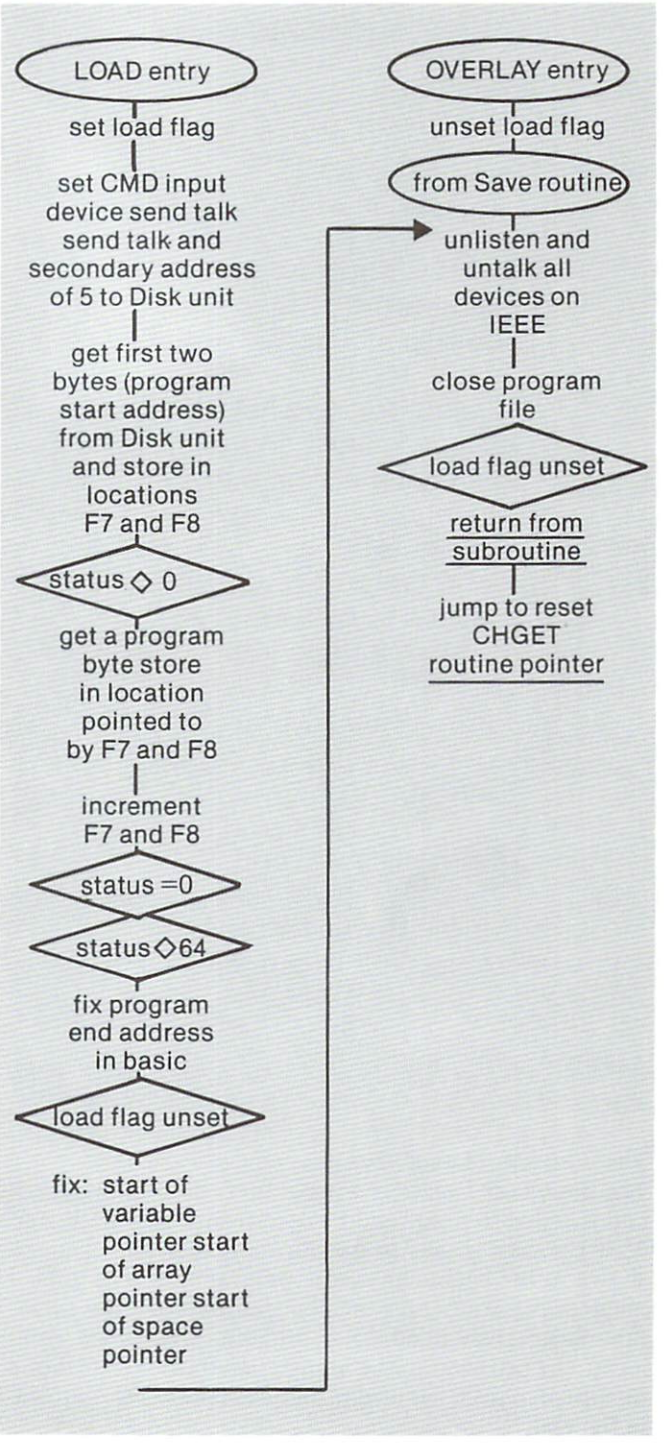
No doubt many owners of the original 2001 series PETs have hankered after the addition of a Disk unit to their system. In the past one way was to fit a set of later ROMs, but this is only easy if the version of the PET uses the 24 pin IC ROMs. The alternative was to examine the possibility of providing a machine code routine to interface to a Disk unit.

The main problem in driving a Disk unit with the Old ROM Basic is that the functions of Load, Save and Verify will not work with Disk. In their original form these functions were designed with the Cassette Tape only in view. As the method used for storing programs on Tape is different to Disk, later ROMs contained considerable changes in these routines. In the area of data handling, the old ROMs are able to Open, Input, Get, Print and Close Disk files. This now narrowed the missing requirements to Loading and Saving programs only, as it was decided that the Verify function was not absolutely necessary.

On the basis of this, I took the plunge and purchased a 8050 Disk unit. The choice of an 8050 was made so that all the latest DOS 2.5 facilities would be available as well as the increased storage capacity. The main point about the Commodore Disk unit is that programs are stored in the same format as data files, the main difference being that the file type is 'Program.' As the Old ROM PET can read Sequential files there was no reason why it could not read Program files so all that was needed was a routine able to open a program file, read it into the RAM store, adjust the various Basic pointers and then return to Basic. With the above in mind and a further constraint that the routine to be produced must fit in a Cassette Buffer, the following program was written.

The Disk Load Routine

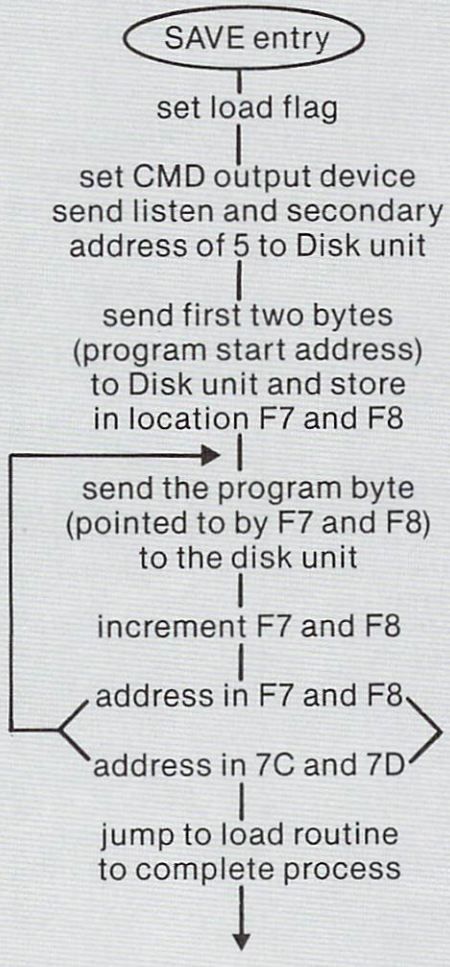
Figure 1 shows the flowchart of the program load routine. Two points will be noticed, these being that there is no mention of opening the program file and there are two entry points. In the first case opening files at machine code level is complicated and can consume considerable code so it was decided to open the required program file as a direct Basic command prior to entering the program load routine. This is simply done by typing:— OPEN 5, 8, 0 "program-name,P,R". The two entry points allow for either just loading a program and then returning to Basic command level or loading and immediately running a program in 'Overlay Mode' retaining the data area from the previous program.



The Disk Save Routine

Figure 2 shows the flowchart of the program save routine. As in the load routine the program file opening is done prior to entering the program save routine. This is done by typing:— OPEN 5, 8, 1, "program-name,P,W". The jump to the load routine on completion of save is purely to reduce code by using a common terminating routine.

Figure 2—The Save Routine



The Basic Disk Load/Save Program

The Basic Disk Load/Save program is shown in figure 3 and is coded to reside and be used from Cassette Buffer 2. If the user wishes to use Buffer 1 then the changes shown at the beginning of the program must be implemented. Naturally afer the changes are done the resultant Basic Disk Load/Save program will, when run, locate the program in the appropriate buffer. For those interested, a disassembled print of the program is shown in figure 4.

```
BASIC DISK LOAD/SAVE PROGRAM

 60 REM   CHANGES FOR BUFFERS NO 1 OR NO 2
 61 REM
 62 REM   BASIC     VALUES FOR BUFFERS
 63 REM   LINES      NO 1       NO 2
 64 REM
 65 REM   110        223,2      159,3
 66 REM   150        223,2      159,3
 67 REM   165        223,2      159,3
 68 REM   173        223,2      159,3
 69 REM
 70 REM   ALSO FOR BUFFER NO 1
 71 REM   CHANGES LINE 100 TO :=
 72 REM
 73 REM   FOR I = 634 TO 784
 74 REM
100 FORI=826TO976
102 READDT:POKEI,DT:NEXT:PRINTCHR$(147):NEW
105 DATA 169,1,208,2,169,0,141
110 DATA 159,3
115 DATA 162,5,32,139,247,32,204
120 DATA 241,133,247,173,12,2,208,59,32,204
    ,241,133,248,173,12,2
130 DATA 208,49,160,0,32,204,241,145,247,230
    ,247,208,2,230,248,173
140 DATA 12,2,240,240,201,64,208,27,165,247,
    133,229,165,248,133,230,172
150 DATA 159,3
155 DATA 240,14,133,125,133,127,133,129,165,
    247,133,124,133
160 DATA 126,133,128,32,125,242,169,5,32,205,242,172
165 DATA 159,3
170 DATA 240,1,96,76,154,197,234,1,162,5,142
173 DATA 159,3
175 DATA 32,220,247,165,122
180 DATA 133,247,32,48,242,165,123,133,248,32,
    48,242,160,0,177,247
190 DATA 32,48,242,230,247,208,2,230,248,165,
    248,197,125,208,239,165
200 DATA 247,197,124,208,233,240,188
```

```
 ADDRESS          MACHINE          ASSEMBLER
DEC    HEX        CODE               CODE
=======================================
826    33A        A9 01            LDA    #$01
828    33C        D0 02            BNE    $340
830    33E        A9 00            LDA    #$00
832    340        8D 9F 03         STA    $039F
835    343        A2 05            LDX    #$05
837    345        20 8B F7         JSR    $F78B
840    348        20 CC F1         JSR    $F1CC
843    34B        85 F7            STA    $F7
845    34D        AD 0C 02         LDA    $020C
848    350        D0 3B            BNE    $38D
850    352        20 CC F1         JSR    $F1CC
853    355        85 F8            STA    $F8
855    357        AD 0C 02         LDA    $020C
858    35A        D0 31            BNE    $38D
860    35C        A0 00            LDY    #$00
862    35E        20 CC F1         JSR    $F1CC
865    361        91 F7            STA    ($F7),Y
867    363        E6 F7            INC    $F7
869    365        D0 02            BNE    $369
871    367        E6 F8            INC    $F8
873    369        AD 0C 02         LDA    $020C
876    36C        F0 F0            BEQ    $35E
878    36E        C9 40            CMP    #$40
880    370        D0 1B            BNE    $38D
882    372        A5 F7            LDA    $F7
```

```
884   374    85 E5        STA   $E5
886   376    A5 F8        LDA   $F8
888   378    85 E6        STA   $E6
890   37A    AC 9F 03     LDY   $039F
893   37D    F0 0E        BEQ   $38D
895   37F    85 7D        STA   $7D
897   381    85 7F        STA   $7F
899   383    85 81        STA   $81
901   385    A5 F7        LDA   $F7
903   387    85 7C        STA   $7C
905   389    85 7E        STA   $7E
907   38B    85 80        STA   $80
909   38D    20 7D F2     JSR   $F27D
912   390    A9 05        LDA   #$05
914   392    20 CD F2     JSR   $F2CD
917   395    AC 9F 03     LDY   $039F
920   398    F0 01        BEQ   $39B
922   39A    60           RTS
923   39B    4C 9A C5     JMP   $C59A
926   39E    EA           NOP
927   39F    EA           NOP
928   3A0    A2 05        LDX   #$05
930   3A2    8E 9F 03     STX   $039F
933   3A5    20 DC F7     JSR   $F7DC
936   3A8    A5 7A        LDA   $7A
938   3AA    85 F7        STA   $F7
940   3AC    20 30 F2     JSR   $F230
943   3AF    A5 7B        LDA   $7B
945   3B1    85 F8        STA   $F8
947   3B3    20 30 F2     JSR   $F230
950   3B6    A0 00        LDY   #$00
952   3B8    B1 F7        LDA   ($F7),Y
954   3BA    20 30 F2     JSR   $F230
957   3BD    E6 F7        INC   $F7
959   3BF    D0 02        BNE   $3C3
961   3C1    E6 F8        INC   $F8
963   3C3    A5 F8        LDA   $F8
965   3C5    C5 7D        CMP   $7D
967   3C7    D0 EF        BNE   $3B8
969   3C9    A5 F7        LDA   $F7
971   3CB    C5 7C        CMP   $7C
973   3CD    D0 E9        BNE   $3B8
975   3CF    F0 BC        BEQ   $38D
```

Using the Disk Load/Save Program

When the PET is first switched on the Basic Disk Load/Save Program is read in from cassette and run. From now on the program will remain in the appropriate buffer until the PET is switched off or the buffer is overwritten.

Load a Disk program from Basic command level

Type OPEN 5,8,0, "program-name,P,R"
SYS(826)

The machine should return with the flashing cursor prompt on completion of the load!

Loading an over lay Disk program from within a program

Including the following code at the end of the calling program.

```
line-no OPEN 5,8,0, "overlay-name,P,R"
line-no REM INTERROGATE THE ERROR
        CHANNEL
line-no INPUT#15,AA,BB$,CC,DD
line-no IF AA<>0 THEN PRINT AA;BB$;CC;DD:
        stop;end
line-no SYS(830):END
```

After loading the overlay program it will then be automatically entered.

Saving a program to Disk

Type OPEN 5,8,1, "program-name,P,W"
SYS(928)

The machine should return with the flashing cursor prompt on completion of the save.
Note:—The secondary address is 1 for save and 0 for load.

The file mode is W (write) for save and R (read) for load.

The logical file number of 5 must only be used.

The entry points shown in the load and save examples are for use when the Disk Load/Save program resides in Buffer 2. The following table gives the entry points when using Buffer 1.

Entry Points

Function	Buffer 2	Buffer 1
Normal load	826	634
Overlay load	830	638
Save	928	736

Possible causes of failure when using the Disk Load/Save Program

The Open command was not typed exactly as in the instructions.

