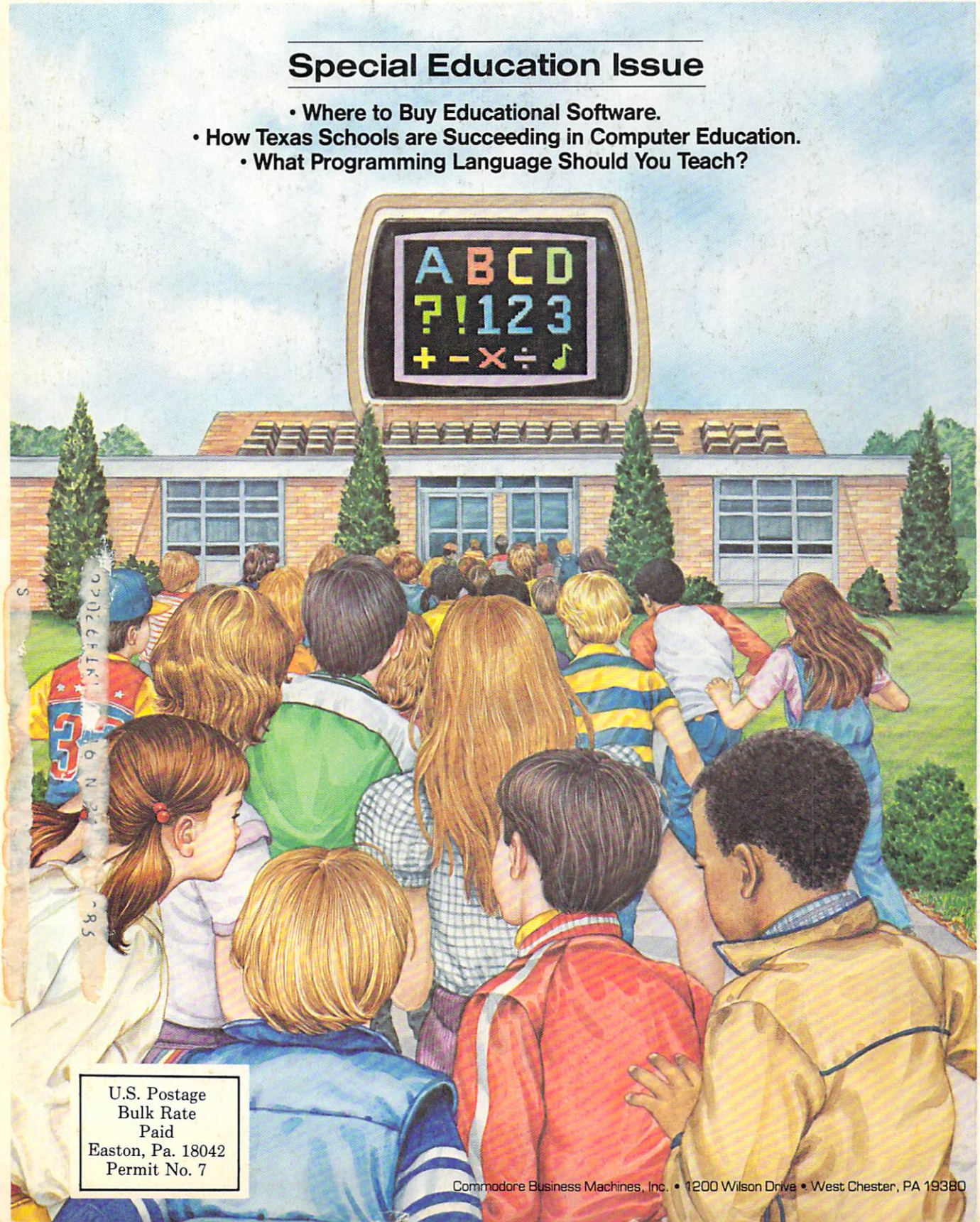
the commodore microcomputer magazine

May 1983 Vol. 4, No. 2
$2.50 US
$3.50 Canada

Special Education Issue

- Where to Buy Educational Software.
- How Texas Schools are Succeeding in Computer Education.
- What Programming Language Should You Teach?



U.S. Postage
Bulk Rate
Paid
Easton, Pa. 18042
Permit No. 7

Commodore Business Machines, Inc. • 1200 Wilson Drive • West Chester, PA 19380

Info-Designs announces . . . Small Business Accounting for the Commodore-64

Now the power of Info-Designs Management Accounting System is available on the Commodore-64 in a full and faithful version!

Thousands of these quality business accounting software packages have been sold on the CBM computer at $595 each. Now, similar features are available to the small business user on the new Commodore-64 for $199 per module!

Select the accounting modules you need—

- Accounts Receivable/Billing
- Accounts Payable/Checkwriting
- General Ledger
- Inventory Management
- Payroll
- Electronic Calendar ($149)

Limited Introductory Offer!

Our SoftPack combination contains the "Big-3" accounting—A/R, A/P and G/L—for only $495. Available for immediate delivery!

Flexible Design

The accounting system will work with one or two VIC-1541 disk drives (or 2031/4040 with IEEE interface), 1525 printer, and color or b&w monitor or TV.

Customer Support Plan

As part of Info-Designs ongoing effort to provide the highest quality microcomputer applications in the marketplace, we offer an optional telephone consulting service to support installation and ongoing operations.

Order NOW. . .for immediate delivery

See your local Commodore-64 Dealer or call us directly at **(313) 540-4010**. MasterCard and Visa accepted.

Info Designs

6905 Telegraph Road • Birmingham, MI 48010 • (313) 540-4010

GROW WITH US.

VIC 20™ and Commodore 64™ expansion products from Micro Systems Development.

A The Interbus Series. Three interfaces for the VIC 20 and Commodore 64: one for IEEE 488, one for RS 232 and one for Parallel.

The VIE and CIE are IEEE 488 interfaces for the VIC 20. When plugged into the expansion port, the cartridge is "transparent," that is, the user can still attach other peripherals without any interference. Devices such as 4040, 8050, 2031, 2032, 4022 and 8023 can be controlled. The IEEE software can be called by using the 'SYS' command, even in the middle of a BASIC program.

The V232 and C232 are serial interface cartridges which allow connection of various input/output devices such as printers, modems, plotters, etc. to VIC 20 or Commodore 64 computers. Features include: positive and negative voltage swings to meet full EIA standards, straps and jumpers to allow reconfiguration to meet pinouts for any RS232 device, and software selectable reconfiguration such as baud rate, parity, and duplex mode.

The VPI and CPI are parallel interfaces for the VIC 20 and Commodore 64. These interfaces provide direct BASIC use of the parallel printer bus and give "invisible" access to the bus. The VPI can be used only on the VIC 20 and uses the expansion port. The CPI will work with both the VIC 20 and Commodore 64 and does not use the expansion port. The CPI also has switches for setting insertion or deletion of line feed, conversion of Commodore ASCII into standard ASCII or visa versa, addresses printer to device 4, 5, 6 or 7, and allows normally unprintable Commodore characters to be printed in a recognizable form.

B Expandoport Series. Expandoport 3 and Expandoport 6 are three- and six-slot expansion boards for the VIC 20. Each slot on the Expandoport 6 has a switch for controlling power to that connector. The switch allows the use of cartridges which respond to the same memory space. The Expandoport 6 also has a fuse and reset switch. The fuse prevents excessive current drain from the VIC 20 and protects it from 'shorts'. The reset switch allows the user to 'Restart' the VIC 20 without turning power off. This feature allows RAM, which is located in the ROM expansion area, to be protected during 'Restart'.

Expandoport 4 is a four port expansion board for the Commodore 64. It has the same features as the Expandoport 6 and even allows for the use of varying width cartridges.

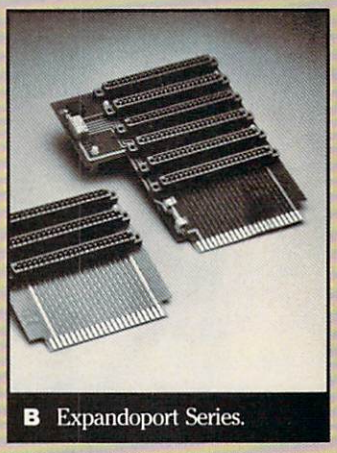
C Terminal Pak Series. The VTE 40 Terminal Emulator (VTE 40) is a hardware and software package which converts the VIC 20 into a 40-column communications terminal. The VTE 40 cartridge is complete. Various set-up parameters such as baud rate, parity, duplex, and bits per character can be selected through a 'menu' format. VTE 40 features are: 40 x 25 text display, user definable communication specs, smooth or normal scroll, print information to printer or disk, generation of control codes, selective omission of data, continuous status line.

The CTE/VTE Terminal Emulator (CTE/VTE) is a software program which converts the VIC 20 or Commodore 64 into a terminal. The user can 'software select' the baud rate compatible with the modem used. Full upper and lower case characters are supported.

D Audio Link. An audio cassette adapter interface for the VIC 20. Features include: use of regular cassette recorders, conversion of VIC 20 digital data to audio and visa versa, normal and inverted cassette signal, remote on/off control and control of external devices.

E Monitor Link. This cable assembly allows the VIC 20 and/or the Commodore 64 to interface with a monitor instead of a TV. The Monitor Link provides separate video and audio output. This enables the sound output to go directly into a stereo system for unmatched audio quality. The Monitor Link is great for applications where a TV is not desired. It allows the Commodore 64 user to have high quality resolution on black and white monitors.

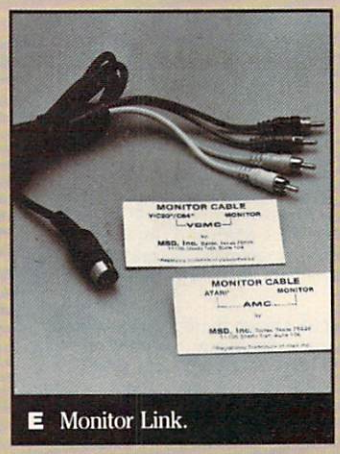
A Interbus Series.



B Expandoport Series.

C Terminal Pak Series.

D Audio Link.



E Monitor Link.

Call toll free for nearest dealer

1-800-527-5285

Lifetime warranty available upon return of Product warranty card

MSD

MICRO SYSTEMS DEVELOPMENT, INC.

11105 SHADY TRAIL • SUITE 104 • DALLAS, TEXAS 75229

FOR YOUR COMMODORE WORDPROCESSING NEEDS INVEST IN THE BEST

WORDPRO PLUS. IN A CLASS BY ITSELF.

When choosing a Word Processor for your Commodore™ computer, there's no reason to settle for anything but the best — in a word...WordPro™

With over 30,000 happy clients churning out letters and documents all over the world, the WordPro Plus™ Series is unquestionably the #1 selling software package on Commodore computers! So when you choose WordPro, you know you're investing in a trial-tested program that's a real winner. And WordPro is NOW available for your Commodore 64™ computer — at prices starting as low as $89.95.