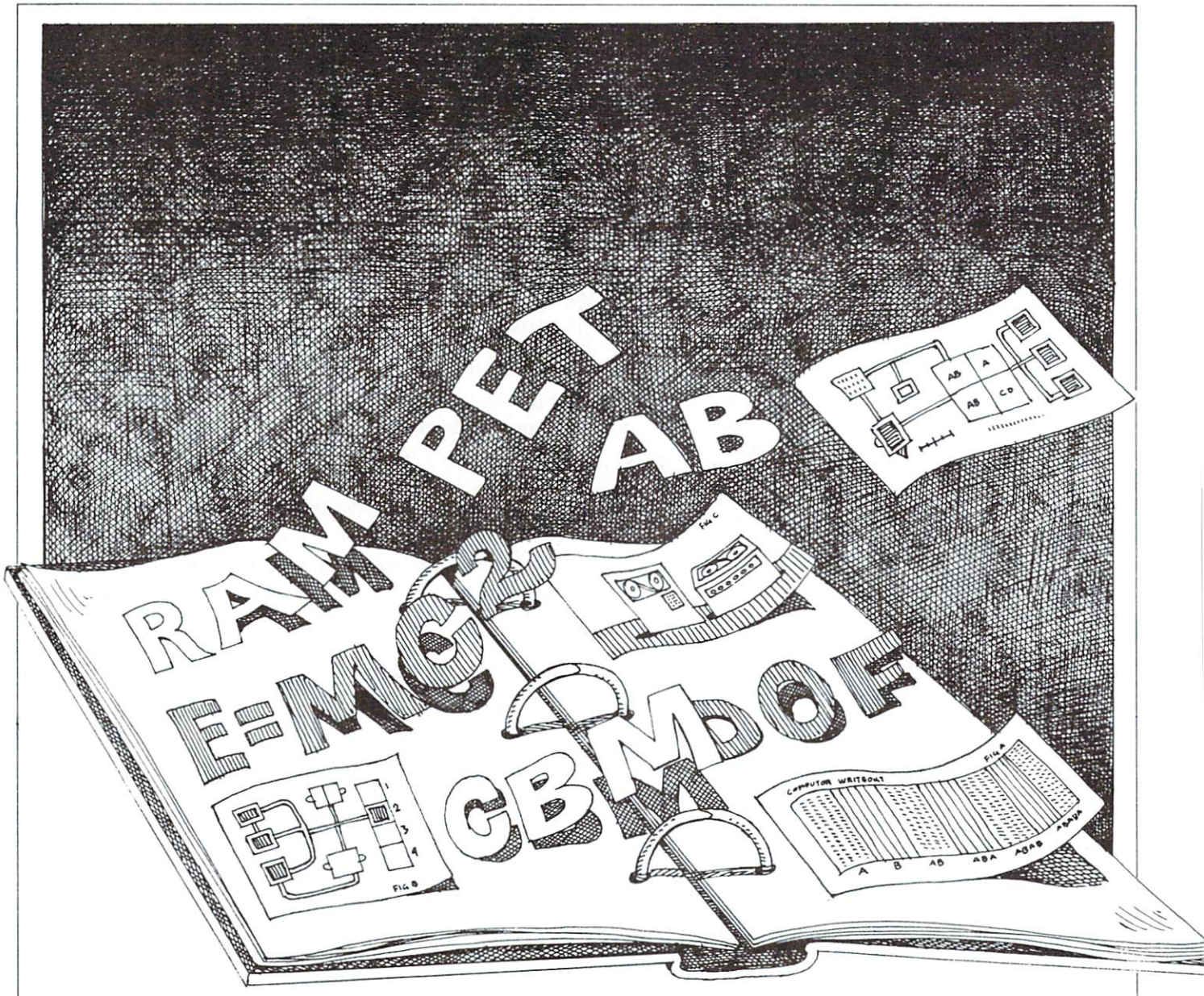
The 'program-name' program does not exist when loading or already exists when saving, on the current disks mounted on the Disk drive. A file with the logical file number of 5 is already open. In this case close it with: = CLOSE 5. Note: When using this system, logical file number 5 should be reserved for program loading and saving only.

If while loading a program the VERIFYING message appears on the screen, ignore it as it appears because the last cassette function was verify and in fact a load will be taking place.

Future Enhancements

Having produced a routine enabling a Disk unit to be used with the Old ROM PET, the next step is to be able to use the standard DOS Support program in conjunction with this routine. In a forthcoming article I shall explain how I produced an Old ROM version of the Commodore DOS Support Program. C=

Reprinted with permission from the June, 1982 edition of ***Computing International*** *Magazine.*



Excerpts from a Technical Notebook

Bits & Pieces

(Reprinted from the Transactor)

***Faster Than A Speeding Cathode Ray!**

```
10 PRINT"[CLR DN DN]IS NOT YOUR NAME
   ABE LINCOLN?":GOTO10
```

The string in the above statement prints and clears so fast that the screen can't keep up! You might expect the text to 'flash' on and off. But, as the trace is scanning the screen, the text actually prints, clears, and prints again before the trace gets a chance to erase. It's hard to say how many, but BASIC prints and clears several times during a single screen scan. Therefore, the text appears to be stationary, as if the Clear Screen character was not even there!

Then, they become 'un-synchronized.' At this point, the text appears to be drawn slowly across the line. The trace draws part of the text and then it's turned off again by the Clear Screen. The same thing happens next time around only a little farther to the left or right. It's rather hard to explain but not hard to imagine when you're looking at it.

Try different combinations by adding or removing CLRs, DNs, characters, commas and semicolons. For an interesting effect, add line 20 by simply duplicating line 10 (remove the GOTO 10 and add it at the end of line 20). Try this one too:

```
10 PRINT"[CLR 6DN]IS YOUR NAME ABE
   LINCOLN?";:GOTO10
```

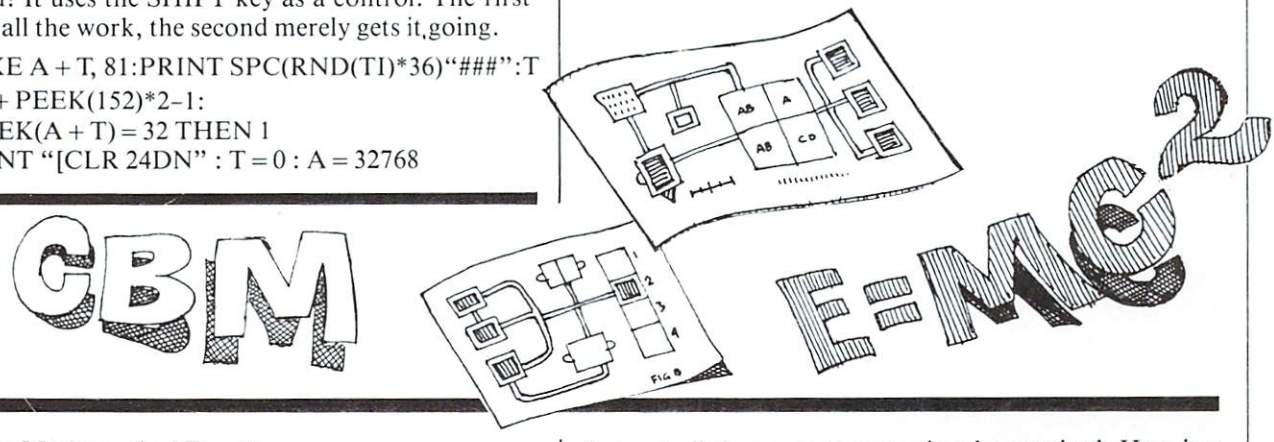
6DN = 6 cursor downs. Different machines produce different results. These were done on forty column PETs. 80 column users will have to modify the statements slightly to get the right effect since the scan speed is somewhat different.

Richard also has a one-line game which surely could be expanded! It uses the SHIFT key as a control. The first line does all the work, the second merely gets it going.

```
1 POKE A+T, 81:PRINT SPC(RND(TI)*36)"###":T
  =T+PEEK(152)*2-1:
IF PEEK(A+T)=32 THEN 1
2 PRINT "[CLR 24DN" : T=0 : A=32768
```

***More One-Liners**

```
1 FOR X=0 TO 999 : POKE 32768+X,
      (PEEK(32768+X)+128)AND255:NEXT
1 a=32768:i=0:j=38
2 S=SGN(J-I) : FOR X=I TO J STEP S : POKE A
      +X, 32 : POKE A+X+S, 87 :
NEXTX : I=39-I : J=39-J : GOTO 2
```



***Deriving Mathematical Functions**

BASIC has some trigonometric functions implemented but not all that may at some time be required. Here is a handy list:

Secant	SEC(X) = 1/COS(X)
Cosecant	CSC(X) = 1/SIN(X)
Cotangent	COT(X) = 1/TAN(X)
Inverse Sine	ARCSIN(X) = ATN(X/SQR(−X*X+1))
Inverse Cosine	ARCCOS(X) = −ATN(X/SQR(−X*X+1))+π/2
Inverse Secant	ARCSEC(X) = ATN(X/SQR(X*X−1))
Inverse Cosecant	ARCCSC(X) = ATN(X/SQR(X*X−1))+(SGN(X)−1*π/2
Inverse Cotangent	ARCCOT(X) = ATN(X)+π/2
Hyperbolic Sine	SINH(X) = (EXP(X)−EXP(−X))/2
Hyperbolic Cosine	COSH(X) = (EXP(X)+EXP(−X))/2
Hyperbolic Tangent	TANH(X) = EXP(−X)/(EXP(X)+EXP(−X))*2−1
Hyperbolic Secant	SECH(X) = 2/(EXP(X)−EXP(−X)
Hyperbolic Cosecant	CSCH(X) = 2/(EXP(X)−EXP(−X)
Hyperbolic Cotangent	COTH(X) = EXP(−X)/(EXP(X)−EXP(−X))*2+1
Inverse Hyperbolic Sine	ARCSINH(X) = LOG(X+SQR(X*X+1))
Inverse Hyperbolic Cosine	ARCCOSH(X) = LOG(X+SQR(X*X−1))
Inverse Hyperbolic Tangent	ARCTANH(X) = LOG((1+X)/(1−X))/2
Inverse Hyperbolic Secant	ARCSECH(X) = LOG((SQR(−X*X+1)+1/X))
Inverse Hyperbolic Cosecant	ARCCSCH(X) = LOG((SGN(X)*SQR(X*X+1/X))
Inverse Hyperbolic Cotangent	ARCCOTH(X) = LOG((X+1)/(X−1))/2

EXCERPTS FROM A TECHNICAL NOTEBOOK

VIC Notes

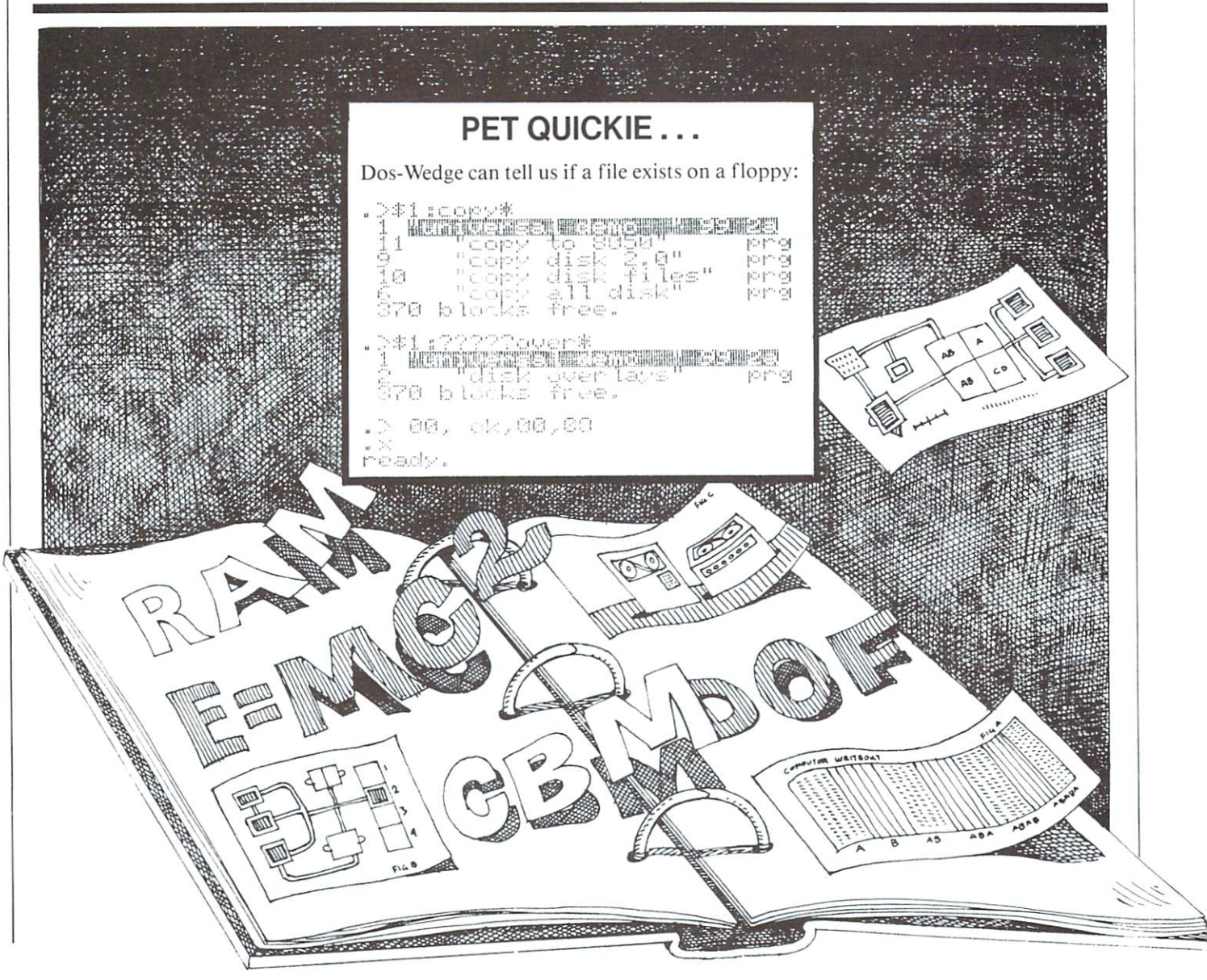
Vic-20 owners that wish to connect to color monitors will need some extra cables. You could wire them up yourselves; for pin connections, see the VIC 20 Programmers Reference Guide, pp. 282. Remember, the connectors shown here are as you look at them from the back of the VIC. The corresponding pins on the *jacks* will be "mirror image" looking into the jack.

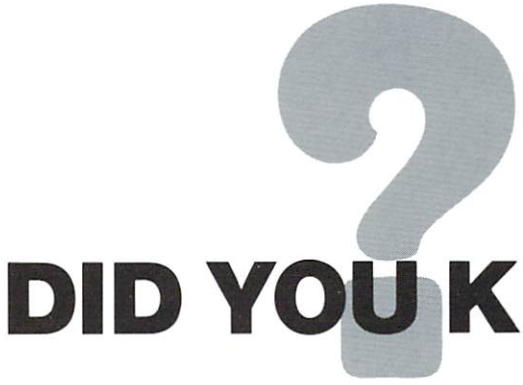
You can buy the necessary cables at any Radio Shack. First you'll need a "Y" Adapter (Part #42-2394); and a 5-pin European plug to 4 phono jacks. This one goes on the audio/video connector. It's not very long, so you'll also need the 1.8m shielded stereo cable (42-2352). Most color monitors use a female BNC connector for video in. In this case you'll need the ARCHER female RCA phono to male BNC adapter (278-254).