Designed for the user who has no computer or word processing experience whatsoever, WordPro Plus brings a new dimension to the term "user-friendly." More than just easy to use, WordPro will turn your Commodore computer into a sophisticated time saving word processing tool — loaded with the same inventory of features found in systems costing much, much more.

Our nationwide team of over 600 Professional Software/Commodore computer dealers will help you choose the WordPro Plus system that is best for your needs. Our full-service dealers have been set up to provide strong customer support. In addition to helping you choose the right system, many Professional Software dealers also offer WordPro Plus training and system installation.

Professional Software offers a complete spectrum of WordPro word processing software for Commodore computers ranging from the Commodore 64 to the more business oriented 8000/9000 series computers. And WordPro 4 Plus and 5 Plus also interact with our database management systems including InfoPro and The Administrator. So whatever your Word Processing needs, there's a WordPro system that's right for you.

Invest in the best... WordPro Plus. In a class by itself.

Call us today for the name of the WordPro Plus dealer nearest you.

Professional Software Inc.

51 Fremont Street
Needham, MA 02194

(617) 444-5224
Telex: 951579

You need more than ABC, BPI, MAS, BEC, EBS, XYZ...

Workhorse solutions for tough questions.

There are lots of bookkeeping and business systems for your Commodore computer. Maybe they all seem about the same.

Ours is different: real business software for real computers, with capabilities you need, at a price you can afford.

When professional computer dealers who were dissatisfied with their current accounting software were allowed to trade it in for our System III, we were inundated. We got practically everything — BPI, EBS, MAS — just about everyone's.

We have general ledger, accounts receivable, accounts payable, payroll, inventory, mail list management and much more. There are special packages for oil accounting, church records, encumbrance accounting, pharmacy management, etc.

You will like our user-defined reports. You decide what your Balance Sheet, P&L, Budget Analysis, etc., will look like.

If you have purchased a Commodore 64*, you will be pleased when you see our complete line of business software for the '64. This software is no rinky-dink, rip-off software that is an upgrade from the VIC. These are real workhorse programs that use file structures developed for the big boys.

You have purchased the right computer when you bought Commodore. Now do it again. Buy the right software. Buy it from the right place: your professional computer dealer. That's where you'll get the help you need as you start.

Call or write for the name of the dealer nearest you.

Dealer Hotline: 1-800-527-4548

*Commodore 64 is a registered trademark of Commodore

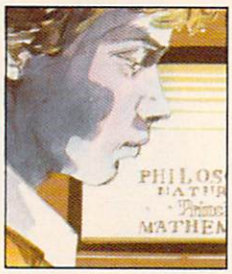
Southern Solutions

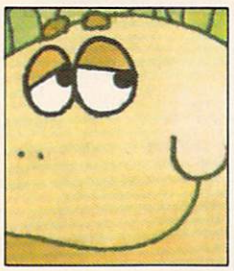
P.O. Box P, McKinney, Texas 75069 - (214) 542-0278

commodore
the microcomputer magazine

features

April/May 1983 Vol. 4, No. 2

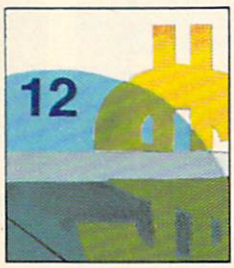




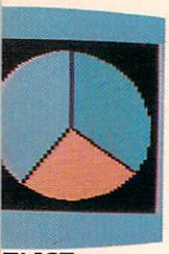
commodore
the microcomputer magazine

April/May 1983 Vol. 4, No. 2

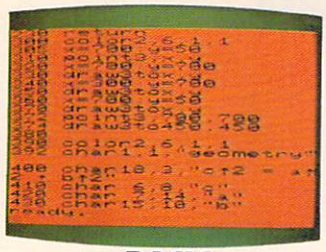
departments



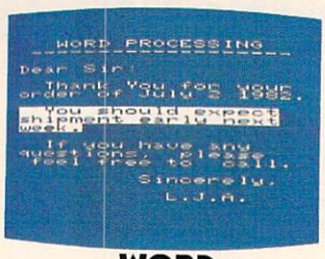
IF YOU OWN A COMMODORE VIC YOU KNOW IT CAN DO ALL THIS.



ENSE
NDAR

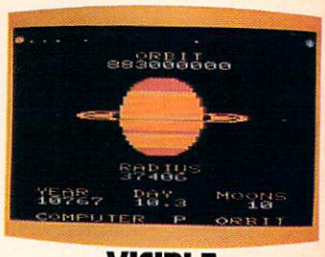


BASIC PROGRAMMING



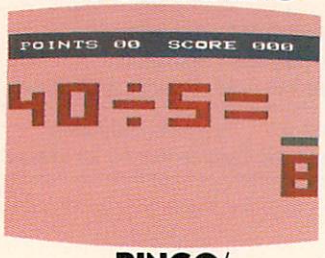
WORD PROCESSING

PERSONAL FINANCE

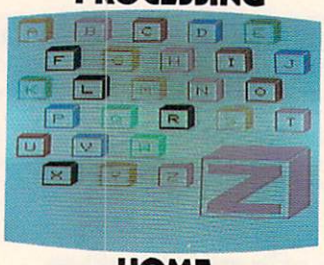


VISIBLE SOLAR SYSTEM

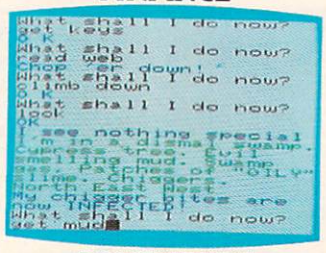
PING
TOR



BINGO/ SPEED MATH



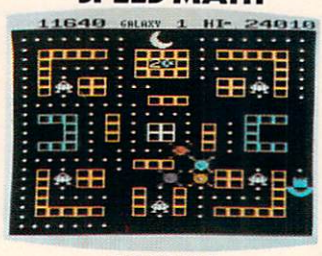
HOME BABY SITTER



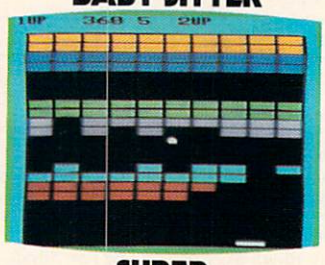
ADVENTURE LAND

BIORHYTHMS

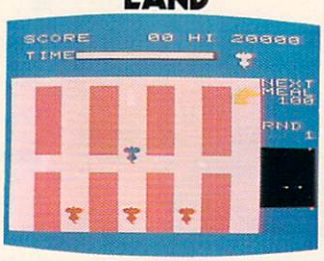
OME
NTORY



COSMIC CRUNCHER

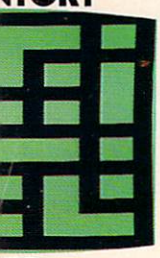


SUPER SMASH



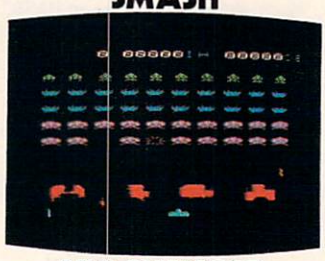
RADAR RAT RACE

JUPITER LANDER

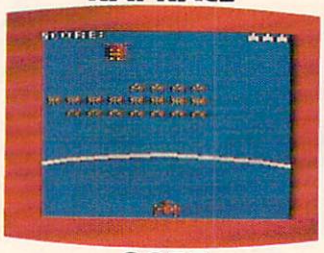


ER
EN

CHESS



VIC AVENGER



GORF

OMEGA RACE

What those extra few dollars get you is a simple little device called a Commodore VICMODEM.

It connects your telephone to your VIC 20™ or Commodore 64™ computer (resulting in something aptly called telecomputing), giving you access to information such as you see on the screens to your right.

Normally, you'd have to type a short program into your computer to help it make the final transition into a telecomputer.

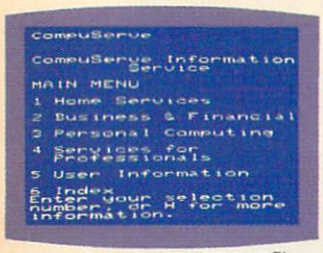
However, when you buy a VICMODEM, you'll find we've included a free software program. You just load it into your Commodore Datassette Recorder, and presto (give or take a moment or two), you have access to a vast library of information and games.

Speaking of free, Commodore also includes a free subscription and a free hour's time on CompuServe™ and Dow Jones News/ Retrieval Service,® a free trial offer on The Source,℠ and a discount program offer with Comp-U-Store and General Videotex Corp.

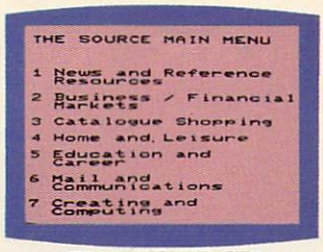
Let's see. Did we leave anything out? Oh, yes. Along with CompuServe comes a free membership in the Commodore Information Network. This is your HOTLINE to Commodore. (How often do you get to speak directly to a manufacturer?) Through it we

Certain offers subject to change. CompuServe is a trademark of CompuServe, Inc. and H.&R. Block Co. Dow Jones News/Retrieval Service is a registered trademark of Dow Jones & Co., Inc. The Source is a service mark of Source Telecomputing Corporation, a subsidiary of Reader's Digest Corporation, Inc.

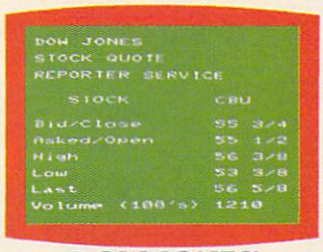
BUT DID YOU KNOW FOR ABOUT $100 YOU CAN ALSO GET IT TO DO ALL THIS?