"Y" Adapter Connections

Black — Video low
Grey — Audio
White — Video High
Red — +5V Regulated @10 mA. max.

The 1.8m extension has black and grey connectors at each end. Use black for video and grey for audio.





DID YOU KNOW

Here is a little tidbit that you may find interesting. Did you ever wish that you could 'hide' parts of a program, while leaving it all in BASIC and copyable? Well here is a simple and effective way to do just that!

Simply load your program into memory, and place five colons (:::::) in front of every statement that you wish to hide, after the line number and before the BASIC line itself. Then just key in the following statements and execute them via a 'goto.' List the program and you will see that your secret code has vanished! To make the lines reappear, just 'goto' the line number you assigned to the second half of the statements below.

Try this little trick on the small program below just to see it work, it's fun and only costs you a little more space for the five colons and the extra statements.

```
10 rem "simple program"
20 print"this line will not be protected"
30 print"this line will not be protected either"
40 :::::print"but this line will. . . ."
50 :::::print"and this one too. . . ."
55 stop: rem "assume that this is the logical end of the program
60 rem "the coding from 60000-60020 will make the lines vanish
70 rem "the coding from 60030-60050 will make the lines reappear
60000 :::::fori = 1024 to 15000
60010 :::::ifpeek(i) = 58andpeek(i + 2) = thenpokei,0:i = i + 5
60020 :::::next:stop
60030 :::::fori = 1024 to 15000
60040 :::::ifpeek(i) = 0andpeek(i + 1) = 58andpeek(i + 2) = 58thenpokei,58:i = 1 + 5
60050 :::::next:stop
```

PLEASE NOTE . . . the value of 15000 in lines 60000 and 60030 just needs to be any value large enough so that it is larger than the number of bytes of your code to ensure that the search for the five colons continues through your entire program.

Have any interesting little programs, tricks, etc. for Commodore micros? Let us know and we'll feature them in Commodore Magazine. Send your suggestions to:

"Did You Know"
Commodore Magazine
487 Devon Park Drive
Wayne, PA 19087

Programming The PET/CBM

by Raeto Collin West

The book described by Jim Butterfield as

"...unquestionably the most comprehensive and accurate reference I have seen to date..."

The Reference Encyclopedia for Commodore 2000, 3000, 4000, and 8000 series computers and peripherals.

Here's just a sample of reviewer and reader reaction:

From reviewers:

Educational Computing Review by Stephen Potts

"Of all the books I have read on the PET this book *Programming the PET/CBM* by Raeto West must rank as one of the most comprehensive and readable accounts on the PET that I have ever had the pleasure to see...

"If you wish to get more from your PET than arcade games and simple teaching programs then this book is a must for your bookshelf. It does not matter whether you run on BASIC 1, BASIC 2, or BASIC 4 since all routines are supplied with addresses and changes to make them run on any machines wherever possible...

"...this book, with its lucid explanations of the PET, its useful routines and programming hints, is an essential purchase."

IPUG Magazine Review (British PET User Group) by Ron Geere

"This publication represents over a year's intensive research ... and the resulting product is a valuable work of reference. A tremendous amount of useful information has been packed in this 500+ page work at which I was so over-awed that I did not know how to start this review at first...

"This book is a must for every CBM/PET user."

From readers:

"...a book the average to advanced user cannot afford *not* to possess..."

"My copy of your *'Programming the PET/CBM* has been in daily use for nearly a month and I am finding it totally addictive, suffering severe withdrawal symptoms whenever I try half-heartedly to move on to other reading matter. It is without doubt the best book on its subject available today..."

"I have recently acquired a copy of your book *Programming the PET/CBM* and must congratulate you on its concept and on packing in so much detail. It's so very much better than anything I have had up to now that it'll be my constant reference manual."

"I have received my copy of *Programming the PET/CBM* by Raeto West and I have recommended it to several of my students. This book is so valuable that I cannot now afford to be without it."

Published exclusively in North America by **COMPUTE! Books**. The book is an astonishing reference manual of useful information. Contents include this and much more:

1 Introduction and overview: Plan of the book, sources of information, features and chronology of CBM hardware.

2 BASIC and how it works: Storage of BASIC and its variables; tokens, pointers, syntax; optimising BASIC.

3 Program and system design: Capabilities of the equipment; charts, algorithms, space, timing.

4 Effective programming in BASIC: Seventeen examples, including subroutines, dates, DATA, INPUT, rounding.

5 Alphabetic reference to BASIC keywords: Full descriptions, with examples, of all keywords, with methods for adding additional commands not present in CBM BASIC, e.g. AUTO, DEL, OLD, POP, PRINT USING, SORT, VARPTR.

6 Disk drives: Descriptions of operation and workings of disk drives, with BASIC and machine-code examples; bugs.

7 Alphabetic reference to disk BASIC commands: BASIC 4 disk commands with examples and notes.

8 Other peripherals and hardware: Tape storage and handling; printers; modem; keyboard; EPROMs; reset switches.

9 Graphics and sound: Tables of CBM characters; CRT chip; animation, bar plots, 80 by 50 etc.; user-port sound.

10 The transition to machine-code: Introductory concepts; a BASIC monitor; use of MLM, Supermon, Extramon; easy examples.

11 More 6502 machine-code: 6502 hardware features; eighteen common problems in programming; debugging.

12 Alphabetic reference to 6502 opcodes: Examples, notes, and explanations on each opcode from ADC to TYA.

13 Using ROM routines: IRQ, NMI, RESET; the Kernel; examples – modifying LIST; ordinary and relocating loaders.

14 Effective 6502 programming: Assemblers; CHRGET and wedges; PIAs, VIA, IEEE; common mistakes.

15 Index to BASIC ROMs and RAM: Memory map of RAM and ROM, detailing and comparing BASICs 1, 2, and 4.

16 Mathematical programming: Precision; equations; statistics; simulation; finance; calculus; machine-code.

17 Programming in business and education: Examples, applications and pitfalls in business and education.

Appendices: 6502 reference charts; Supermon listings; ASCII; glossary

Plus many programs, diagrams and charts. Paperback, 504 pages. ISBN 0 942386 04 3. **$24.95.**

To Order

Programming The PET/CBM

Call

TOLL FREE 800-334-0868

In NC Call 919-275-9809

Or send coupon to

COMPUTE! Books, P.O. Box 5406, Greensboro, NC 27403

In England, order from Level Limited, P.O. Box 438, Hampstead, London, NW3 1BH. Price in England is £14.90, including P & P.

Please send _____ copy (copies) of **Programming The PET/CBM** at $24.95 each. (In the US and Canada, add $3.00 shipping and handling. Outside North America add $9.00 for air mail delivery, $3.00 for surface delivery).

All orders must be prepaid in US funds (money order, check, or charge).

☐ Payment Enclosed

Please charge my ☐ VISA ☐ MasterCard ☐ Am. Express

Account No. _____ Expires ___/___

Name _____

Address _____

City _____ State _____ Zip _____

Country _____

Allow 4-6 weeks for delivery. Foreign surface delivery allow 2-4 months.

Silicon Office

by
Mike Heck

In creating Silicon Office, Bristol Software Factory has made good use of the expanded memory available on Commodore CBM 8032 systems with the 64K upgrade. This application is a complete Electronic Office System in one program. Data base functions such as lists, accounting functions, word processing and communications are all handled by one sophisticated information system.

Another important feature is that all functions can use the same set of data or automatically call upon information stored in multiple files. For example, names and addresses could be used from a customer record and receivables from the accounting file—all easily inserted in a customized letter written with Silicon Office's word processor function. Up to 3 files per disk or a total of 6 files may be used at any time.

Getting Started

With a package of this sophistication and magnitude, documentation and ease-of-use are important considerations. And Silicon Office ranks high in both categories. Documentation includes a comprehensive 145 page Training Guide that covers all operations with clear examples and illustrations. A Reference Guide details each function, providing all possible options.

Silicon Office follows the format of its predecessor, OZZ, in guiding the user with extensive help screens showing all commands. Most commands are entered as two letter abbreviations with the system responding with the full command name for verification. After loading the program disk and entering the current date, Silicon Office is in the command mode. You now have immediate access to all features by just entering the appropriate command.

The evaluate command, EV, allows you to perform a full range of calculations to 14 place precision, including trig functions (sine, cosine, etc.), MIN, MAX, LOG, SIGN, and absolute value. You could enter 'EV RESULT = 3.14159 * 12,' or a number of related calculations. DM (display memory) will show all current variables with current values. However, the evaluate command is most commonly used in performing calculations on existing data.

Data Entry

Silicon Office uses two types of disks to store information. A 'data' disk holds actual data, such as a mailing list. A 'work' disk stores results of calculations, programs, and word processing text files. To start building your data base, PD, Prepare Disk, is used to format a data disk. FR, Format Record, now allows you to format the screen in any way you wish information to appear. Fields are defined as either text or numeric, with the starting location marked by typing<ESC> N for numeric. The end of both fields is marked with <ESC> F.

You move around the screen freely using the cursor keys. Lines can be inserted or deleted and text moved with the editing keys. Inverse and underline labels can also be added to make the screen display more understandable. The only limitations are the amount of data a record holds—252 characters, and the 'key field,' which is information unique to a record, must be the first field.

Once you've decided on the screen format, records can be added using the IR, Insert Record command. The entry mask that you designed will be displayed and information is entered in the appropriate spot. Of course, once information is entered there must be a way to use it. Silicon Office has a number of commands to search files for specific information. GR, Get Record, is used to find records containing specific information such as a name. For example, 'GR JOHN DOE' would display the first record containing JOHN DOE in the key field.

Even if you're not sure about the correct information, the Search Record function will look at all the data in a record and display the record if the specified information is contained in any field.

Additional commands allow you to move through the file forward or backward, change data, and delete records.

Calculations

File variables (plus the names given to data fields) can be entered into calculations. Perhaps you have a field named BALANCE in your file. Typing 'EV BALANCE = BALANCE + TOTAL' would update the BALANCE field by adding TOTAL to it.

One powerful feature is the ability to use information from other files at the same time. For example, 'EV TOTAL = BALANCE [CUSTOMERS] + GOODS TOTAL [INVOICE]' would add GOODS TOTAL from the invoice file to BALANCE from the customer file.

Programming

Silicon Office provides a way to actually write a program that automates all the commands available. Commands are entered on numbered lines, much like a BASIC program. The system provides the line numbers plus defined areas to

enter the command and description. Up to 100 program steps may be entered at a time. To get a feel for how this works, try to follow this example:

```
001| EV  |TOTAL = 0
002| GR  |#1 [CUSTOMER FILE]
003| EV  |TOTAL = TOTAL + BALANCE
004| NR  |:6
005| GO  |3
006| DM  |
```

What we've done is write a program to look at each record in a CUSTOMER FILE starting at number 1 and add the outstanding balance to a running TOTAL. After the end of the file is reached, DM displays the final total.

Line 1 clears TOTAL to zero. Line 2 gets the first record. Line 3 keeps track of the total. Line 4 instructs the program to look at the next record. The ':6' after NR tells the program to branch to line 6 if the end of the file is reached. Line 5 sets up a loop to add the balance figure in the current record to the running total.

The 'IF' command is extremely powerful for selecting information. You could design a program to total the balances of only those records meeting certain criteria. For example, 'IF CITY = PHILADELPHIA' would select only those people living in Philadelphia.

Programs are stored on 'work' disks for later recall. In this way you can create a system that anyone can operate without going through the manual process of individually selecting records or performing other functions. Programs can also be chained together, that is one program calls upon another. This gets around the limitation of only 100 steps per program. In effect, it allows you to create programs of unlimited size.

The program function also has the ability to handle variables and operator input. A program could go through each record and prompt the operator to enter a name or other information, automating the data entry function.

You could even write a program to create a new file from information contained in various existing files, even redefining the key field.

Word Processing

If all this were not enough, Silicon Office also contains a very complete word processor. It is entered using the ET, Edit Text, command. Though there are some limitations—there isn't a lot of flexibility in formatting text—the word processor also contains features found on very few other systems, such as its own help menu, search and replace commands, and screen justification.

When Edit Text is first selected the document format such as page width and length is determined. From that point you are presented with a blank page with the right and left margins outlined and a status line displayed on the bottom of the screen.

Text is entered anywhere on the page and will be printed exactly as shown. The cursor keys allow you to move around the page so text can be edited at any point. The DEL key deletes text, while the INST key places you in the insert mode. In this case, all following text is moved down to make room for the new material.

The <ESC> key serves as a control key for additional word processing functions. Using <ESC> plus the two letter short-form of the command will perform functions such as clearing, centering, or modifying large blocks of text. The disadvantage is the number of keystrokes needed to perform these and other special word processing operations.

The word processor is a page-oriented system, meaning as a page is filled it is stored on disk. However, if you later delete a paragraph from page 5 of a 20-page document, the remaining pages will be automatically adjusted to fill the void.

You may also store individual paragraphs and insert them anywhere in the text. Another powerful feature is the ability to easily enter and edit columns. This is one of the most difficult typing jobs and Silicon Office makes it a snap.

But more than that, the word processor was designed to be integrated with your data files. For example, to customize a letter with a name from the customer list you would enter 'Dear [Name[CUSTOMER FILE]]' and the actual name would be inserted in the proper spot.

Printing

The word processor printing function is designed to handle standard text, and reports using data files. To create a report including name, address, and balance you could set up a page like this:

```
[NAME      ]  [ADDRESS      ]  [BALANCE]
```

Those fields would be replicated down the entire page and the appropriate information would be inserted at print time. The calculate function could also be used to provide page totals. And for the really ambitious, a program could be written to automate the entire report process, selecting only certain records or specific information.

Communications

Finally, Silicon Office includes a communication facility to transmit any data or work files using the Commodore 8010 modem. This could be used to print a report on a remote printer or just chat with another system in a telex-like mode.

Summary

Silicon Office contains so much that it's almost too much to do justice to in a capsule view. However, it does what it set out to do—combine most office functions in one comprehensive system. All the pieces work together, eliminating duplicate effort. With the calculation and programming functions, almost any job can be tackled—from simple data storage and retrieval to complete accounting functions.