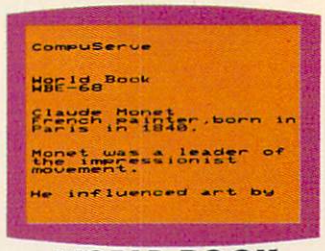
COMPU SERVE™



THE SOURCE™

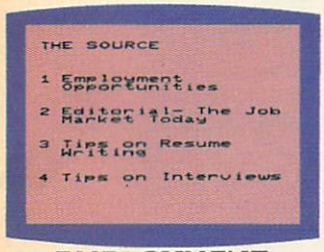


DOW JONES®

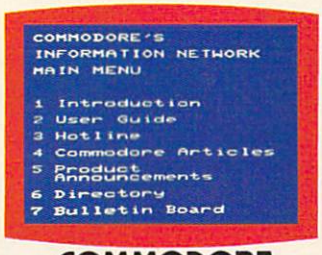


WORLD BOOK ENCYCLOPEDIA

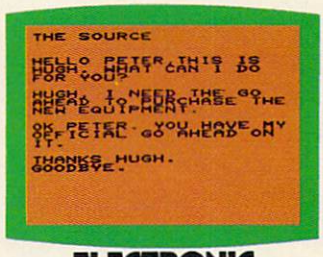
STOCK



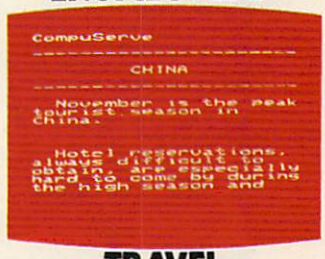
EMPLOYMENT OPPORTUNITIES



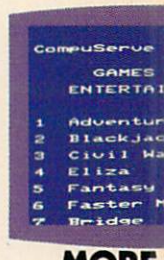
COMMODORE INFO. NETWORK



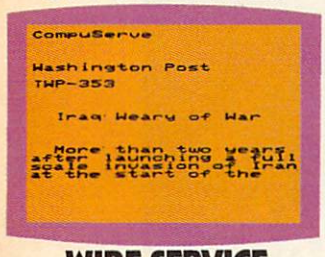
ELECTRONIC MAIL



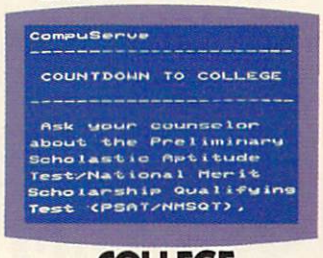
TRAVEL INFORMATION



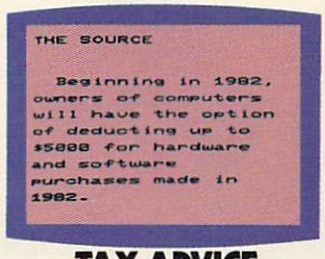
MORE



WIRE SERVICE NEWS

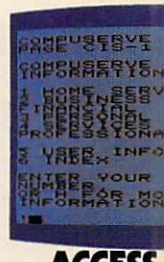


COLLEGE PLANNING



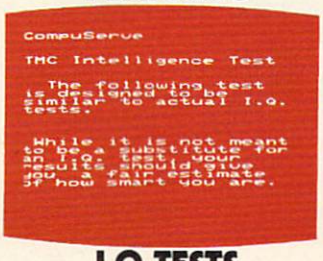
TAX ADVICE

MOVIE REVIEWS

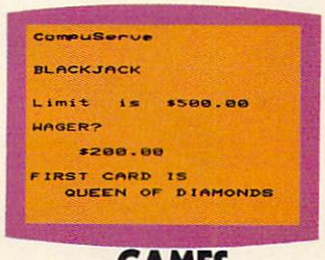


ACCESS COMP

SHOP AT HOME



I.Q. TESTS



GAMES

SPORTS NEWS

BUSINE

can answer any questions you might have about your computer, or programming, or anything else Commodore-related, via electronic mail.

The Commodore Information Network is also your direct line to the Commodore Bulletin Board, which Commodore owners use to keep in touch with each other, for programming tips, Public Domain Software, and technical support.

THE COMMODORE VIC 20, A REAL COMPUTER FOR THE PRICE OF A TOY.

Altogether, these little extras we've included with our VICMODEM add up to a value of $197.50. A nice return on an investment of around $100.

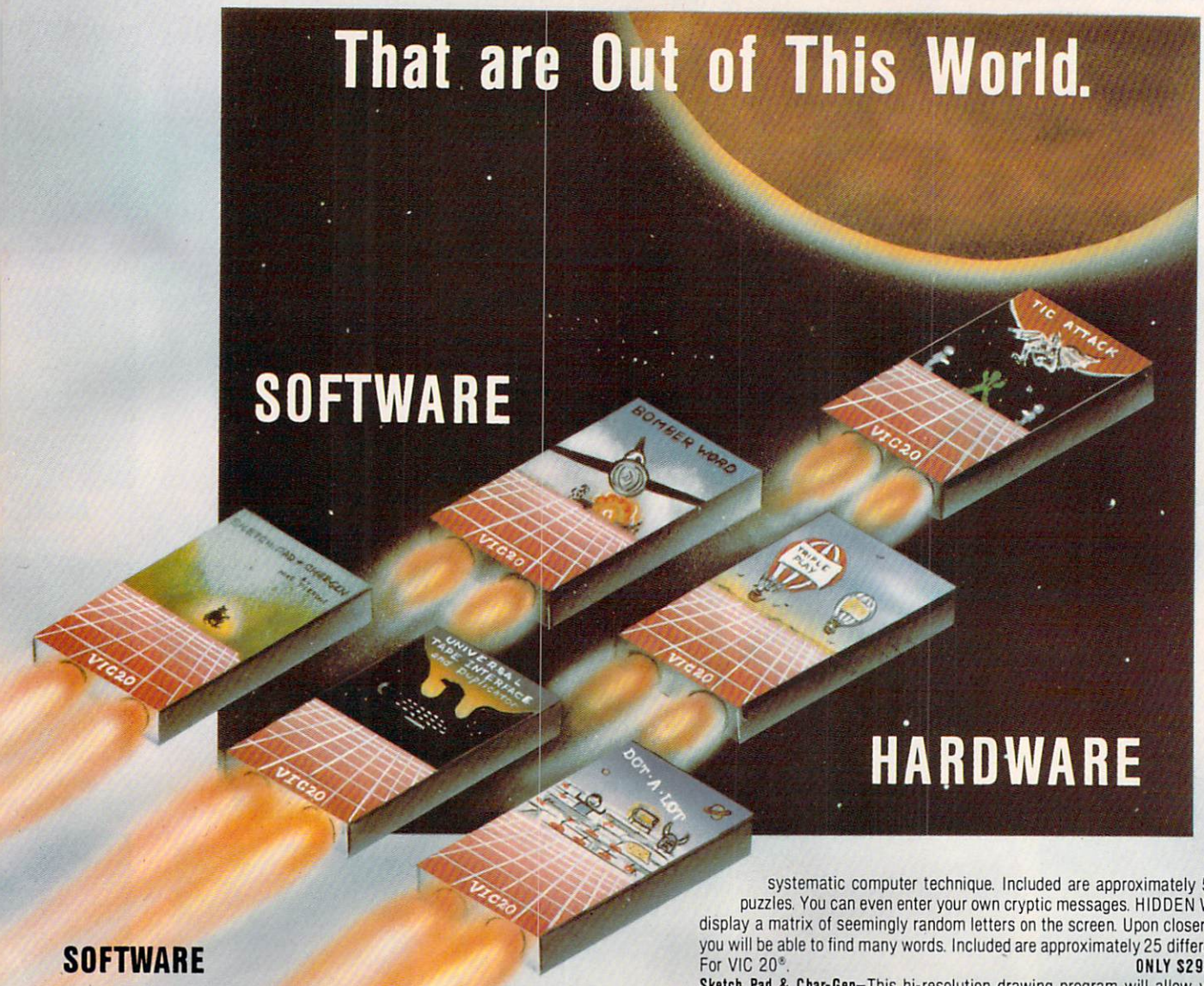
Some computer companies think it's reasonable to ask as much as $500 for telecomputing capabilities such as ours.

However, with the Commodore VICMODEM selling for around $100, we feel we're being a whole lot more reasonable. Don't you agree?

commodore COMPUTER

These are just a few examples from our existing library of software available for the VIC 20. U.S.A.–P.O. Box 500, Conshohocken, PA 19428; Canada–3370 Pharmacy Avenue, Agincourt, Ontario, Canada M1W 2K4.

Products for Commodore 64™ and VIC 20™



SOFTWARE

Word Wizard For The Vic 20®–(Requires at least 8K memory expansion) A user friendly WORD PROCESSOR with optional joystick control. Easy edit and string manipulation commands that follow the standard format. Full use of function keys for ease of use. 100% machine language with Delete Word, Search functions and Full Justification. Use VIC Graphic printer, or any centronics compatible printer connected to the user port. On Tape (supports disk). **$34.95.**

ZAP!–Climbing the corporate ladder could be fun except for all that falling paperwork. This Hires arcade type game allows up to 4 players to advance through each floor and change levels to scale the corporate ranks. Be careful, it's easy to be ZAPPED! CARTRIDGE for VIC 20.® **$29.95**

Bomber Word–A unique graphic word game on cartridge that provides the full thrill of arcade action. Complete with six modes of play options for added enjoyment. Play against the computer or another player. 6 to adult. For VIC 20®. **$29.95.**

Tic Attack–A fast action arcade game on Cartridge that challenges all of your dexterity. Written in machine language for special audio & visual effects. Over 100 levels of play. High score indication. For VIC 20®. **$29.95**

Dot-A-Lot–As you wander through the maze of life collecting Berries, you happen upon some magical fruit. Pick one and the treasures appear, but the Meanies are out today looking to spoil your fun. Defeat them and continue on to a higher level. An ever changing maze plus arcade type animation and sound will provide a real winning CARTRIDGE for the VIC 20®. **$29.95**

Triple Play–Three word games that are both fun and educational. The games that are included are CROSSWORDS (requires at least 8K expansion). Five complete puzzles are included and each puzzle has up to 100 different words. CRYPTO-SOLVE will help you solve those cryptic messages found in newspapers, books, and magazines with a systematic computer technique. Included are approximately 50 different puzzles. You can even enter your own cryptic messages. HIDDEN WORDS will display a matrix of seemingly random letters on the screen. Upon closer inspection, you will be able to find many words. Included are approximately 25 different puzzles. For VIC 20®. **ONLY $29.95 for all 3**

Sketch Pad & Char-Gen–This hi-resolution drawing program will allow you to draw pictures in detail. Use either the keyboard or optional joystick. A fill command will allow you to fill a block and other commands allow you to easily clear the screen. You can also save and load pictures. Char-Gen is a simple to use custom character generator that will allow you to design different characters for each printable key on the computer. This program is an excellent device to design game creatures, foreign alphabets, secret symbols, or other special characters. One set is included and you can make and store others quite easily. Both for VIC 20®. **ONLY $24.95**

HARDWARE

Expand-O-Ram–16K Expansion Board for the VIC 20® with reset, memory write protect, full memory allocation, plus TWO expansion slots. Like having 2 products in 1. Can even be used as a cartridge development system. **$119.00**

Universal Tape Interface & Duplicator–(Use on the CBM 64® and VIC 20®). With this device, you can easily load, save or even duplicate tapes easily with your recorder. Full 3 LED indication of Data transfer makes this the most reliable way to Load, Save and Duplicate. A complete I/0 device with extras. NOTE: Duplication requires 2 recorders. **Only $49.95**

Universal Parallel Interfaces–Now you can use most any parallel Centronics® type printer with your VIC 20®/CBM 64®. The inexpensive model will allow you to access your printer through the user port. This cable and driver is only **$19.95**. Our other model from TYMAC is more extensive with graphic capabilities. Call or write for more information and prices.