And the information needed to put this power to work is presented in complete, understandable documentation.

```
                        OFFICE PROGRAM


|001| gr |#1 [asset]
|002| sd |2
|003| gr |life end date = date(date aquired,est life mths,m)
|004| ev |depn amount = purchase price/depn period mths
|005| ev |time = period (today, date aquired,m)
|006| if |time < 1 : 13
|007| ev |net value = purchase price - (depn amount * time)
|008| ev |ytd = period (today, 01 Jan 81,m)
|009| ev |depn = ytd = depn amount * ytd
|010| ev |net value = net value - depn amount
|011| ev |depn ytd = depn ytd + depn amount
|012| ev |reserve = purchase price - net value
|013| gr |terminal date = date(date aquired, est life mths,m)
|014| sd |0
|015| ev |excess indicator = period(terminal date,today,m)
|016| sd |2
|017| ev |last update = date(today,0)
|018| ev |replmnt val = purchase price * 1.15
|019| ev |last increase = date (15 Mar 81,0)
|020| ev |last update = today
```

```
                    B E S T   B U S I N E S S   E Q U I P M E N T
                WALTER E. HENRY INSURANCE AGENCY : CUSTOMER FILE


LAST NAME |Duffy          |          FIRST NAME  |Greg       | INITIAL |S |
Address   |167 S. Spring Mill Rd.  |
City      |                |  State |PA|                      M=Monthly
Zip Code  |19468|                                            Q=Quarterly
Phone (H) |215-687-9750|                                     A=Annually
Phone (W) |            |
Bill To   |Same        |

-------------------------------------------------------------------------

Birth Date |09/03/52| Age  | 29| Occupation | Prof.  | MODE | 1| FREQ |2|

-------------------------------------------------------------------------

                          P O L I C Y   D E T A I L

Policy 1  | 15627772    |  Type |a|   Policy 2  | 399210011    |  Type |q|
Policy 3  | 68833399    |  Type |a|   Policy 4  |              |  Type | |
Policy 5  |             |  Type | |   Policy 6  |              |  Type | |

-------------------------------------------------------------------------

                                        Commission Amount |
Comments | None                      |  Total Premiums    |    987.50|
```

Microphys Programs

PET

VIC-20

2048 Ford Street
Brooklyn, New York 11229
(212) 646-0140

PET

VIC-20

FOREIGN AND DOMESTIC DISTRIBUTOR INQUIRIES WELCOMED.

Microphys, a leader in educational software development, is pleased to announce the release of several recreational software programs for use with the Commodore VIC-20 microcomputer. The VIC programs, described below, require a 3-K expansion cartridge and utilize the VIC's excellent color graphics and sound capabilities. Each program retails for $15 and is accompanied by complete instructions.

PROGRAM DESCRIPTIONS

PV901 - Missile Math: this program presents in a game format, an opportunity for youngsters (ages 5-15) to practice and develop the basic skills of addition, subtraction, multiplication, and division. Four levels of difficulty in each skill area may be selected. Problems in a given skill are randomly generated and missiles are launched at correct answers. The computer displays the results on each program run and may be directed to generate the same sequence of problems so that review and 'match play' (against an opponent) are possible.

PV350 - Cryptograms: this program permits the generation of 'secret' messages which are to be decoded. These cryptograms are displayed along with their unique code number classifications. To decode a cryptogram, the program is run from line 9000. Family members can challenge each other with their individually created messages. If you enjoy solving the cryptograms appearing in newspapers and crossword puzzle magazines, this program is perfect for you. Note: two VIC users may exchange encoded messages. User 1 creates a secret message and transmits this to user 2. The code number will permit user 2 to have **his** VIC decipher the message should he encounter any difficulty.

PV340-349 - Anagrams: this series of programs provides an educational challenge for virtually all age groups. The VIC randomly generates scrambled words which are to be identified. Two clues are provided in order to assist in this process. The clues in the school and college categories are generally definitive in nature. Many of the words used are part of the Microphys Spelling and Vocabulary series for the associated grade levels. Thus, reading, vocabulary, and spelling skills are reinforced by these Anagram programs. Note: the same sequence of words generated may be requested so that 'match play' is possible. There are 5 level-of-difficulty categories each consisting of two programs.

PV340-341 Recreational	PV342-343 College	PV344-345 High School
PV346-347 Junior High	PV348-349 Elementary	

PV375-380 - Wheel-of-Fortune Word Games: this series of programs represents an exciting challenge for every member of the family. Players try to fill in missing letters in a randomly generated title or phrase and earn and lose points according to the graphic display on a 'Wheel-of-Fortune'. The scores of as many as four players are displayed, 1000 points being required to win a given game.

PV375 Song Titles	PV376 Famous Places	PV377 Entertainers
PV378 Statesmen	PV379 Scientists	PV380 Sports Figures

PV601-644 - Missile Spelling: this series of 36 programs enables youngsters in grades 4 through 12 to practice and develop basic spelling skills. Each program contains 60 graded words. The VIC randomly selects groups of 5 words, one of which is spelled incorrectly. Missiles are launched in order to destroy the word misspelled. The words chosen for grades 7 - 12 correspond to the Microphys Vocabulary series. Note: there are 4 programs in each grade level.

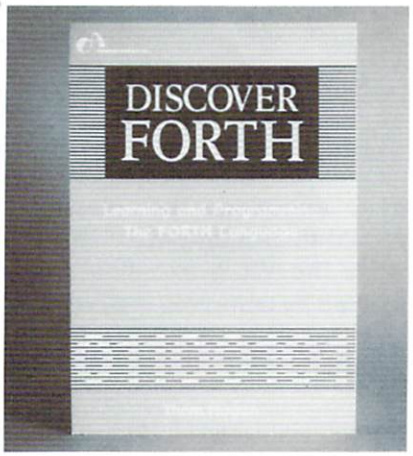
PV601-604 Grade 12	PV606-609 Grade 11	PV611-614 Grade 10
PV616-619 Grade 9	PV621-624 Grade 8	PV626-629 Grade 7
PV631-634 Grade 6	PV636-639 Grade 5	PV641-644 Grade 4

PV401-460 - Vocabulary: each vocabulary program randomly generates graded words which are to be defined. A sentence, in which the word is properly used, is displayed when an incorrect response is made. Using this contextual clue, a second opportunity to define the word is given. Reading and spelling skills are also reinforced as a more powerful vocabulary is developed. There are 10 programs in each grade level.

PV401-405 and PV431-435 Grade 12	PV406-410 and PV436-440 Grade 11
PV411-415 and PV441-445 Grade 10	PV416-420 and PV446-450 Grade 9
PV421-425 and PV451-455 Grade 8	PV426-430 and PV456-460 Grade 7

Educators should write for the new Microphys Fall Catalog which describes over 200 programs for use in Chemistry, Physics, Calculus, Mathematics, Vocabulary, and Spelling classes on both the high school and college levels.

NEW PRODUCT DEVELOPMENTS



Company:
A Osborne/McGraw Hill
630 Bancroft Way
Berkeley, CA 94710

Product:
Osborne/McGraw-Hill has released a new book by Thom Hogan entitled *Discover FORTH: Learning and Programming the FORTH Language.*

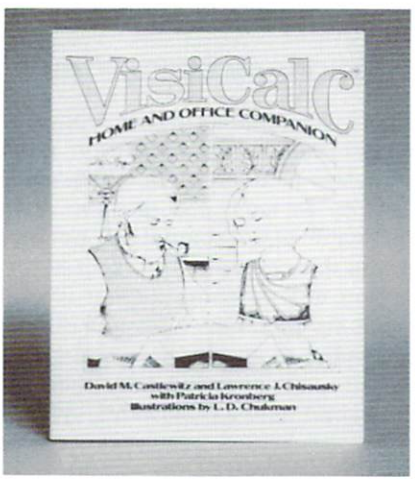
This thoughtfully organized introduction to FORTH, the computer language of building blocks, is written in a friendly, informal style. Beginners will find a wealth of information on this multi-faceted language and instructions to guide their programming skills up to an intermediate level. More experienced programmers can use Hogan's book as a valuable reference tool.

In the text, Hogan describes FORTH syntax, specifically applicable to both FORTH-79 and FIG-FORTH. Notes are included on logical extensions and alternatives to the current standard FORTH syntax.

DISCOVER FORTH provides a history of the language and a synthesis of material from programming manuals, independent programmers, and publications of the FORTH Interest Group. Many of Hogan's observations come from his own extensive use of FORTH in a major software project.

DISCOVER FORTH is the second book that Hogan has published with Osborne/McGraw-Hill. His first book, *Osborne CP/M User Guide,* was released in 1981 and has received much acclaim.

Price:
$15.00 (paperback)



Company:
A Osborne/McGraw Hill
630 Bancroft Way
Berkeley, CA 94710

Product:
Users of the popular spreadsheet program, VisiCalc®, will be delighted to learn of Osborne/McGraw-Hill's new release *VISICALC®: Home and Office Companion.* Not a tutorial, authors David Castlewitz, Lawrence Chisauski and Patricia Kronberg have compiled a compendium of VisiCalc models.

Fifty ready-to-use VisiCalc models, complete with sample print-outs, descriptive narratives, and coordinate listings, are presented. Users merely enter their own data in place of the sample data, or substitute a label or list of entries, to tailor the models to their specific needs. Algorithms and modeling techniques provide the experienced user of VisiCalc® with the opportunity to custom-design new and useful models.

The models cover seven areas of application: Loans and Investments, General Business, Inventory Control, Advertising and Sales, Personnel and Departments, Personal Finance, and Household Aids. All models are designed to accommodate most computers and versions of VisiCalc®. The models can be entered on SuperCalc™, too.

VISICALC®: Home and Office Companion delivers immediate, hands-on results for both the business professional, whose time is too valuable to re-invent commonly used financial and management techniques, and the home user who is looking for new applications for this versatile program.

VisiCalc is a registered trademark of VisiCorp.
SuperCalc is a registered trademark of Sorcim.

Price:
$15.99 (paperback)

Company:
Briley Software
P.O. Box 2913
Livermore, CA 94550-0291

Product:
BRILEY SOFTWARE announced that a new Northeast data module for its air navigational program called "RNAV3 NAVIGATOR" is now available to pilots. Data modules for the Pacific states and Western U.S. have been available since November. The program uses Commodore PET/CBM microcomputers.

The program currently has data modules for the Pacific Region (6 states), Western Region (11 states), and the new Northeast Region (17 states). Only the Continental States are included in these data modules. Versions covering the rest of the United States are being planned (Midwest and Southern).

The program is available from

local Commodore computer dealers or direct from BRILEY SOFTWARE, Box 2913, Livermore, CA 94550-0291 (8K PACIFIC: $25, 16K WESTERN: $30, 16K NORTHEAST, $30).

Price:
See description.

Company:
Graceful Solutions Inc.
1018 Landover Drive
Reading, PA 19605
(215) 921-3708

Product:
General Ledger Accounting System—Provides multi-department tracking of all financial transactions. The system is modular and supports subsystems for:
- Rental Property Management
- Calculating Depreciation
- Securities Portfolio Management
- Job Cost Accounting

Soon to be released subsystems include:
- Accounts Receivable
- Accounts Payable
- Payroll

Other functions are available on a custom basis. The system runs in the 8032/8x50 environment.

Price:
Software.....$575.00
Includes System diskette & Manual.
Install. & Training.....$45.00/hr
Minimum visit 1/2 day
Custom Software ...$Fixed Price
Call for quote.

Company:
Graceful Solutions Inc.
1018 Landover Drive
Reading, PA 19605
(215) 921-3708

Product:
Property Management Subsystem—Tracks the income and expenses for 250 properties or components. P/L reports on individual properties or groups of properties are provided. Income and expenses may be posted to up to 500 categories. These categories may be subtotaled in the P/L reports. All transactions are posted to both the general ledger and the property ledger with a single entry. The system, as provided, is expandable in groups of 250 properties. As required the system will automatically calculate depreciation for posting to the ledgers. Our General Ledger system is required. The system runs in the 8032/8x50 environment.

Price:
Software.....$475.00
Includes System diskette & Manual.
Install. & Training.....$45.00/hr
Minimum visit 1/2 day
Custom Software ...$Fixed Price
Call for quote.

Company:
Graceful Solutions Inc.
1018 Landover Drive
Reading, PA 19605
(215) 921-3708

Product:
Security Management Subsystem—Tracks the Income, Gain and Loss for 250 Securities. Each security is tracked by purchase lot. Portfolio summaries giving hi/low/average price and number of shares are available on demand. Simultaneous posting of:
- Purchase transactions
- Sale of securities
- Dividends
- Return of Investment

is made to both the security ledger and the general ledger. Also handled are the changes to cost basis caused by:
- Stock dividends
- Splits
- Acquisitions and
- Partial liquidations

The system, as provided, is expandable in groups of 250 securities. Our General Ledger system is required. The system runs in the 8032/8x50 environment.

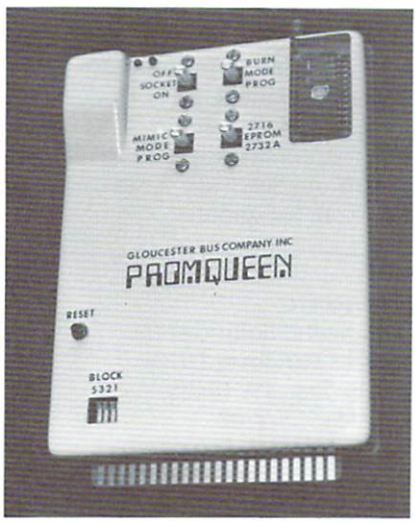
Price:
Software.....$475.00
Includes System diskette & Manual.
Install. & Training.....$45.00/hr
Minimum visit 1/2 day
Custom Software ...$Fixed Price
Call for quote.

Company:
Graceful Solutions Inc.
1018 Landover Drive
Reading, PA 19605
(215) 921-3708

Product:
Job Cost Management Subsystem—Tracks the Income and expenses for up to 250 active Jobs. The system provides a P/L report on each job or groups of jobs. Transactions are posted to both the General Ledger and the Job Ledgers from a single entry. Each of the up to 500 categories may be directed to a unique General Ledger Account. Full departmentalization of activity is provided in the General Ledger and the Job Ledgers. The system accepts job cost estimates which are reported as a comparison in the Job P/L Report. Provision is made for posting of labor costs to the job ledgers without corresponding posting to the General Ledger. This feature allows labor to be posted to the General Ledger thru a Payroll Journal and the various disbursements for benefits. The system, as provided, is expandable in groups of 250 Jobs. Our General Ledger system is required. The system runs in the 8032/8x50 environment.

Price:
Software.....$1050.00
Includes System diskette & Manual.
Install. & Training.....$45.00/hr
Minimum visit 1/2 day
Custom Software ...$Fixed Price
Call for quote.