MasterCard VISA

Dealer and Distributor Inquiries Invited

NOTE: We solicit hardware and software items for the VIC 20® and CBM 64®. Royalties, license fees, or outright purchases can be negotiated. CBM 64® & VIC 20® are Registered Trademarks of Commodore Business Machines Inc.

MICRO WARE
DISTRIBUTING INC.
1342 B Rt. 23, Butler, NJ 07405
201-838-9027

staff

Publishing Manager
Neil Harris

Editor
Diane LeBold

Technical Editor
Jim Gracely

Staff Writers
Jim Bussey
Ron Cohen
Jeff Hand
April M. Koppenhaver
Dan Kunz
Stephen Murri
David Rosenwald
Michael Tomczyk
Pat Walkington

Contributing Writers
Doris Dickenson
Tom Jorgenson
Anthony J. LaMartina
Len Lindsay
Christopher Phillips
Sandra Pratscher
Ronald E. Randolph
John B. Roberts
Lee Silvan
John Stockman
John Taylor
Bill Trice

Technical Staff
Paul Goheen
Pat McAllister
Cyndie Merten
David Middleton
Tom Rizol, Jr.
David Scott

Circulation Manager
John O'Brien

Circulation Assistant
Kathy Reigel

Advertising Coordinator
Sharon Steinhofer

Graphic Design
Neumann Greenberg Schlenker

Cover Art
Mike Adams

Printing
Oak Lane Printing

Commodore—The Microcomputer Magazine *is published bi-monthly by Commodore Business Machines, Inc., 1200 Wilson Drive, West Chester, PA 19380. Phone: (215) 687-9750. Domestic Subscriptions: six issues, $15.00. POSTMASTER: Send address changes to* Commodore—The Microcomputer Magazine, *1200 Wilson Drive, West Chester, PA 19380.*

Advertising Policy

Advertisements within this magazine are presented for the information of our readers. Acceptance of any ad does not constitute an endorsement by Commodore Business Machines. We stand by only those products manufactured by us. In addition, products displaying the "Commodore Approved" logo have been tested by Commodore.
If any product advertised herein fails to perform as advertised, or you are unable to resolve a problem with an advertiser, please bring the matter to our attention. Write to Commodore Publishing Group, 1200 Wilson Drive, West Chester, PA 19380.

Watch for these upcoming issues!

Commodore, June/July: Our Programming Special.

Power/Play, Summer: More new games for the Commodore 64: more music and graphics programs.

letters

Pets and PETS

We all know very young children can learn to use computers. But did you think computers would become so user-friendly even our little furry pals would soon be programming? Take Cedric, the pet of Mrs. RuthAnn Pickens of Glen Cove, New York, who seems to be about to start typing in a program on Mrs. Pickens' other PET. He's undoubtedly created a new version of Radar Rat Race—where the cats always win.

In addition to sending us this photo of Cedric, Mrs. Pickens writes:

Gentlemen,

I am a teacher at Sacred Heart School in N. Merrick, New York, and I bought my own computer as an aid in preparing programs for my students. I am thoroughly convinced that excellence in education is through technology and I know that working with computers will better prepare my eighth-grade students for their future role in society.

We think so, too, Mrs. Pickens, and this special education issue is a testimony to that fact. Ed. **C**

WHAT'S SMALLER THAN A BUSINESS CARD?

FASTER THAN CASSETTES?

AND FAR LESS EXPENSIVE THAN DISKS?

Why the ESF-20/64 Stringy Floppy from Exatron, of course. Our exciting little storage alternative gives you near floppy disk speed and reliability at a budget-minded price. Our high quality digital mass storage system is the perfect product to fill the gap between cassettes and floppy disks. And that's especially true in the case of the Commodore VIC-20® and Commodore 64® microcomputers.

The Exatron Stringy Floppy system is based on a small endless loop tape cartridge we call "The Wafer". This wafer measures only 2 3/4" x 1" x 3/16" — or about the dimensions of a standard business card. Wafer was born to run fast — at around 7200 baud, or 14 times faster than a standard cassette and has a memory capacity of 64K bytes. The ESF-20/64 system costs less than $200 and wafers are less than $3.00 each. No wonder computer people call us the "poor man's disk system".

If you'd like to get the world on a string, and bring your VIC-20® or Commodore 64® into the 20th Century, just fill out the coupon below and mail it to EXATRON, 181 Commercial Street, Sunnyvale, CA 94086. If you need to place an order call (408) 737-7111, outside California (800) 538-8559 and ask for Chrissy or Natalie.

The ESF-20/64 Stringy Floppy System from Exatron for use with the Commodore VIC-20® and Commodore 64® microcomputer.

exatron

For exciting alternatives in mass storage.

VIC-20 AND COMMODORE 64 are trademarks of commodore electronic ltd.

Name _____

Address _____

City/State/Zip _____

Mail today for information on the ESF-20/64 from EXATRON, 181 Commercial St., Sunnyvale, CA 94086

Educators, Kids and Computers

Diane LeBold

Educators are going to love this issue, whether they are just beginning to consider computers or are looking for new and better ways to use the computers they already have. Because educators' experience with computers varies widely, we've provided a broad range of information, from a simple computer glossary for the sake of those who are brand new to the field to a discussion of the PILOT language for more experienced educational users. We put the issue together with the untiring help of our Education Marketing Department, who, in their great enthusiasm, gave us enough material to create two or three issues. That's because they know educators ask a lot of hard questions, all of which they wanted to answer.

We haven't answered all those questions, to be sure. But we have created a solid source of information we think will serve as a valuable reference guide in the upcoming year. The long lists of resources we've included here may look boring, but believe us, some day before this year is out you'll be glad you have this issue on hand—or regret not having it, whichever may be the case.

If you're not an educator yourself, this issue is still worth your time—because chances are good you're involved with education somewhere in your life, whether it's your own education or your children's. If your school district, for instance, doesn't have computers or isn't using them to their full capacity, get over to a school board meeting with this issue and show them what they're missing—and how to get it. So many schools think they can't afford computers, for example, but perhaps don't realize that block grant funding is available for just such projects. If you need more copies of the magazine in order to make your point, let our Education Marketing Department know.

When we talk about computer education there's one important thing we have to remember—there are real live kids at the other end of all our talk. Since I have a ten year-old daughter and therefore have a vested interest in education myself, I'm going to get philosophical for a minute about those kids and their computers.

When we talk about computers in education, I'll bet more than half of you think immediately of what we call CAI (Computer Assisted Instruction—usually in the form of drill-and-practice). Right? That's because CAI is probably the most typical way schools use computers, right now.

CAI is certainly a very useful method for teaching things—from simple arithmetic to complex college chemistry. Students seem to enjoy interacting with CAI programs and, what's more, they seem to retain the information they learn. But there is much more to computing than simply running programs—CAI programs or otherwise. That's the point Seymour Papert, the father of the LOGO language, makes so adamantly in his book *Mindstorms: Children, Computers and Powerful Ideas*. What it comes down to, in Papert's estimation, is whether we want kids to program computers—or computers to program kids. He, of course, opts strictly for the former.

I can see a great deal of value in using CAI, but, to be honest, I like to see my daughter take hold of her VIC 20 and teach it to turn somersaults and jump through hoops, rather than vice versa. One method I've discovered of getting

editor's notes

her to do this is with a language called turtle graphics. Turtle is an offshoot of LOGO, so if you read our LOGO article on page 43, you'll get the gist. With a language like turtle—or LOGO—or any language, for that matter—kids learn to think a problem through systematically and at the same time have a great tool for creative expression. Inevitably one of those kids will become the Shakespeare (or Rembrandt or Bach) of the computer medium.

Both CAI and programming courses have a place in a well-planned educational computing curriculum. In addition, students should be able to learn business computing—the wonders of word processing and data base management, for instance—and teachers and administrators should take advantage of the computer's doggedly reliable record-keeping capabilities to free themselves for more productive work.

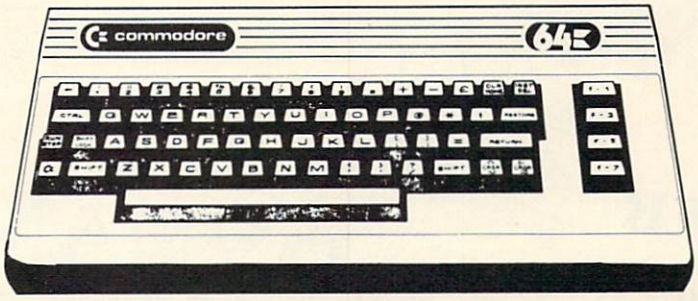
Just because people drive cars doesn't mean they will all become great mechanics. Likewise, just because people use computers—which will soon be practically as common as driving cars—doesn't mean they will become great programmers. Nevertheless it certainly helps, if you want to drive a car, to have an understanding of how to do essential things like check the oil, get gasoline or change a flat tire. And it helps even more if you know how to do simple maintenance like change the oil or tune up the engine. The same goes for using computers. If you're using them regularly it helps to understand the basics of how they work and how to get them to do what you want. That's where computer education can be most valuable, and where educators, it seems to me, can make the greatest contribution to their students.

BUSICALC

BUSICALC A Honey of an Electronic Spreadsheet

Why electronic spreadsheet programs?

Electronic spreadsheet programs allow the user to create a gridsheet, spreadsheet, worksheet, or any other table of information, using the memory of the computer as pencil and paper. The computer display or terminal acts as a window through which the user views the information as it is entered. Textual information (such as headings), numerical values, and formulas can easily be entered into the spreadsheet.



For Commodore 64™
For Commodore VIC 20™
For Commodore PET®/CBM™ 40 columns
For Commodore CBM 80 column/SuperPet™

BUSICALC Your Computer Drone for Repetitive Calculations

The outstanding advantage of using a computer is that it acts not only as a pencil and paper but as a perfect eraser and an automatic calculator. The user can quickly and easily make any number of alterations to the data within the table. The BUSICALC will evaluate any formula using the data that has been entered. Further, it retains the formulas and displays the resulting value. With BUSICALC controlling the entry of data, providing a comprehensive memory, and performing arithmetic, the preparation of a spreadsheet is faster and more accurate than if it were prepared by hand.

BUSICALC With the Sting Removed from the Prices

BUSICALC 20. . . . . only **$49.00** for the VIC 20
BUSICALC 64. . . . . only **$69.00** for the CBM 64
BUSICALC 40. . . . . only **$79.00** for the original 40 column PET/CBM
BUSICALC 80. . . . . only **$89.00** for the original 80 column CBMs and SuperPets

BUSICALC AVAILABLE NOW FROM YOUR LOCAL DEALER
(800) 227-9998
FOR THE NAME OF YOUR NEAREST DEALER

California, Canada, Alaska and Hawaii please call (415) 965-1735

Skyles Electric Works
231G South Whisman Road
Mountain View, CA 94041

Europe please contact Supersoft, Winchester House, Canning Road, Harrow Wealdstone, Middlesex, England HA3 7SJ, Tel. 01 861 1166

VIC 20 is a trademark of Commodore Electronics Ltd.