NEW PRODUCT DEVELOPMENTS



Company:
Gloucester Computer Bus. Co., Inc.
6 Brooks Road
Gloucester, MA 01930

Product:
The Gloucester Computer Bus Co. Inc. PROMQUEEN CARTRIDGE provides EPROM programming, operating, and emulating capability for the Commodore VIC-20 computer. All necessary connections are made when the PROMQUEEN is plugged into the VIC's expansion port. The internal programming voltage power supply includes current limiting to prevent damage to the cartridge or the computer in the event of operator error or faulty EPROMS. The PROMQUEEN supports 2716, 2732A and 2732 EPROMs.

The PROMQUEEN contains 4 kilobytes of RAM for testing programs before burning them on EPROM. A MIMIC switch permits an external computer to access programs written into PROMQUEEN RAM. In MIMIC mode a jumper cable connects between the PROMQUEEN zero insertion force socket and the socket in the external computer for which an EPROM is desired. This feature allows use of a VIC with PROMQUEEN as a development system.

A DIP switch determines which of the four VIC expansion blocks is occupied by the PROMQUEEN. This allows the PROMQUEEN RAM to be used to expand the user BASIC memory of the VIC. It also allows the PROMQUEEN to be used in the Commodore VIC Expander in conjunction with other cartridges, such as Commodore's VICMON 6502 assembler/editor without address conflicts.

EPROMS in the ZIF socket can be run directly on the VIC. A ZIF socket isolate switch permits EPROM changes with the VIC on. Thus a variety of routines from different EPROMs can be used to develop new programs in the PROMQUEEN RAM.

The PROMQUEEN has a reset button for recovery from crashes, eliminating the need to store programs before testing them.

A toggle switch sets the PROMQUEEN for either 2716 or 2732A EPROMs, changing the programming voltage as required. 2732s can be programmed after readjusting a pot.

A general purpose hex loader program on EPROM is included. The program loads into the VIC's memory, freeing the EPROM socket. The hex loader permits convenient loading of code for processors other than the VIC's 6502. It includes labels, block move, offset calculation, number conversion and more. It also runs the burn process, checks EPROM erasure, and verifies burns. Software for storing BASIC programs on EPROM is included. The user's manual includes step by step instructions as well as tabular summaries of the Keystroke command options.

Price:
$169.50 plus shipping

Company:
Hayden Book Company
50 Essex Street
Rochelle Park, NJ 07662

Product:
PET GRAPHICS instructs the PET user on how to program graphic displays and provides a collection of machine-language subroutines that enable the PET owner to write more efficient programs. The book was given Commodore's seal of approval for use with a PET microcomputer.

Written by Nick Hampshire and published by the Hayden Book Company, Inc. of Rochelle Park, New Jersey, this eight-chapter book, with two appendices, offers a collection of error-free, machine-language subroutines. The subroutines written in the book provide a wide range of normally unavailable graphic functions that will help PET owners write more efficient programs.

The book includes fine resolution plotting, double density plotting, multiple screen page displays, interfacing the light pen with the PET Video Circuitry and ASCII codes used by the PET.

All programs in the book are also available on PET disk $25.00.

Price:
$16.95

Company:
Hayden Book Company
50 Essex Street
Rochelle Park, NJ 07662

Product:
THE LIBRARY OF PET SUBROUTINES provides a PET software designer with 53 proven subroutines that can save hours in the design of application programs. The book was given Commodore's seal of approval for use with a PET microcomputer.

Written by Nick Hampshire and published by the Hayden Book Company, Inc. of Rochelle Park, New Jersey, this 17-chapter book includes the logical framework used to build new programs in addition to the 53 sample subroutines. The purpose of each subroutine is explained along with

problems that may occur in its implementation. Most subroutines in the book are written in PET BASIC but some machine-language programs are also presented.

The subroutines provided include: drawing a border around the screen, bar graphs, displays of data files, sorting arrays, merging files, printing menus, random access files and many others.

All subroutines included in the book are also available on PET disk, $25.00.

Price:
$14.95

Company:
INI, Inc.
4013 Chestnut Street
Philadelphia, PA 19104

Product:
SWiFT is the interface between WordPro and database packages, such as Jinsam*; mailing list packages, such as ZIPPER*; other wordprocessors, such as Papermate*.

Most programs which interface with WordPro do so by creating sequential files, which WordPro can then merge into its own text files when outputting (such as when you print form letters).

SWiFT translates sequential files into WordPro-readable text files. Once the conversion is made, files created by other packages can be read into WordPro, edited, and printed or saved the way you like.

SWiFT can also be used to convert text files created by other wordprocessors into WordPro text files. For instance, Papermate saves text files in sequential format. If you had been using Papermate and then decided to switch to WordPro, you would have to type all of your text files over again—that is, if you did not have SWiFT.

SWiFT is designed to run on the Commodore 8032 CBM computer with 8050 disk drives.

(*WordPro is a trademark of Professional Software; Jinsam is a trademark of Jini Micro Systems, Inc.; ZIPPER is a trademark of INI Inc.; Papermate is a trademark of Michael Riley.)

Price:
N/A

Company:
INI, Inc.
4013 Chestnut Street
Philadelphia, PA 19104

Product:
ZIPPER is a mailing list package by INI, with flexible data format and output options, as well as sophisticated selection and sorting capabilities. Yet it is also quick and easy to use, with plain English prompts and error messages, and a special HELP key.

SET-UP:
*define your own label formats
-up to 16 items per label
-up to 70 characters per item

ENTERING DATA:
*define your own input formats, or INI's predefined formats
*specify a constant for any item (such as "Mr." for Title)
*reenter the last thing typed in for an item in a single key stroke
*automatically checks for duplicate labels upon input

OUTPUT:
*output to screen, to printer, or to disk file
*sort on any item or items
*select on any item or items
-use less-than and greater-than comparisons
-use full pattern matching
-combine comparisons using "and"s and "or"s
*print all labels, every n'th one, choose manually, or choose based on your own selection criteria
*define any output formats
-up to 8 labels/line
-any items and/or comments on any line

SIMPLE TO USE:
*menu-driven with simple commands and prompts
*complete error checking with meaningful error messages
*a HELP key gives you quick on-screen information
*an ESCAPE key allows you to back out of an option

INTERFACES WITH OTHER PACKAGES:
*reads labels from sequential files to take advantage of database packages or to convert from another mailing list
*outputs to sequential file so you can use a word processor to create personalized form letters

MISCELLANEOUS:
*over 2000 labels per disk
*merge or split files
*quick access to any label
*add, edit or delete labels quickly and easily

Price:
N/A

Company:
Microphys Programs
2048 Ford Street
Brooklyn, NY 11229

Product:
Microphys announced two software systems to assist administrators and classroom instructors in the *creation* and *analysis* of standardized tests employed in academic disciplines. Both systems are fully described in the new Microphys catalog. Be certain to contact your local Microphys dealer for more information and/or a complete demonstration.

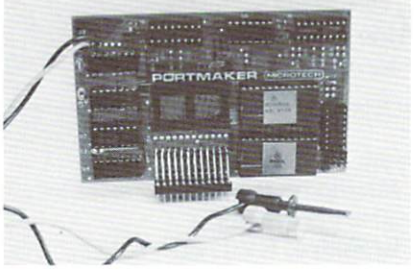
The *Exam-Generating System* is intended for use in a computer environment consisting of the Commodore 8032 (or 4032) computer; the Commodore 8050 (or 4040) disk drive; and a Commodore 4022 or 8022 printer. The system is supplied on diskette and is accompanied by complete instructions. The system is available from authorized Microphys dealers and retails for $200.

The Exam-Analysis System is intended for use in a computer →

environment consisting of the Commodore 8032 (or 4032) computer; the Commodore 8050 (or 4040) disk drive; the Commodore 4022 or 8024 printer; and a Chatsworth Data Mark Sense Card Reader. The system is supplied on diskette and is accompanied by complete instructions. The system is available from authorized Microphys dealers and retails for $200.

Note: the mark-sense cards for use with the above system are available directly from Microphys. Cards may be designed according to individual school district's specifications.

Price:
See description.



Company:
Microtech
P.O. Box 102
Langhorne, PA 19047

Product:
Serial I/O Port with Intelligent Terminal software for the Commodore PET/CBM!

PORTMAKER is a small dual serial port I/O board that plugs into the ROM socket of a PET/CBM to add serial 'RS232' capability. Portmaker contains two 6850 ACIA communication parts complete with buffer circuits and a simple baud rate clock. A ROM socket is also included allowing use of the original ROM for all but its highest 16 locations. Portmaker is also designed to operate with the Rockwell AIM.

A special intelligent communications software package to use with Portmaker is available. The Standard Terminal Communications Package (STCP) from Eastern House Software converts the PET/CBM into a data management center. Disk files can be serially transmitted. Received data can be recorded on disk or printer (or both) with complete control of all peripherals. A real time clock with display and alarm capability is also included. All STCP routines can be controlled from BASIC allowing automated telemetry. XON and XOFF protocol control codes are provided.

Portmaker is available in vertical or horizontal mount for $69.95. A complete package including a special Portmaker, cable, and the STCP software called COMPACK is available for $129.95. COMPACK will operate with all large keyboard Commodore machines. The software is available on 4040, 8050, or PEDISK II diskette. Contact CGRS Microtech, P.O. Box 102, Langhorne, PA 19047, (215) 757-0284, or your dealer.

Price:
See description.

Company:
Midwest Software
Box 214
Farmington, MI 48024
(313) 477-0897

Product:
MASTER LIBRARY PROGRAM—The ultimate utility program to organize all of your files. This program is not limited by the memory of the machine you are using as it writes linked sequential files and is capable of cataloging over 8000 titles. The titles, disk ID and blocksize are read from each disk almost as fast as you can feed them into the drive. The options allow you to create a new file, add to an old one or print only. As many files as necessary are created to handle your entire collection of programs and when you are ready, one continuous alphabetized listing with 100 titles per page is printed. Fast machine language sort of course.

Also included on the disk is a fast SEARCH program which lets you check an incoming disk for duplicate titles. The duplicates are then flagged for you on the incoming disk with a back arrow.

A bonus program is also included. This is DISK-A-BETIZER. This program alphabetizes the titles on your disks for you and recovers scratched files should you need this option.

Requires 16K or 32K of memory, single or dual drive and CBM or ASCII printer.

Price:
$29.50 (U.S. funds)

Company:
Midwest Software
Box 214
Farmington, MI 48024
(313) 477-0897

Product:
MASTER GRADES PROGRAM—A complete grades management system for secondary teachers. This program was written by teachers for teachers and took two years to develop and test. With it you can produce pages for your record book, alphabetized grade summaries of all students by grade level or subject, progress notes to parents and weekly (or oftener) reports in alpha or percentage order for all of your classes.

MASTER GRADES is completely menu driven, very user friendly, and uses a fast machine language sort. Most possible errors have been anticipated and trapped. Wherever possible, warnings and reminders are given to prevent you from making mistakes. Written for TEACHERS, not programmers.

Requires 16K or 32K of memory, a single or dual disk drive and a CBM or ASCII printer.

Price:
$29.50 (U.S. funds)

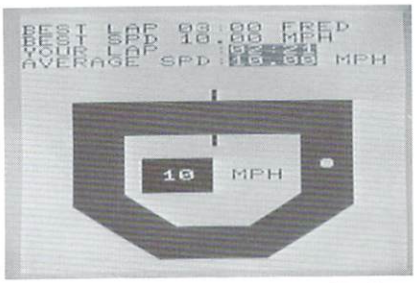
Softpedal

Company:
Practical Applications of California
P.O. Box 255768
Sacramento, CA 95825

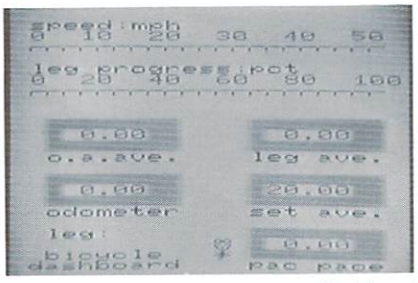
Product:
Practical Applications of California introduces *Softpedal,* a software program and transducer pickup system that converts an existing bicycle, or exercise bike into a computer-aided training and exercise machine.

The Softpedal programs display a simulated race course on any color television or video monitor, and allows the user to pace himself, race the clock, or another competitor. User input options add to the excitement. Unlike other training methods of this type, Softpedal overcomes the boredom of indoor bicycle training because of its unusual graphics and display options.

Softpedal is a program tape which consists of two interactive user programs. The first program, *BIKE RACE,* displays several real-time parameters. After entering the rider's name, a gun is fired to signal the start of the race. Average speed and current speed (in MPH), elapsed time, and distance are shown on the screen to give instant feedback on your progress around the course. The best time in completing the course and the user's name remain on the screen until another challenger beats him. This is an advantage when more than two players are competing together.



Bike Race



Dashboard

The second program called *DASHBOARD* presents a completely different approach to individualized training. Selections of how many lap "legs" are desired, the average speed to maintain for each leg, and the length of each leg are input at the start. A percentage completed bargraph and speed in MPH bargraph are displayed at the top of the screen. Your "bicycle dashboard" shows the average speed per leg, the distance covered up to that point, and your average speed. This program is designed with the serious enthusiast in mind, but is still "user friendly" so even younger riders are not intimidated. As with BIKE RACE, enhanced graphics are standard with DASHBOARD. Special characters like "Pac Rat" are visible on the screen and will signal you by jumping up and down and waving his arms while he squeaks to let you know you're dropping below the preselected lap speed.

Expected Softpedal retail price is $145.00 with stand (dealer inquiries invited).

Price:
@$145.00

Company:
Professional Software Inc.
51 Fremont Street
Needham, MA 02194
(617) 444-5224

Product:
Professional Software Inc. announced the introduction of *The Administrator,* a database system for the Commodore 8032 microcomputer.

The Administrator makes use of "invoice formats" so you can tailor displays and report formats to your own specifications.

A feature of *The Administrator* allows the linking of "master"records to "transactional" records. This creates dynamic record lengths, allowing better and more efficient use of disk space than ever before. Finally, a database sophisticated enough to track transactional history (e.g., inventory tracking, medical records, legal docket control, etc.) is available at an affordable price.

Price:
$595.00

NEW PRODUCT DEVELOPMENTS

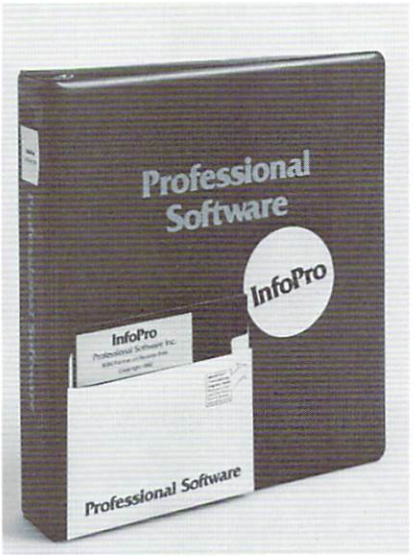
Company:
Professional Software Inc.
51 Fremont St.
Needham, MA 02194
(617) 944-5224

Product:
Professional Software Inc. has announced *WordPro-MAIL LIST,* a program designed for *mailing list* applications on Commodore computers. *WordPro-MAIL LIST* has also been designed to be used in conjunction with Professional Software's WordPro series of word processing software.

WordPro-MAIL LIST allows the user to easily enter, edit (revise/delete), or search hundreds of names and addresses with ease. *WordPro-MAIL LIST* has numerous time saving features including one that can read names and addresses already stored on disk and automatically enter them into a *WordPro-MAIL LIST* file. In this way, users who have been utilizing another program to manage their mailing lists can easily convert those files to *WordPro-MAIL LIST* without retyping.

WordPro-MAIL LIST will be available from Professional Software dealers in August 1982 with a suggested retail price of approximately $195.00.

Price:
See description.



Company:
Professional Software Inc.
51 Fremont St.
Needham, MA 02194
(617) 444-5224

Product:
Professional Software Inc. has announced the introduction of *InfoPro,* a new Information Management software program for the Commodore 8000 and 9000 series computers.

Using *InfoPro,* an end-user will be able to work with large amounts of information easily and effectively. *InfoPro* can manipulate or sort the information and print user-defined reports with specific formats. The special *Super Scan* feature allows the user to search very quickly through stored data for specific information, and then display that information almost instantaneously.

Another powerful and indispensable feature of *InfoPro* is its ability to interact with *WordPro*™, Professional Software's family of word processing programs. When used with commercial versions of *WordPro, InfoPro* allows the user to automatically create hundreds (or thousands) of personalized letters or documents.

Price:
$295.00

Company:
Professional Software Inc.
51 Fremont St.
Needham, MA 02194
(617) 444-5224

Product:
Professional Software Inc. is pleased to announce the introduction of a *multi-lingual* version of the WordPro 4 Plus word processing program. "WordPro-ML" (Multi-Lingual) was designed specifically for the Commodore 8000 Series computer and gives the user access to WORD PROCESSING IN FIVE LANGUAGES—English, German, French, Spanish, and Italian—with just a few easy keystrokes. In fact, the user can switch back and forth between languages without losing the text in memory. With *WordPro-ML,* characters and accents used in each of the five languages are displayed on the computer screen as well as upon printout. *WordPro-ML* will soon be adapted for use on the NEC Spinwriter, Diablo 630, and Commodore 8300 Letter Quality Printers.

For more information, contact: Professional Software Inc., 51 Fremont Street, Needham, MA 02194, 444-5224, TLX: 951579.

Price:
N/A

Company:
Scientivic Software
Dept. CM, 525 Lohnes Dr.
Fairborn, OH 45324

Product:
'High Resolution Plotting Routine' turns the 1515 Printer with a VIC-20 into a high resolution plotting machine. Resolution is 72 dots/inch horizontally and 63 vertically (the highest possible on the 1515). Will plot the results of user supplied programs, functions, or data logging routines (including from the keyboard). Size of the plot and automatic scaling with number labels and tic marks are continuously user adjustable. Scales are provided on all four sides. Size of the plot is limited only by the paper, allowing strip chart recording with folded paper (the end scale is generated after the plot is finished). Requires only about 2.7K of RAM (the rest of the memory you have, including expanders, is available for your point generating program). Written in BASIC, it should run on any VIC system.

Includes instructions and demonstration program.

Price:
$10.00

Company:
MicroSpec Ltd.
2905 Ports O'Call Court
Plano, TX 75075
(214) 867-1333

Product:
MicroSpec Ltd. announces a Low Cost Disk Based Data Manager System for the VIC-20 Computer.

MicroSpec Ltd. has just released its disk Data Manager for the Commodore VIC-20. It is a comprehensive data management system that allows the user to define and manage their own database and record entries on diskette. The system's menu includes:

*CREATE A FILE builds the data base and creates a description file on the disk.

*ADD A RECORD allows entry of records to the disk.

*DELETE A RECORD deletes records from the file and makes the record's space available to the user.

*CHANGE A RECORD allows the user to change or edit any previously entered record.

*BROWSE THROUGH THE FILE allows the user to scan or sequentially review the records in the data base.

*SEARCH THE FILE allows the user to search the data base and display specific records on the screen according to user criteria.

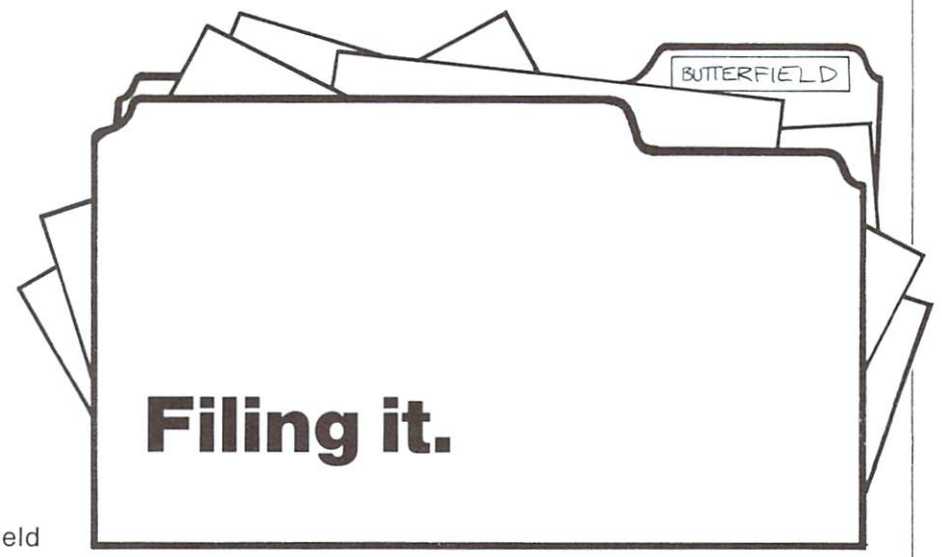
*PRINT THE FILE allows the user to print a hardcopy report according to his own specifications. The order, limit, number of fields, field position, and optional totals by numeric fields may be specified.

*EXIT saves the binary record key map and returns to BASIC.

The system will store up to 1200 records on a single disk. The data base file is a true random access file in that any record can be retrieved and displayed directly without having to read any intervening records. The system requires a minimum of 8K free memory to operate. The Disk Data Manager System sells for $59.95 complete with documentation and storage binder.

Price:
See description. C=

BUTTERFIELD ON COMMODORE



Filing it.

by
Jim Butterfield

Once you have learned how to input from the keyboard and output to the screen, it's easy to take the next step, and input or output using other devices.

The printer is handy, of course. But the super power comes from devices you can both read and write. That way, information can be stored now and brought back in later. You can store names, addresses, phone numbers . . . and read them in when you need them. This gives you two advantages: first, the information isn't lost when you turn the power off; and second, you have lots of storage space even if your computer memory is small.

The golden rule.
You'll find it easy to remember the golden rule of input and output: the information going out is almost exaclty as you would see it on the screen. So if you asked to print out a value of 167, the following characters would be sent to the cassette, printer or disk: Space; 1; 6; 7; Return. That's almost exactly the same as would go to the screen; we wouldn't see the Return on the screen, but we'd see its effect since a new line would be started.

The opposite side of the golden rule concerns input. If the above value was written to a device, and later we rewind and ask to INPUT from that device, the program will receive exactly the same information as if we typed on the keys: Space, 1, 6, 7, and Return. On the keyboard, RETURN signals that we are finished; and it means the same when the information comes from some other device.

Two special situations should be mentioned. You might have noticed that if we say PRINT 167 an extra character is delivered to the screen: behind the last digit, 7, there's a cursor-right. You may not notice it, since it doesn't print, but it's there. This extra character will not be sent to other devices. That's good because we don't need it; we save the space and no harm is done.

The other situation is another invisible character. Many versions of Basie send one or more characters after RETURN. Basic 4.0 does not normally send it, but most other Basics send a special character called a Linefeed. The Linefeed is a nice character for certain types of printer: it may be needed to move the paper up ready for printing the next line. But it's wasted in data storage, and might even give us a little trouble. More on this later.

Writing a file.
It's easy to write a file. All we need to do is: Open it, which tells the computer to get everything ready to go; Print the stuff; and then Close it, which tells the computer to wrap everything up.

Let's do it. If you have cassette tape, type:

```
OPEN 6,1,1,"DATAFILE"
```

or if you have disk, type:

```
OPEN 6,8,2,"0:DATAFILE,S,W"
```

...and in either case, your file number 6 is ready to go.

Now we can write a few things. Let's try some numbers:

```
PRINT#6,3
PRINT#6,123
PRINT#6,3*4+5*6
```

And a few names:

```
PRINT#6,"HELLO"
PRINT#6,"MY NAME IS FRED"
```

Finally, we wrap up the file with:

```
CLOSE 6
```

A few notes. Did you notice that after we opened the file, the coding was the same no matter whether we were going to tape or disk? The OPEN statement sets everything up for us. This can make things very easy.

Note that we use one print statement for one item. Don't try punctuation: PRINT#6,3,123 would not work right—we will need that extra RETURN when we read back the data. It's also interesting to see that expressions are worked out before being printed, so that 3*4+5*6 will be placed on the file as value 42.

Now for that sneaky Linefeed. You don't really need to worry about this if you have 4.0 Basic or if you are

using cassette tape, but it's good practice. Those PRINT# statement wrote the information we asked; then a RETURN, which we wanted; then a Linefeed, which we didn't want. We can get rid of the unwanted Linefeed by writing the Return ourselves—it codes as CHR$(13). So we might more correctly write:

```
PRINT#,"HELLO";CHR$(13);
```

...and don't forget both semicolons.

Reading it back.

This is just as easy, except that we need to write these statements as a program. INPUT and INPUT# won't work as direct statements typed on the screen. so we code:

```
100 OPEN 4,0,0,"DATAFILE"
```

or, for disk:

```
100 OPEN 4,8,3,"DATAFILE"
```

And continue with:

```
110 INPUT#4, A$
120 PRINT A$
130 IF ST = 0 GOTO 110
140 CLOSE 4
```

What's ST doing? That's the Status word. If it's zero, we are reading our file normally. It it's non-zero there is something going on—usually we are at the end of the file (ST will equal 64 in this case).

Your data should come back very nicely just as you wrote it.

Conclusion.

It's not hard to write and read files. We'll pick up a few fine points next time around. C=

Book Review

PET Graphics

written by Nick Hampshire
Reviewed by Elizabeth Deal

By writing PET GRAPHICS Nick Hampshire has, once again, contributed a major work about the PET that should be of interest to many PET/CBM users. PET GRAPHICS has originally been published by Nick Hampshire—Commodore Publications in England. In the U.S. it is published by Hayden Book Company. The British version is reviewed here. It is my understanding that Hayden made no changes in the original plates. The price of the book is $17, the price of the accompanying floppy is $25.

The book is a collection of graphic routines written in machine code, numerous Basic programs which use the routines, a thorough description of PET's graphic system as well as complete documentation of the code. It is a goldmine of useful routines, and the documentation provides a learning experience of great value.

If you have been annoyed in the past by getting a jazzy routine, or typing one in from DATA statements in a magazine and you never know HOW the code worked, or HOW you could customize it—then all your problems are over. PET GRAPHICS answers both questions.

Basic programs are listed in a neat fashion. They contain comments to aid understanding of the code and the process of hooking up the machine code routines. The machine code routines are fully documented within their assembly listings. The assembly listings use standard 6502 notation. I wish one existed, perhaps a simple ASCII version that users could somehow stuff into their assemblers . . .

General Review

It is not at all necessary to know machine code to use the routines; herein lies their utility. However, this review will touch on the subject in an attempt to present a complete picture of what might be involved.

The description of how to use the routines is perfectly unambiguous. The process itself is simple, anywhere from two to four POKE statements followed by a SYS call will do some nifty job. The book is neat, well organized; routines that perform similar functions have been grouped together. There is an index and a table of contents, both correct; both simplify the task of finding a listing in this large text (a bit over 200 pages).

The book lucidly describes many aspects of PET's video circuit, character generator and general aspects of graphics. It's good education. The author takes the reader from a simple PRINT statement to more complex and fast machine language graphic macros which perform feats such as:

Quarter character point and line plots, improved from the PET REVEALED version, "high resolution" (1/8-character lines) graphics in either x or y dimension at one time, including drawing "high resolution" bar graphs and function graphing; routines for setting up windows, for filling a screen area with a character, for reversing an area, for moving drawings around the screen (four directions), for scrolling a function horizontally; routines for setting up large (macros) characters and moving them around, "large screen" routines, where the screen is considered just a window to a conceptually larger area; a set of routines for interfacing a lightpen to the PET, and lots more.

The 24 demonstration programs use the routines in dazzling graphic displays, as well as more mundane 'enter product name and transaction number' type jobs. The latter elaborate form-type entry of data is described in the LIBRARY of PET SUBROUTINES. Some programs are simple demos, while others are fairly elaborate routines that can be used for graphing values read from instruments attached to the user port, etc. All are fun to watch, many constitute nice building blocks for further uses.

The hookup to Basic consists of nothing more than, for instance:

POKE86,column:POKE87,line

POKE88,width of block:POKE89,height+1
SYS30629

in order to scroll down a block. The block leaps down in no time, making animations possible. The code talks back via the error status, which indicates out of range parameters.

I have tested just about all the routines in the book with the exception of the LIGHTPEN code. Having none, I was only able to simulate a lightpen with switches to see the logic. The code looks all right to me. Further, I have it from a reliable source that the schematics are excellent and a good lightpen can be built.

All lightpen and most function graphing routines properly place the 0,0 point in the lower left hand corner of the screen. Several other routines, particularly those that deal with animation or setting up windows for scrolling data use a system of row-column positioning on the screen, where rows correspond to PET's line numbers.

A small problem exists, which is both east to detect and fix: in the block move the code suffers form the common 'off by one' symptom. Hence, you will have to correct for it in your Basic code by adding or subtracting one when the need is apparent.