ORDERING BLANK

To: American Peripherals
122 Bangor Street
Lindenhurst, N.Y. 11757

Ship to: Name _____

Street _____

Town, State, Zip _____

☐ *Check here for latest 64 Catalog*

☐ *Check here for latest VIC Catalog*

ITEM	DESCRIPTION	PRICE
	NY State Residents only add 7¼% tax	
	Shipping (If COD, add $1.50)	$1.50
	TOTAL AMOUNT	

U.S. Funds - Foriegn Countries add an additional $3.00

Commodore Sponsors Olympics of the Mind World Finals

A brand new event will take place at the World Finals of the Olympics of the Mind this year. For the first time since its inception in 1978, the Olympics will feature a computer problem, which will be designed and sponsored by Commodore.

The Olympics of the Mind is a yearly competition for students from kindergarten through twelfth grade. The purpose of the competition is to foster the development of students' creative and intellectual abilities. Each year teams of students from over 2,000 schools solve unique problems in engineering, architecture, communications and the arts, participating in competitions at the county, regional, state and finally world levels.

This year the World Finals will be held at Central Michigan University, Mount Pleasant, Michigan, on May 26 and 27. Commodore will provide both technical assistance to help develop the computer problem and the use of computers for testing the students' solutions. The results of the computer event will then be used to design a new problem for use in next year's Olympics of the Mind.

The founders of the Olympics, New Jersey educators Theodore J. Gourley and C. Samuel Micklus, are excited about the addition of a computer problem to the competition.

"We have had requests for a computer problem," Micklus said. "It is a growing area of interest in schools, especially among the gifted students who are often involved in the Olympics of the Mind."

In 1978 Gourley and Micklus began the Olympics, thinking it was a one-time event, in response to a challenge from the United States Department of Education. The challenge was to design a creative intellectual competition among students similar to a sports competition. Their first Olympics of the Mind was held at Glassboro State College in New Jersey and involved about 300 students. It was such a success the idea took hold and now over 40,000 students throughout the United States and Canada participate in the program annually.

The problems posed in the competition are designed to excite students by presenting real challenges that force them to think originally and creatively. For instance, students have been required to design a robot that would do all of the following: 1) move six feet across the floor, 2) remove a cigarette from a dummy's mouth, 3) put the cigarette in a can, 4) pour water on the cigarette, 5) speak to the dummy and 6) proceed across the floor to a predetermined area—all for under $10.00.

Other problems required students to remove wooden fish from a "pond" without using a fishing pole, line or hook; develop a band that does not use musical instruments to play tunes; create a vehicle powered by five pounds of sand; and create a structure from ½ ounce of balsa wood that would support as much weight as possible.

Interestingly, student solutions can sometimes exceed the expectations of the competition. For example, the sand-powered vehicle that won the state competition in New Jersey one year traveled further than the track set up for the event. And students designed balsa wood structures that supported hundreds of pounds. The possibilities posed by the new computer event, needless to say, are of great interest to computer enthusiasts both in and out of the schools.

If your school would like more information about participating in the Olympics of the Mind, contact Dr. Gourley or Dr. Micklus at:

Olympics of the Mind Association
P.O. Box 27
Glassboro, NJ 08028

500% Jump in Number of Commodore School Districts

While Commodore has made significant gains in all markets, we are particularly proud of the gains made in education. During the last year the number of school districts that use Commodore microcomputers increased 500% according to an independent study.

The PET line has been widely accepted by educators because of its reliability, friendliness and unusually low cost. With the introduction of the Commodore 64 a whole new world of computing power is opening to schools. In addition a low cost modem that allows access to major data bases and a local networking capability that will soon be introduced further assist educators in getting the most computing for their money.

Commodore expects to surpass this already outstanding usage increase during the upcoming school year.

Commodore Introduces Voice Synthesizer

Commodore has announced a sophisticated new voice synthesizer for the Commodore 64 home computer. This is the first voice I/O product to be developed at the company's Speech Technology Division in Dallas, Texas.

The low-cost speech add-on is capable of generating a wide variety of voices—including female and children's voices—for games and learning cartridges, and will be usable from the Commodore BASIC programming language.

The voice peripheral has three modes of operation, two of which are immediately available with the basic module, and a third with the purchase of optional cartridges.

The Commodore voice module can be used as soon as it is inserted into the cartridge port of the Commodore 64. When the Commodore 64 is powered up with the voice module, the user may create speech through simple BASIC commands, such as:

SAY "A", "B", "C" or

```
10  SAY   "ENTER YOUR NAME"
20  INPUT B$
30  SAY   "THANK YOU"
```

This short program instructs the computer to speak the name typed in from the keyboard.

Besides the capabilities added to the BASIC language, the module features a built-in learning activity called "A Bee C's"™, an instructional program that

commodore news

uses speech to teach the alphabet to preschoolers.

The most exciting feature of the voice module, however, is its capability to integrate voice into games and learning cartridges. Commodore is now developing cartridges that will offer the widest variety of exciting games and instructional material for home use. The voice synthesizer's ability to accept different vocabularies and different characters' voices means Commodore computer owners will be able to choose the type of voice (male, female, child's, cartoon character, etc.) used with various programs.

This flexibility is achieved through a special technique which allows speech to be generated while the computer's microprocessor is performing other functions such as graphics/cartoon animation. For example, the technique allows the computer to put an animated face on the screen simultaneously while speech is being generated.

Pricing for the speech synthesizer has not been announced but the product is expected to retail for under $100, with delivery scheduled for Spring 1983.

Commodore Introduces $300 Color Monitor

Commodore has introduced a low-priced color monitor especially designed for its line of home, school and business computers. The new 13-inch monitor—designated the CBM 1701—retails for $299.95 and is especially designed for use with the Commodore 64 and VIC 20 computers. The monitor accepts a standard 75 ohm composite video signal or a Commodore video signal, with separate provisions for luminance and chrominance signal input and audio input.

The monitor was developed in conjunction with a major television manufacturer and includes special circuitry that greatly enhances video resolution on the Commodore 64 and VIC 20. It was designed to meet the needs of Commodore computer owners, who need an affordable dedicated monitor to prevent the inconvenience of having to keep connecting and disconnecting their computer from their family television set.

According to Robert Lane, Commodore's president in charge of North American operations, a color monitor is a key component in any color computer system, and in 1983 most computer retailers will begin to sell computer systems just like stereo component systems.

Commodore Digi-Drum™ for the Commodore 64 and VIC 20

Imagine an electronic drumset controlled by a computer, with volume control and sound generated through a TV set, monitor or stereo system—but priced as low as a video game cartridge!

Commodore has introduced just such a low-priced three-pad electronic drumset, called Digi-Drum, for the Commodore 64 and VIC 20 computers.

The new peripheral plugs into the computer's expansion port and comes complete with special software which lets the user simulate a snare drum, bass drum and "high hat" cymbal, with startling realism.

The combination of computer and Digi-Drum visually displays three animated drums on a television screen, which "perform" with each drumstroke. The three drums can be combined in an infinite variety to produce high quality rhythm effects suitable for entertainment, learning or sound effects.

Digi-Drummers can use their VIC 20 or Commodore 64 computer to create and save drum routines and play them back through their stereo system or television speaker.

First delivery of Digi-Drum is scheduled for Spring/Summer 1983.

Commodore and Zilog Announce Licensing Agreement

Commodore International has signed a definitive licensing agreement with Zilog,

Inc., an affiliate of Exxon Corporation, to begin manufacturing Zilog's Z8000 16-bit microprocessor and related family of peripheral support circuits for use in Commodore microcomputer systems. In exchange, Commodore will supply Zilog with manufacturing rights to selected custom circuits used in current and future Commodore microcomputer systems.

According to Irving Gould, Chairman of the Board of Commodore International, the agreement with Zilog will give Commodore a substantial increase in semiconductor manufacturing capacity while reducing the time and investment that would be required for Commodore to expand its own semiconductor manufacturing. C

The Commodore 64 CP/M* Cartridge:

Extending the Capabilities of the Commodore 64

by
Tom Jorgenson, Software Wizardry, Inc.

A report from one of Commodore's CP/M beta test sites.

Possibly the most exciting feature of the Commodore 64 is its ability to operate with what has often been described as THE standard operating system of the microcomputer world—CP/M. I have been using CP/M on several larger machines for several years now, and am quite excited by the Commodore 64 implementation. Although a large number of CP/M systems already in existence are larger than the 64's, no other computer brings CP/M within the reach of so many people at so little cost.

What is CP/M?

These days most home computers structure their operating system around BASIC, since its commands are generally the easiest for a beginner to learn. The BASIC language is, however, a double-edged sword. What we gain in simplicity, we lose in speed and versatility. BASIC runs interpretively, that is, each instruction is "translated" into binary code (the computer's own language) as it is encountered. This is considerably slower than using a machine code monitor, in which instructions have already been converted (i.e., compiled) into binary code.

BASIC is also not always the best way to solve a problem. As good as it can be as a general purpose math processor and string handler, it can sometimes lead to programs that contain hidden traps. In addition, BASIC, like all high-level languages, tends to hog memory. And it cannot directly handle certain capabilities of the computer, which is why so many BASIC programs contain machine language subroutines.

On the other hand, operating systems with machine language monitors—like CP/M—are generally faster and more versatile and use less memory than interpretive languages like BASIC. But the best feature of CP/M on any system is its universal nature. CP/M looks and behaves essentially the same on every computer on which it finds itself. For that reason a user can usually expect that a CP/M program developed on totally different computer hardware will run on his own system with little or no modification.

This independence from system hardware has been a real boon to computer users and software developers alike. No greater market exists for microcomputer software than for CP/M. For this reason, software developers have naturally concentrated on producing programs for CP/M above others. Consequently, an extremely large and powerful base of programs now exists for this operating system—everything from games to powerful business database systems.

In addition, the CP/M community has a number of sources of public domain (i.e., free) software (see Table I). Dial-up bulletin boards around the country as well as telecommunications systems such as CompuServe Consumer Information Service have become major distribution nodes for such software. What a single user donates to the public domain for one computer system, therefore becomes a building block for other systems as well. At last count there were over 80 volumes of public domain software available through CP/MUG (the national CP/M Users Group) alone.

CP/M is able to run on any computer with little or no

modification because it is built from three basic modules (see Table II): the CCP (what the user communicates with), the BDOS (the module that builds sectors into files) and the BIOS (the hardware-dependent portion). The first two modules are identical in every standard of CP/M implementation. Only the BIOS differs.

Implementation of CP/M on the Commodore 64

CP/M on the Commodore 64 comes as a plug-in cartridge with documentation and a system diskette. Unlike most cartridges, which are simply ROM (Read Only Memory) packs, this cartridge contains a substitute processor for the system—a Z80 chip. It is necessary to bring in a new processor because standard CP/M can run only on an 8080 or Z80 microprocessor and the Commodore 64's built-in microprocessor is a 6510.