Moving objects on a diagonal is slower than in other directions as more POKE statements are required. Careful setup is needed, as Basic has to remember and update position of the object. The code, unfortunately obliterates some of that information.

It is a bit tricky to handle movement of multicharacter blocks near the edges of the screen. The key decisions (disappear, wraparound, bounce) need to be coded in Basic. Basic tests slow down an otherwise fast process.

Miscellaneous Management Details

After loading some 125 blocks into the PET, RUN initializes things, sets the top of PET to $7000 (for use by the demo programs) and is ready for use. I find it to be an odd procedure but one that is foolproof and easy on the user who does not want to be bothered by any pointers and machine code details.

If you do want to know the details, this may be of interest: the actual GRAPHICS code is in the $7400-7CAC area. Alternately, the LIGHTPEN code goes into $7766-7823 area. For quick loading and normal use it makes sense to just use the required code and set up pointers by yourself. Lightpen code requires two routines from the Graphics package to be relocated. It would have been nice if the relocated code were provided on the floppy, and might well be as this review is being written.

The package coexists well with "normal" programs and various system routines. With one exception, the PET is left in a clean state.

About 20 locations in the second tape buffer are used for temporary storage. If you use the buffer for machine code system-type routines, you should move those routines; they don't belong there anyway.

Most importantly, memory locations used by such common system programs as POWER™ and TOOLKIT™ are not written over. The usual DOS-wedge area also remains intact, though with a big gap.

I found one instance of conflict: the code uses locations 0 and 1 for temporary storage and does not restore them. Subsequent use of the USR function (perhaps hours later) mysteriously fails since the graphic package does not replace the "JMP" in location 0. You can fix it by POKE 0,76 when you're done. These locations also interfere with Nick Hampshire's own repeating key routine (PET REVEALED), which should be disconnected before the graphic package is used: type LOAD, then STOP.

The book is written primarily for Basic users of the Upgrade PETs; Basic 4.0 version is in the works. But if you can't wait, and if you know how to make small changes, I think it can be done considering the complete documentation. One ROM address *must* be changed ($E3D8) and different locations in the buffer *may* be needed. Consult your dealer and ask the SUPERMON for help.

80-column systems are not supported; all screen address calculations depend on the 40-character screen width. An 80-column version is in the planning stage. User conversion to an 80-col PET would be tricky, if not impossible without the assembler code. As I see it, it requires inserting several instructions and that could get a bit nasty in ten pages of code.

People who are at least slightly familiar with machine code can customize the code. Relocation (very desirable!) is possible due to the excellent documentation. Most routines will require some address changes to be made in moving—many contain constants at absolute adresses, and most contain one or two internal jumps.

There is/was one bug in the code. It sits there quietly causing no trouble, until Basic data or programs go beyond $3000 (12K). One variable has gotten loose. A1 needs to live at $0344, not $3044; therefore three instructions need to be changed: they begin at $7B9D, $7BB9 and $7C89.

It is likely that by the time this review hits the presses the bugs will have been corrected (I know AB-Computers has made the changes). If not, fixing is rather simple.

Overview

I find the book and the floppy both fun and relatively easy to use. The book contains interesting details about the PET. The routines are easily connected to Basic. They are there and they are useful. You don't ever have to write them from scratch.

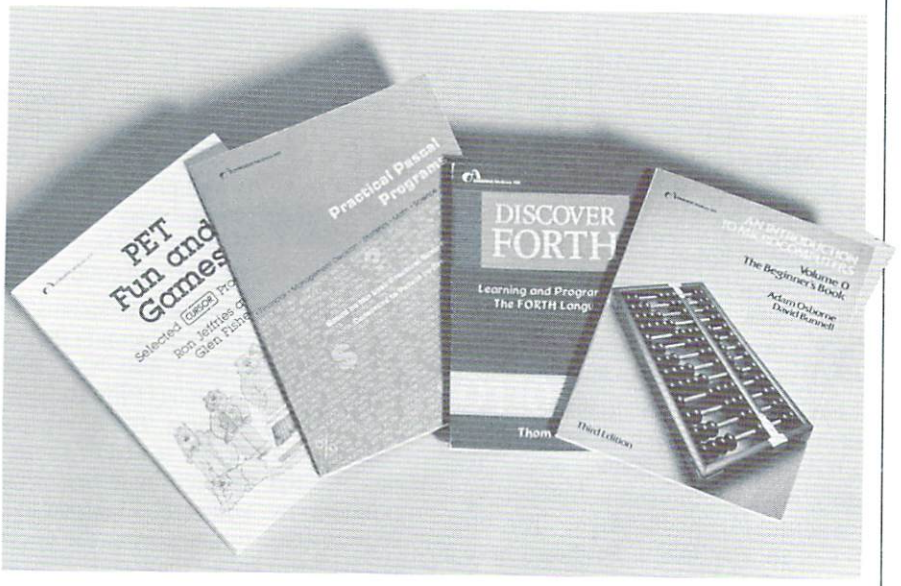
ATTENTION PROGRAMMERS

Commodore is compiling a list of software written for our computers. If you have software you would like to have included in this listing please submit the following for review:

- copy of program on disk or tape
- documentation describing purpose and utility of the program
- specify equipment necessary for program operation
- information on price of program and where the program can be purchased

Please submit this information to:

SOFTWARE Committee
Commodore Business Machines
487 Devon Park Drive
Wayne, PA 19087



Practical Pascal Programs

by
Greg Davidson, McGraw-Hill

For those of you who are familiar with the types of books out that are a collection of BASIC programs for fun, business, education or whatever, this book is another in that type of series but with a twist. The programs are in UCSD Pascal.

This book presents over 30 responsible programs in areas such as business, math, statistics, and personal finance. I found the coding, from a programmers point of view, to be concise and the programs that I actually entered worked as written.

I recommend this book to people who want some good examples of UCSD Pascal code for a learning experience, and to people who just would like to have the use of some good programs.

PET Fun and Games

Selected CURSOR Programs
by
Ron Jeffries and Glen Fisher,
McGraw-Hill

Well, the CUSOR people have helped do it again. Not only have they produced one of the best electronic magazines out to date, but have now allowed some of their best games to be published in hard-copy as well.

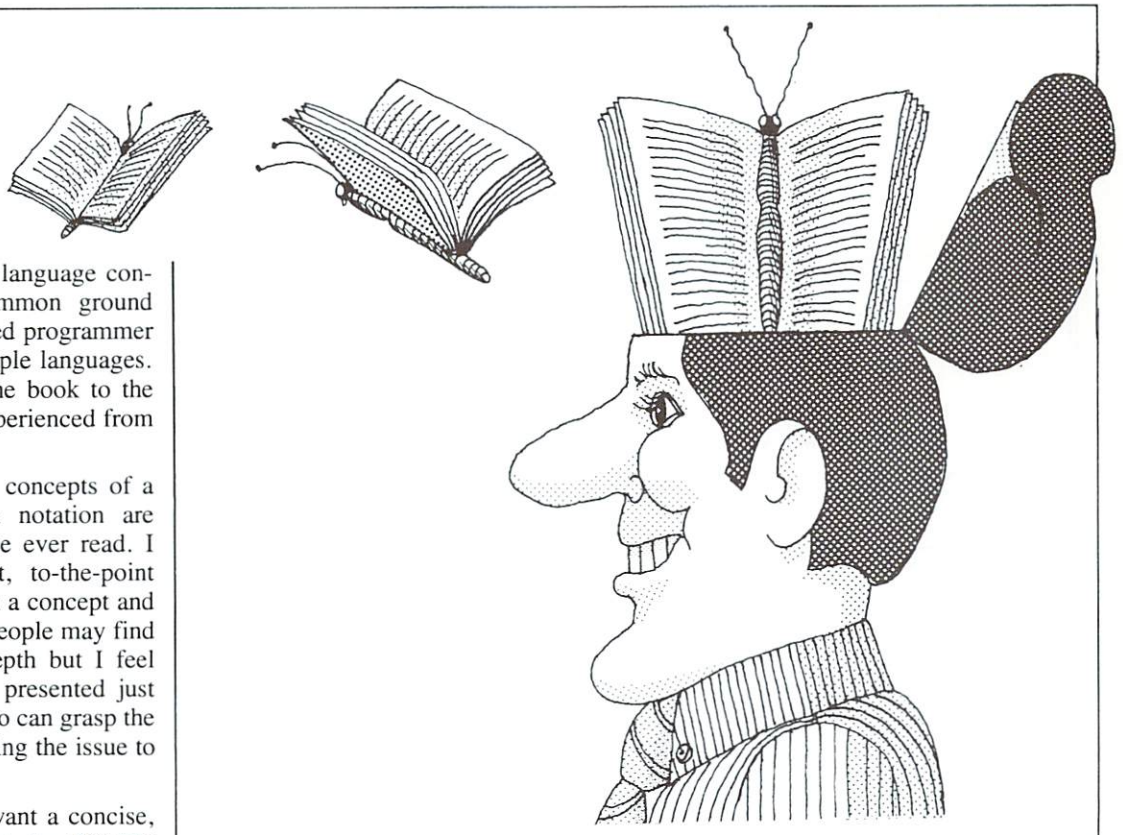
The introduction presents to the reader the possible machine differences, so that you can run these programs on the PET as well as the newer CBM series machines. I found the special symbols used for the different 'unprintable' graphic characters used for the cusor movements to be inventive and most readable. This is important as this one item causes a lot of good programs to be entered incorrectly and to be ignored.

Buy this book! It's fun, and serves as a very good example of how to program the PET and CBM to the machines' capability.

Discover Forth

Learning and Programming
the FORTH Language
by Thom Hogan
McGraw-Hill

In a word this book is Great! It presents the HOW, WHY, and WHY NOT of the up-and-coming new language FORTH. The book is easy to read and the information is presented clearly. Technically I do not recommend the book to the computer novice



as it deals with several language concepts that are not common ground except to the experienced programmer who has mastered multiple languages. But I do recommend the book to the novice as well as the experienced from an informative basis.

The explanation of the concepts of a STACK and Post Fix notation are some of the best I have ever read. I also enjoyed the short, to-the-point chapters, that dealt with a concept and then moved on. Some people may find this style lacking in depth but I feel that the information is presented just enough so that those who can grasp the idea have, without beating the issue to death.

For those of you that want a concise, and entertaining look at the FORTH language, I recommend this book highly.

An Introduction to Microcomputers

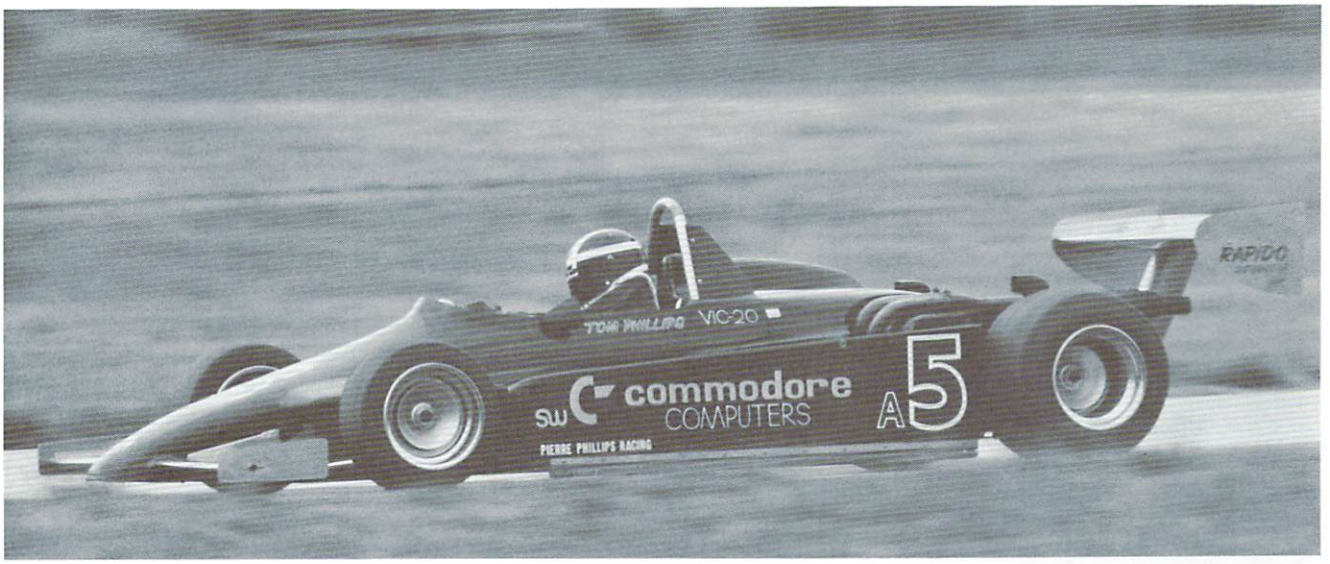
Volume 0
The Beginners Book by
Adam Osborne and David Bunnell
McGraw-Hill

This is one of the few 'introduction' books that is truly informative and entertaining for the beginner and yet does not offend those of us who grew up 'in the business'. The information is presented in clear, concise sentences. A fair representation of hardware and software is made with no real subjective inferences about either.

The book is sectioned into six parts, from "The Parts That Make the Whole" to "Putting It All Together." I found the first couple of sections directed, as they should be, to the interested novice and the last few chapters dealt with the more technical descriptions of a microcomputer and how it works. This gradual approach makes the book readable, and logical.

I would recommend this book to anyone that wishes a little more insight or just a refresher course into what makes a 'micro' a computer. C=

Review compiled by Paul Goheen



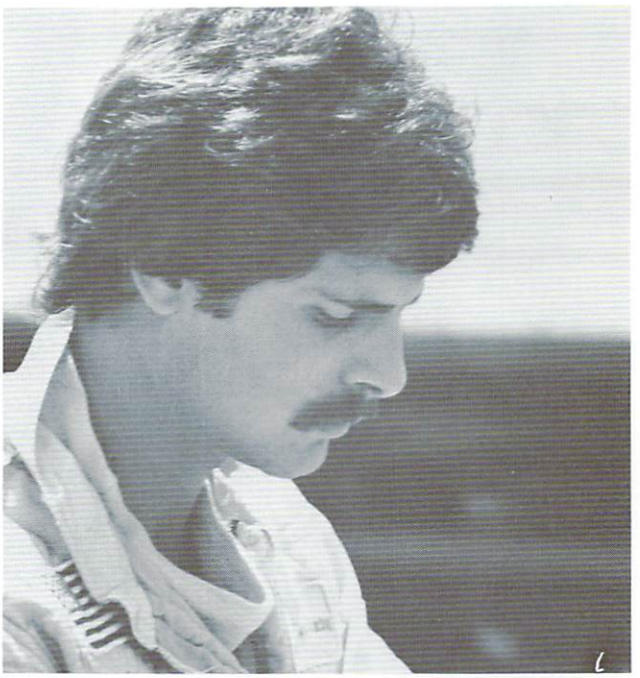
Commodore Backs A Winner!

The game combines road racing, time trials, and night driving. Fast-paced, authentic and challenging! This is obviously a description of Commodore's exciting game cartridge "Midnight Drive." However, the company recently became involved in the "real thing" when it supported Commodore dealer SW Computers and Electronics (Portland, Oregon) sponsorship of Tom Phillips and his car in the 1982 Rose Cup Formula Atlantic Race in Portland.

How did Phillips fare? Well, he only came in first and set a new lap record for the track. Phillips, 22, of Gresham, Oregon was described by the English racing magazine "Autosport" as a "sensational newcomer." Phillips has won 21 of his first 53 races. His father Pierre is also a race team manager.

The sponsorship was the brainchild of Steve Wilson, manager of SW computers. Wilson said, "I've been involved with racing for years and I have watched Tom Phillips come up through the ranks. I've known and worked with the Phillips' for years and they are winners. I've had the idea for a while and I wanted to go with a winning team because Commodore has a winning lineup."

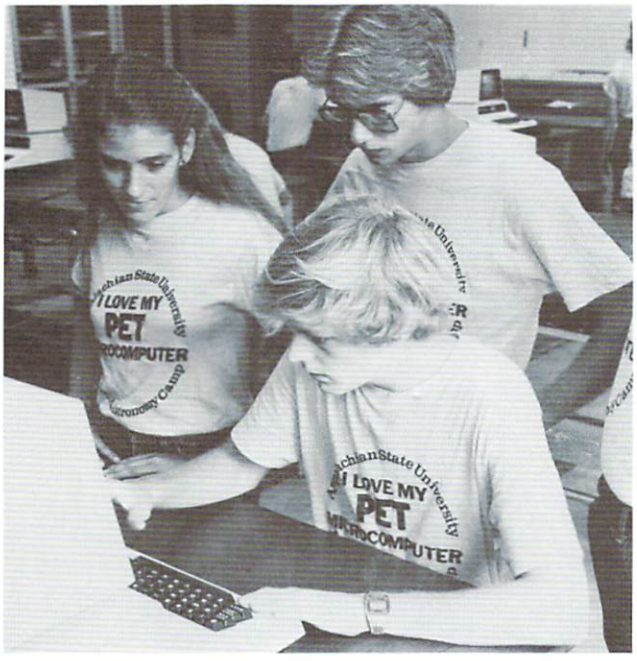
The idea of a computer company sponsoring a race driver may seem a bit odd because normally you see automobile related products sponsoring racers. Phillips doesn't think it is odd at all. "Computers are starting to play a big part in race cars and it is really coming in to vogue now," he said. According to Phillips one important area that computers can be used in is the design of race cars. "The idea is that if you have a design idea you can check with the computer first, avoiding a lot of trial and error in the shop, and engine troubleshooting is another area where computers are going to be important in racing."



So Commodore, with so many firsts in the computer industry, has become one of the few computer companies in the sport of auto racing. Which is just fine with Tom Phillips. "It was really nice of Commodore to help me out," he said. "It's just the kind of people that someone needs around racing."

Summer Fun

Last summer, computer camps were in the news as something unique and exciting. To prove it wasn't a fad we want to tell you about another addition to the computer rolls. "The Wonderful World of Microcomputers, Physics & Electronics" is a camp sponsored by The Faculty of the Department of Physics and Astronomy, Appalachian State University, Boone, North Carolina.



Camp director Dr. Thomas L. Rokoske said, "One reason the camp is a success is that we give people two hours every night to just do whatever they want with computers, and they get very creative with things like graphics and animation. People really loved the camp last year and some returned this year. "The camp uses PETs because Rokoske said "We're on a low budget so the PETs made a lot of sense."

Last year's camp emphasized astronomy with some unusual software to do things like tracking the Soyez 6 satellite. Campers also toured an observatory and watched the Cosmos series. This year, the emphasis was on electronics. Topics covered were Use of the Microcomputer, BASIC programming, animation, electrostatics, power supplies, electronic devices, circuit boards, digital gates, binary codes and computer music. Campers also toured the campus Computer Center and an electronics component manufacturer. Time was also set aside this year for anyone who was interested in building a microcomputer from a kit.

This year, campers came from as far west as Kansas City, Kansas, as far north as Washington, DC and as far south as Florida. However, there were still traditional camp activities including swimming, volleyball, handball, tennis and hiking. It's nice to know that there are some activities that can never be replaced by a computer!

PROJECTIONS & REFLECTIONS

Free Hardware, Free Manufacturing, Free Distribution, Free Advertising

Now that I have your attention, tear out this page, make 100 copies of it and give it to everyone you know that can write marketable programs.

For those of you that you didn't realize, Commodore has just announced five new hardware products, including the Commodore 64, which is one of the most remarkable introductions in the microcomputer market this year. Would you like to become part of a growing crowd of people that are already writing software for the Commodore 64? Companies like Professional Software, Terrapin, TMQ, Cimarron, and many more have already begun creating the software products that will enjoy substantial sales on this new and exciting machine.

We are interested in your software ideas, and will try to make it easy for any software vendor to complete and distribute their products. We are not saying that we are giving away the farm, or will underwrite all projects. But we do want to hear your proposals, make your company part of our growing database of vendors, and evaluate how and where we might help you develop software for the Commodore line.

Some of the products currently underway are wordprocessors, databases, and mailing list systems. We would like to see software for the personal market in finance, education, agriculture, and of course entertainment. On the business side, the price of the Commodore 64 ($595) and an affordable 170k disk drive make it a natural for a series of 'stand alone' products like inventory, A/R, A/P, and G/L. The color graphics and 39k work space allow the financial analyst to prepare some really great projections. And we must mention the communications ability with our $100 modem.

So, please, if you or anyone else that you know of has marketable software that is finished or in design, send me your proposal on what your product is, what market it addresses, and any other information that you feel is needed. If the product currently runs on one of our other products, or even on another machine, send me a copy.

My goal is to have as many software vendors as possible writing for our products. To accommodate you, we have added staff and budget. We are ready to help you help us. Please send your proposal to:

Paul Goheen, Director Applications Software
Commodore Business Machines
487 Devon Park Drive
Wayne, PA 19087 C=

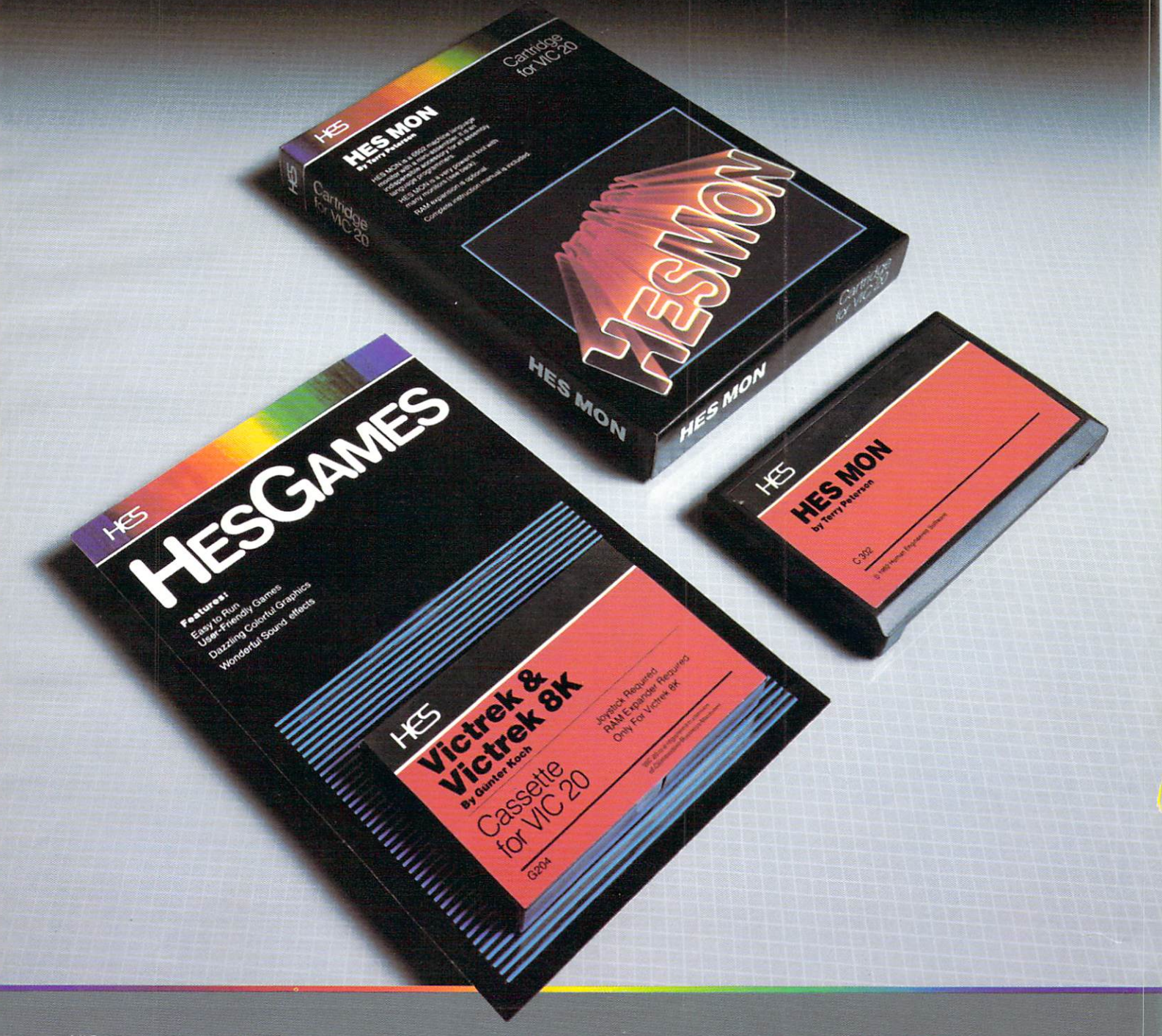
Circle #43 on the Reader Service Card

Advertisers' Index

ABM Products 8
R. J. Brachman Assoc., Inc. 73
CMS Software Systems 15, 44
Cascade Computerware 30
The Code Works 70
Commodore Business Machines, Inc. 7, 58, 59
COMPUTE Books 84, 85
Computer Marketing Services, Inc. 47
Computer Mat 30
Connecticut microComputer, Inc. 27
Cow Bay Computing 104
Cyberia, Inc. 35
Eastern House Software 6, 27, 66
Electronic Technology Corp. 68, 104
French Silk 50
Human Engineered Software ..... IBC
INI, Inc. 99
Info Designs ..... IFC
Interlink 10
Leading Edge Products, Inc. ..... OBC
Lemdata Products 66
Madison Computer 16
MAG, Inc. 23
Microphys Programs 89
Mirco World Electronix, Inc. 107
MIS 75
Micro Spec., Ltd. 54
Metron Computer Systems 103
Microsignal 55
Peripherals Plus 13
Professional Software, Inc. 3, 9
Skyles Electric 31
Small Systems Engineering 48, 64
Toronto PET Users Group 70
TNW 43
Voyager Software 104
Wunderware 51

commodore

Your VIC 20 never had it so good!



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

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

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

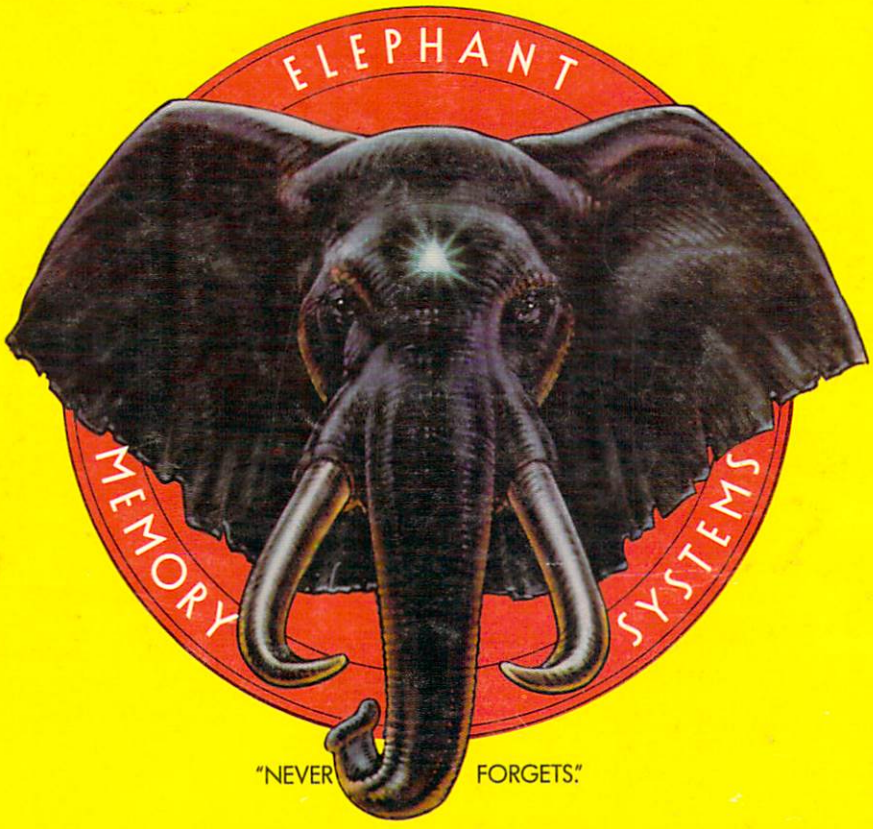
HES

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

Circle #44 on the Reader Service Card

VIC 20 is a registered TM of Commodore

REMEMBER:



MORE THAN JUST ANOTHER PRETTY FACE.

Says who? Says ANSI.

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

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

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

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

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

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

ELEPHANT.™ HEAVY DUTY DISKS.

Distributed Exclusively by Leading Edge Products, Inc., 225 Turnpike Street, Canton, Massachusetts 02021
Call: toll-free 1-800-343-6833; or in Massachusetts call collect (617) 828-8150. Telex 951-624.

Circle #45 on the Reader Service Card