With the CP/M cartridge installed in the Commodore 64, the system still powers up the same as always. Only when a CP/M system disk is loaded in the disk drive does the Z80 take control. Commodore's 6510 processor is still available, and, in fact, performs all input/output for CP/M. As a result, virtually all the normal Commodore capabilities are available on the system (although not always as readily) and compatibility can be maintained with most common peripherals.

A BIOS (hardware) function called by the system will cause the Z80 to set up a "communication area" in RAM (Random Access Memory) to tell the 6510 which functions to perform and how to perform them. The 6510 will then temporarily take over the system. During the time the 6510 is active the Z80 will patiently remain in a suspended state until control is returned to it via the communication area.

Advantages Over the Other CP/M Implementations

When you compare it to others, two features stand out in Commodore's implementation of CP/M: user-oriented documentation and easy installation.

When CP/M was developed by Digital Research, the standard manuals were written more with software developers and system OEM's (Original Equipment Manufacturers) in mind than average users. In most cases the suppliers of the CP/M operating system have been including this Digital Research manual along with their own (usually crude) leaflets as documentation for end users. As we all know, hardware manufacturers usually do not make good manual writers. Put another way, if I had a nickel for every person we lost to CP/M documentation, I'd probably own this magazine rather than write for it.

Commodore, however, has done a superb job of completely rewriting this manual. The regular Digital Research notes, as well as many other resources, are referenced within the manual for those who insist on learning the hard way. But the beginner is spared the greatest part of the effort.

As for installation, there is almost nothing you can do wrong on the system. On many systems it's relatively easy to connect the wrong piece at the wrong time. But CBM intelligently keyed the cartridges and peripherals to save the user from himself. About the only injurious action you can take is to remove the cartridge—or insert it—when power is on. Even here Commodore has taken the precaution of warning the user repeatedly about the consequences of such an action.

One advantage I hardly need to mention here may outweigh all others in the minds of many people. An average CP/M system typically costs about $3000. But even including the cost of a Commodore 64 the price is far less for the Commodore implementation. Someone who already owns a 64 will, obviously, spend FAR less, since he will be paying only for the cartridge and software.

Limitations of the System

As good as the Commodore 64 implementation is, it wouldn't be quite fair to write a review without finding fault with some aspect of the system. Considering the cost and advantages of this system I don't think there are any limitations that are of major concern to the average user. But I would like to point out what you should consider if you are purchasing a 64 specifically to use CP/M in business or intensive applications.

As I mentioned above, the 6510 processor remains operational within the system while it is running CP/M. The CP/M memory map is therefore limited to a maximum of 48K with a serial disk drive or 40K with an IEEE-488 drive. Most CP/M programs have no problem executing within this memory size, but there are some business programs and other large packages that require 56K. You should, therefore, be sure to ask your software supplier how much memory is required for a utility before you invest in it, just in case.

Also, the Commodore system with which I used CP/M had a serial disk drive. On this type of system the drives are somewhat slower than those on the larger (high-speed) systems I've used in the past. However, I understand IEEE-488 drives are several orders faster than this drive, so this may be a welcome addition if the major system you plan to use will be CP/M. Again, serious business users should consider things like disk storage and system speed before planning to use the Commodore 64's CP/M capability.

These limitations, however, should not be of major conse-

quence to many users. And they are actually quite a small price to pay for the power and value of the Commodore 64 CP/M system. The fact is few, if any, CP/M implementations exist without one or more of these limitations—and there are many others Commodore has avoided entirely. C

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

Table I
Sources of Public Domain CP/M Software

The CP/M Users' Group (CP/MUG)

CP/MUG is probably the largest single source of public domain software in the CP/M community. At last count they had somewhat over 80 volumes of software on 8" disk format (only). They also publish a monthly magazine called *Lifelines*. For more information, write:

CP/MUG
1651 Third Avenue
New York, NY 10028

CompuServe Consumer Information System

A timeshare system with nodes all around the country. On this system you will find both a Commodore special interest group and a CP/M interest group. More information can be obtained by writing or calling:

CompuServe Consumer Information System
Personal Computing Division
5000 Arlington Boulevard
Columbus, OH 43220
(614) 457-8600

Remote CP/M Systems

RCPM's are public dial-up bulletin board systems around the country which allow you to access their files by modem for the cost of a phone call (no charge for their use). Many of these systems now have Winchester disks connected to them and the volume of software to be obtained on these is nothing less than monstrous! One such system (this one is a bulletin board only) where you can obtain information on the currently active RCPM's is:

Ward & Randy's
(312) 545-8086
(8 bits, 1 stop bit, no parity, full-duplex)

Table II
Organization of CP/M in Memory

	Highest available memory
BIOS Basic I/O System (JMP table at bottom)	The BIOS contains a series of subroutine calls accessible via a jump table at its start. All hardware-dependent code is intended to be in this area.
BDOS Basic Disk Operating System	CP/M's BDOS area takes file-oriented requests from a user program and determines which sectors are allocated to those files. It also handles devices independently of the hardware.
CCP Console Command Processor	This portion of CP/M converts user functions (like DIR, TYPE, SAVE, etc.) into BDOS calls and performs them. The CCP can be overlayed by a transient and restored by warm boot process.
TPA Transient Program Area Always starts @ 100 hex	TPA stands for Transient Program Area, and is the load area for user programs. The CCP can be thought of as part of the TPA, actually, since it can also be used in this manner as well.
Reserved Area for System Default Buffers, etc.	CP/M reserves this area for a number of purposes. 8080/Z80 interrupt vectors are located here as well as CP/M default buffers. At the bottom of memory are jumps to the BIOS and BDOS.

Complex Rhythms
Part 2

by Lee Silvan

When we stopped last issue we were exploring the COMMODORE 64's ability to render to the composer/programmer accurate performances of rhythms that humans find extremely difficult or impossible. The musical concept of dividing four notes evenly against five is where we stopped with a promise of some interesting code for playing these rhythmic combinations. Well, here is what I think is very interesting code. The program that follows was developed by a very talented Commodore programmer, Cyndie Merten, and it allows us to do some very complex things. Listing 1 has been formatted to be easy on the eyes, but remember that all those extra spaces are not necessary when keying in the program itself. Also, it is a good idea to save the program in this form (without the necessary DATA statements) since it can be used with many different musical data sets.

Listing 1

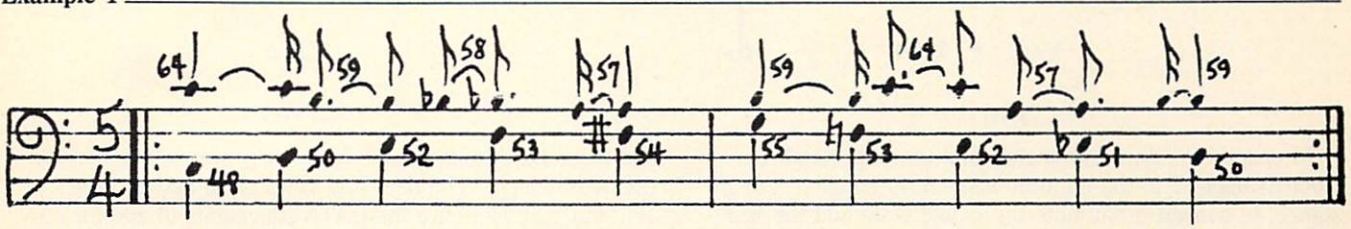
```
10 S=54272:FORL=STOS+24:POKEL,0:NEXT          (Initialize SID chip and clear all registers)
20 DIMH(2,200),L(2,200),C(2,200)              (Dimension 3 activity arrays @ 1/16 measure per location)
30 DIMFQ(11)                                  (Dimension array holding the top octave frequencies)
40 V(0)=17:V(1)=65:V(2)=33                    (Set waveform control byte for each voice)
50 POKES+10,3:POKES+22,240:POKES+23,          (Set pulse width for voice 2; set high freq. filter cutoff; set
   244                                          filter resonance & filter voice 3)
60 FORI=0TO11:READFQ(I):NEXT                  (Read in top-octave frequencies for each note name)
100 FORK=0TO2                                 (Begin the decoding loop for each voice)
110 I=0                                       (Initialize activity array pointer)
120 READNM                                    (Read a coded note)
130 IFNM=0THEN250                             (If the coded note is 0, then do next voice)
140 WA=V(K):WB=WA-1:IFNM<0THENNM=-NM:WA=      (Set waveform control to proper voice, silence=0)
    0:WB=0
150 DR%=NM/128:OC%=(NMAND112)/16              (Decode duration and octave)
160 NT=NMAND15                                (Decode note name)
170 FR=FQ(NT)                                 (Get top octave frequency for this note)
175 IFK=0THENOC%=OC%+1                        (Raise voice 0 one octave above where written)
180 IFOC%=7THEN200                            (If highest octave, then skip division loop)
190 FORJ=6TOOC%STEP-1:FR=FR/2:NEXT            (Divide top octave frequency appropriate number of times)
200 HF%=FR/256:LF%=FRAND255                   (Decode high and low frequency bytes)
210 IFDR%=1THENH(K,I)=HF%:L(K,I)=LF%:C(K      (If 16th note then set activity array: high frequency, low fre-
    ,I)=WA:I=I+1:GOTO120                        quency, and waveform control voice on)
220 FORJ=1TODR%-1:H(K,I)=HF%:L(K,I)=LF%:      (For all but last 16th of the note length, set activity array:
    C(K,I)=WA:I=I+1:NEXT                        high frequency, low frequency, waveform control on)
230 H(K,I)=HF%:L(K,I)=LF%:C(K,I)=WB           (For last 16th of note length set waveform control OFF!)
240 I=I+1:GOTO120                             (Increment activity array pointer, begin process again)
250 IFI>IMTHENIM=I                            (If next note is longer, reset number of activities)
260 NEXT                                      (Begin decoding the next voice)
500 POKES+5,63:POKES+6,240                    (Set Attack/Decay, then Sustain/Release for voice 0)
510 POKES+12,9:POKES+13,9                     (Set Attack/Decay, then Sustain/Release for voice 1)
520 POKES+19,119:POKES+20,7                   (Set Attack/Decay, then Sustain/Release for voice 2)
530 POKES+24,31                               (Set volume to max, with low pass filtering)
```

```
540 FORI=0TOIM                          (Start loop for each 16th note equivalency in measure)
550 POKES,L(0,I):POKES+7,L(1,I):POKES+14   (POKE low frequency from activity array for all voices)
    ,L(2,I)
560 POKES+1,H(0,I):POKES+8,H(1,I):POKES+   (POKE high frequency from activity array for all voices)
    15,H(2,I)
570 POKES+4,C(0,I):POKES+11,C(1,I):POKES   (POKE waveform control from array for all three voices)
    +18,C(2,I)
580 FOR T=1 TO 40 :NEXT                 (Timing loop for 1/16 of a measure)
585 NEXT                                (Begin next 16th note portion of measure)
590 GOTO540                             (Begin selection again / or change line to END)
600 DATA 34334,36376,38539,40830        (Top
610 DATA 43258,45830,48556,51443        octave
620 DATA 54502,57743,61176,64814        data )
```

If you try RUNning the program now, you will get an "out of data error" in line 120. Now we will add the data for the following musical example. There are two short melodic lines that move at different rates of speed. More precisely, the top melody plays four notes, evenly spaced in time, against five notes, also evenly spaced, in the lower part. The musical notation that follows is the most accurate way to notate this.

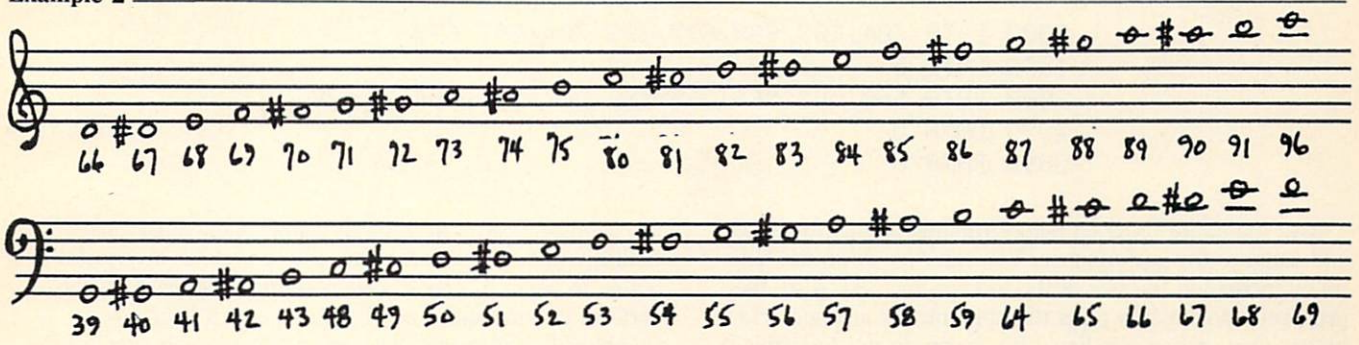
Incidentally, such rhythmic combinations are not unusual in modern music. Charles Ives demanded rhythmic coordination far more complex than this over sixty years ago! (Ol' Charles would have just loved the COMMODORE 64!) Today these "cross-rhythms", as they are called, are common to modern jazz as well as the classical avant-garde.

Example 1



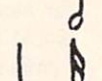
To code this music, programmer Merten found a way to get all necessary pitch and duration information into one integer. Lines 150-200 contain the algorithm which extracts high and low frequency bytes as well as note value. Here is a portion of the 64's pitch spectrum with each note assigned a number.

Example 2



If you check the musical example you will find the appropriate "pitch number" next to each note. The durational values are determined through use of the following lookup table:

Example 3

Note Value	Note Symbol	Duration
1/16		128
1/8		256
1/8 + 1/16		384
1/4		512
1/4 + 1/16		640
1/4 + 1/8		768
1/2		1024
1/2 + 1/16		1152
1/2 + 1/8		1280
1/2 + 1/4		1536
whole		2048

To encode the music we merely add the note number from Example 2 to the duration value in Example 3. For instance, to convert a half note on middle C we add the half note value of 1024 to the note number for middle C (64) and place the resulting 1088 in the appropriate place in the DATA statement. The decoding takes place between lines 160 and 200. Here are the DATA statements to add for our musical example.

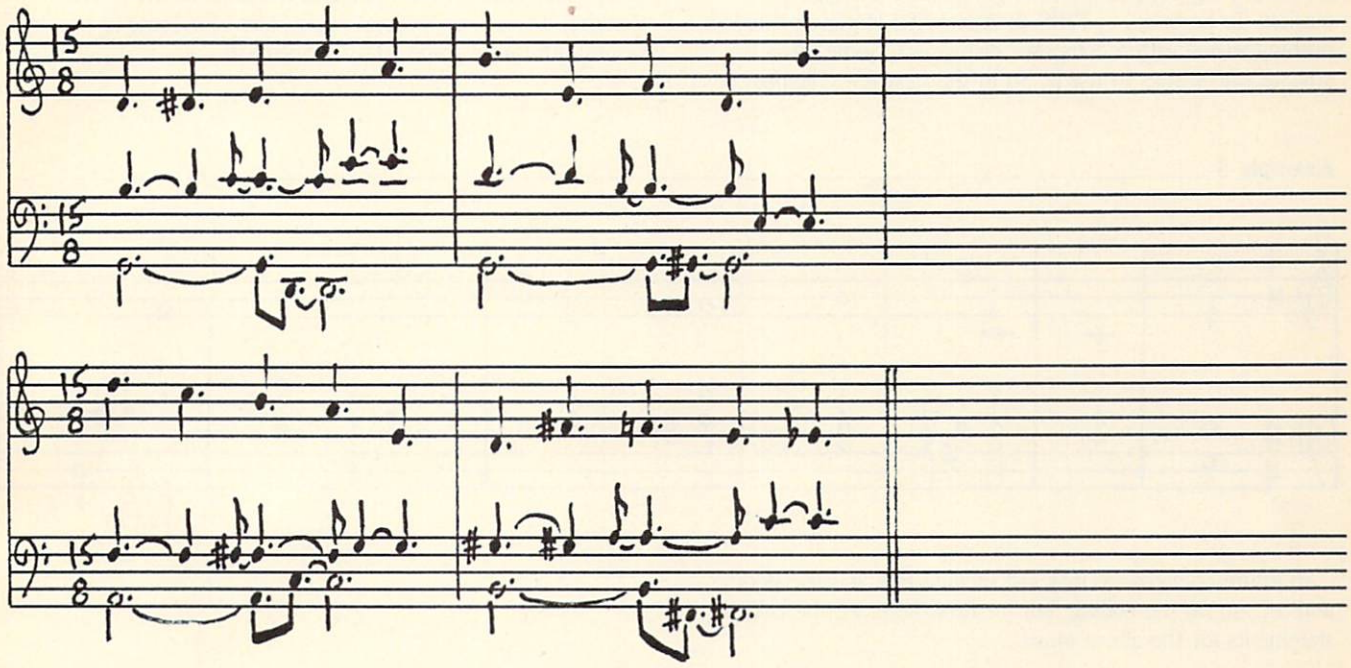
Listing 2

```
1000 DATA 704,699,698,697,699,704,697,699
1999 DATA 0
2000 DATA 560,562,564,565,566,567,565,564,563,562
2999 DATA 0
3999 DATA 0
```

Save the entire program before running it, then type RUN and listen to the performance. You can slow down the tempo by increasing the size of the loop in line 580. Also, the program automatically plays the uppermost voice one octave higher than written (see line 175). If you do not want this spread between voices then delete line 175.

Are you ready for a *triple* cross-rhythm? Let's hear how two notes divided evenly against three divided evenly against five sounds! The music is given below. Many musicians would have to practice this for quite a while at a very slow tempo to begin to play it. Our powerful little machine can perform it immediately and let the musician hear how it should sound at the proper tempo. (Say, do you suppose the 64 could be helpful teaching music students about how complex rhythms should sound *without mistakes?* . . ! . . ! ! !)

Example 4



If you have a piano handy and are inclined to a friendly challenge try playing the music to see what the coordinational stress is on your own central processing unit. Here are the changes we'll make to the original program. First, let's get the upper voice down where it belongs (line 175), then make some changes to the ADSR settings on lines 500-520. These will allow the notes to ring a bit longer. Finally, the DATA statements. SAVE, then RUN. (Be patient while the decoding is done . . . about ten seconds!)

Listing 3

Delete line 175
then

```
500 POKES+5,60:POKES+6,128
510 POKES+12,10:POKES+13,12
520 POKES+19,10:POKES+20,11
```

then

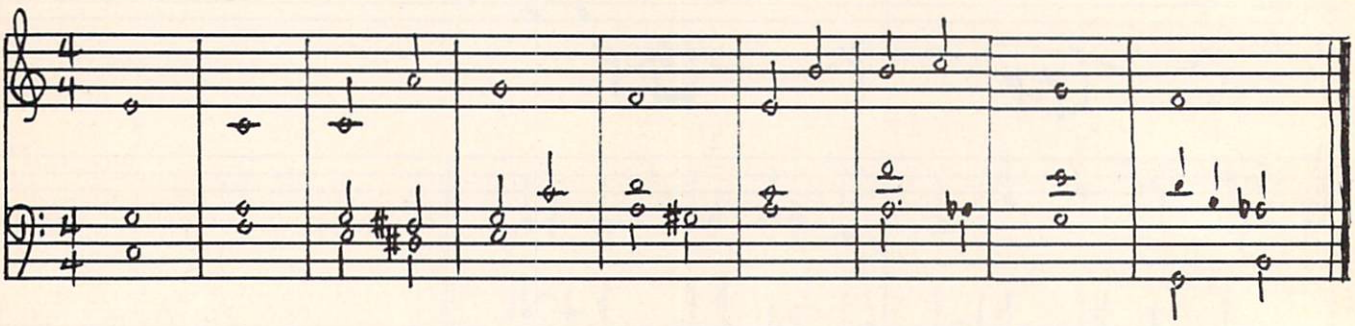
```
580 FOR T=1 TO 10 : NEXT
```

and finally

```
1000 DATA 834,835,836,848,841,843,836,837,834,843,850,848,843
1010 DATA 841,836,834,838,837,836,835
1999 DATA 0
2000 DATA 1339,1344,1348,1346,1339,1332
2010 DATA 1333,1334,1335,1336,1337,1344
2999 DATA 0
3000 DATA 1959,1954,1959,1960,1961,1968
3010 DATA 1963,1958
3999 DATA 0
```

One of the things some young programmers have asked me is about using popular music in programs they hope to market. To be brief . . . THIS IS A NO-NO! However, there is nothing wrong with a composer giving away some music to anyone who wishes to use it. So in the next few columns there will be some free "musical goodies" for your use—totally free forevermore! Here is a first offering. To my way of thinking it is a sort of "soap opera intro" for some as yet unwritten adventure game. Have fun with it!

Example 5



In future columns we may just include the "musical goody" and let you do the coding; but for now, here are the DATA statements for the above music.

Listing 4

Delete line 175.

then

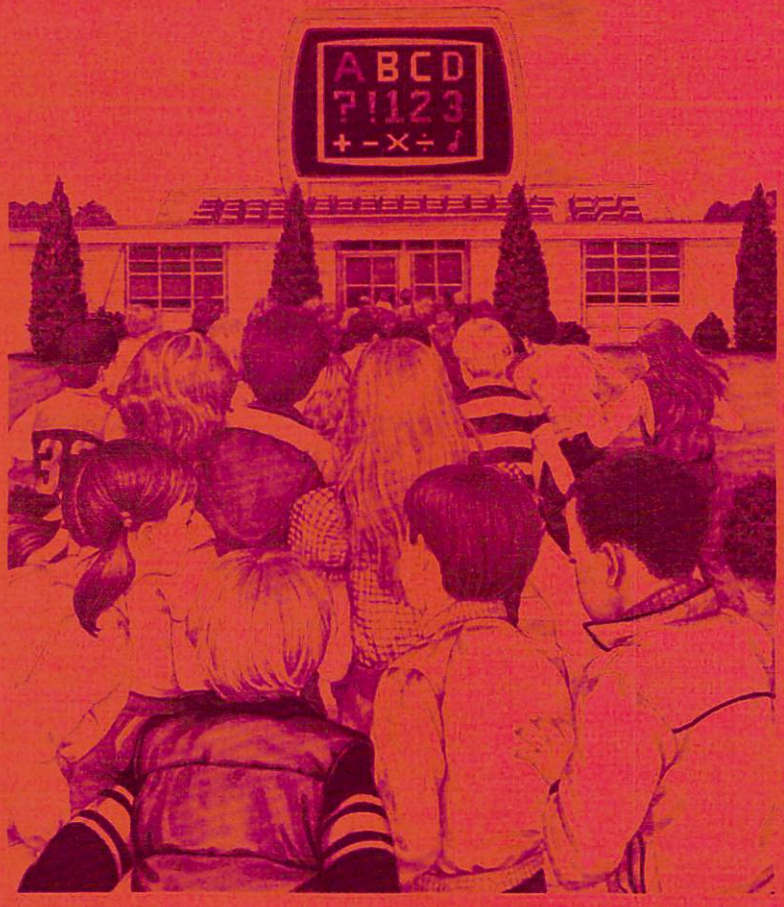
```
500 POKES+5,14:POKES+6,15
510 POKES+12,14:POKES+13,14
520 POKES+19,14:POKES+20,14
```

then

```
1000 DATA 2116,2112,1088,1097,2119,2117
1010 DATA 1092,1099,1099,1104,2119,2117
1999 DATA 0
2000 DATA 2103,2105,1079,1078,1079,1088,2114
2010 DATA 2112,2117,2116,578,571,1080
2999 DATA 0
3000 DATA 2096,2101,1076,1075,2100,1081,1080
3010 DATA 2105,1593,568,2103,1063,1067
3999 DATA 0
```

Next issue we will find out how to give our sound programs a "jazz feel" so they really SWING! **C**

commodore
the microcomputer magazine



Educators, parents, future parents and maybe even kids will find our special education section an invaluable resource in the year to come. Do you need to know where to find educational or administrative software for Commodore computers? Are you wondering which programming languages are most useful for students—and teachers? Find the answers to these questions and more in this super education section.

COMMODORE MEETS THE CHALLENGE OF A GROWING EDUCATION MARKET

BY DAVID ROSENWALD
Director of Education Marketing

I am sure it won't come as a surprise to any educators to learn that the "computer invasion" predicted by futurists has really begun—in force. In 1980 there were about 49,000 microcomputers installed in schools and colleges in the U.S. By the end of this year there will be about 175,000. The most astonishing thing about those figures is that they are only the beginning. By 1987 industry experts predict schools will have more than 2.5 million personal computers, or one for every 23 students. In terms of dollars, this means the market for computer hardware and software in education will grow from about $310 million this year to some $2 billion by the end of 1987.

This rapid growth in the number of computers in the schools places increased pressure on educators to prepare for—and cope with— a new approach to learning. It in turn places an attendant responsibility on computer manufacturers like Commodore to continually expand and improve our support for those educators.

As I pointed out in the last issue of *Commodore Magazine*, Commodore is more deeply committed than ever to increasing service and support to our educational users. As a result, we have established a comprehensive program that includes: 1) a regional network of education specialists who are familiar with both Commodore equipment AND the concerns of educators; 2) special discounts on equipment for qualifying schools; 3) education resource centers, through which educators help educators; 4) regional seminars for teachers; and 5) increased participation in state, regional and national educators' conferences.

Our regional education specialists, all of whom are former teachers and administrators, work with educators and Commodore dealers to make sure questions are answered and needs fulfilled. They also support teacher seminars in their respective regions. Please be sure to contact the specialist in your area:

Midwest: Mrs. Terry Anders
Commodore Business Machines
2246 N. Palmer Drive
Schaumburg, IL 60915
312-397-0075

South: Mrs. Pam Spillios
Commodore Business Machines
4350 Beltwood Parkway S.
Dallas, TX 75234
214-458-1000, ext. 220

East: Ms. Pat Walkington
Commodore Business Machines
761 Fifth Avenue
King of Prussia, PA 19406
215-666-6800

West: Mr. Jim Bussey
3824 Cougar Place
Modesto, CA 95356
209-526-0223

Working with me at headquarters are Dr. Dan Kunz, Program Manager for Educational Systems; Cindy Doms, communications and administrative coordinator; and Emma Jean Mungin, our secretary.

In addition to our education specialists, we are establishing many education resource centers to serve as a primary grassroots support system for educators. It is here that direct contacts, teacher-to-teacher, are made. The information exchanged through our resource centers is very useful for teachers working with computers, many for the first time, and many on limited budgets and tight time schedules.

To qualify as an education resource center you must be a non-profit school organization or agency (public or private) serving students in one or more of the following categories: preschool, K-12, post-secondary education, two- or four-year college or university or adult education. You must have at least five Commodore computers (this can, however, be modified if you meet the other requirements) and must be willing to provide, at reasonable intervals, support, in-service training and advice to other educators.

A school or intermediate unit may, with the approval of their school board, apply to Commodore to become an education resource center. If you would

like to find out more about this program, write to the Education Resource Center Coordinator, Commodore Business Machines, 1200 Wilson Drive, West Chester, PA 19380.

In addition to increasing and improving support programs for education, Commodore will also continue to increase the amount and quality of software available to educators. In fact, we hope to be able to provide, through large educational publishers, software and coordinated texts in many subject areas.

We see educational software as having three different uses: academic, business education and administration. In the academic area we are making available—in the public domain—656 programs designed to run on most Commodore computers. These programs are teacher-developed and address specific classroom needs. Each program is catalogued by topic, grade level and degree of student/teacher participation. Also included as part of that package are programs for administration—grade reporting, attendance, statistical analysis and similar functions. This is just the beginning of what will continue to be an unprecedented commitment to educational software. Recent announcements include the soon-to-be-available LOGO and PILOT software for the Commodore 64. In addition, a PET emulator has been developed that allows the 64 to use

Illustration—Robert Neumann

over 90% of the software existing for our original PET computers.

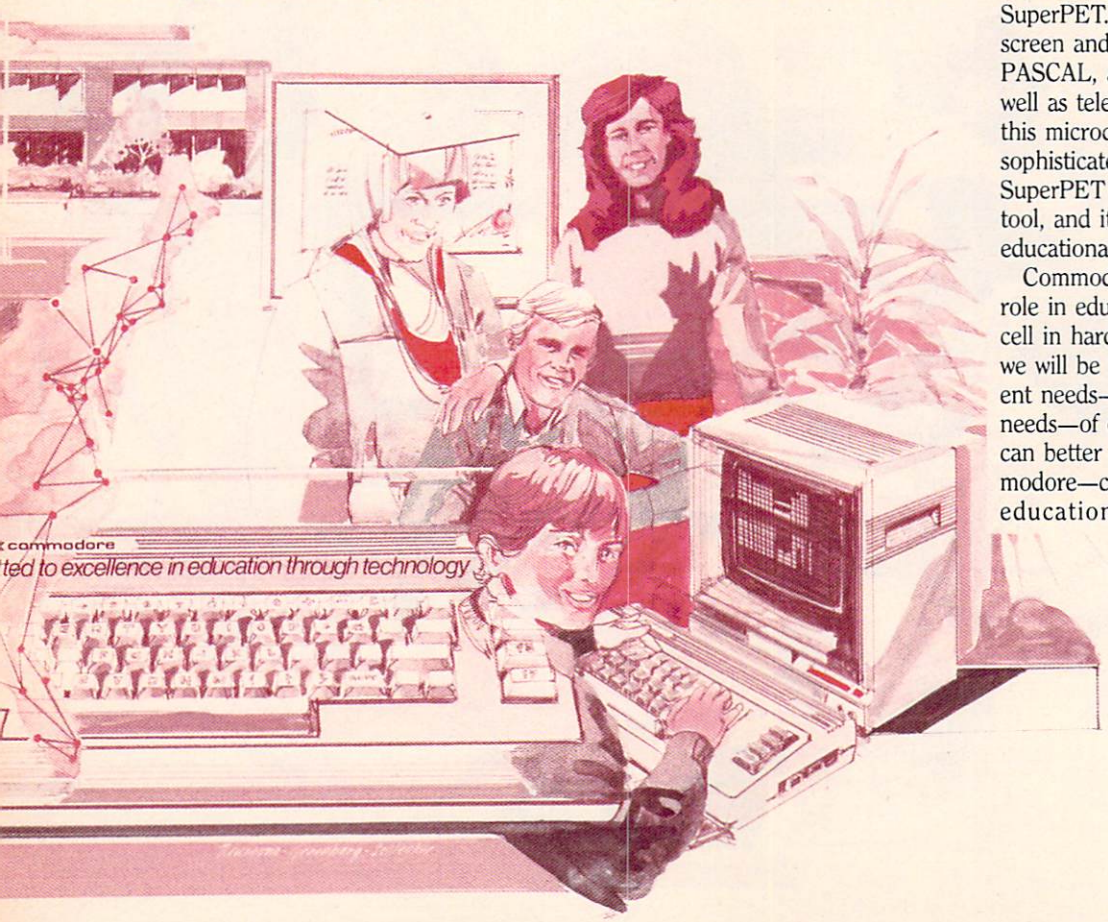
In the area of business education, we have found that the best way to learn a business application is to actually use the software used in industry. For instance there seems to be little advantage to only learning ABOUT word processing. To gain the greatest benefit students need to USE a word processor—or electronic spreadsheet. We at Commodore continually try to make such products available to our educational users.

Of course, Commodore is first and foremost a leader in the production of advanced computer hardware. Although we do many things well, this is the backbone of our reputation for excellence.

Our recent developments are no exceptions to that rule. The Commodore 64, for example, with its high resolution color graphics, music synthesizer and extraordinary memory capabilities—at an affordable price—will continue to be an outstanding tool for educators. The 64 is even more cost effective when it is networked in the classroom, or when it is connected to any of the major telecommunications data bases with our very inexpensive modem. The availability of sprite graphics further enhances the learning impact of the 64.

For more advanced educational applications—most specifically for teaching programming in the whole spectrum of computer languages—the SuperPET is the logical choice. In fact, there is no other microcomputer on the market with the built-in language capabilities of the SuperPET. With 96K, an 80-column screen and APL, FORTRAN, BASIC, PASCAL, and COBOL languages, as well as telecommunications capabilities, this microcomputer is ideal for sophisticated school applications. The SuperPET is an unexcelled development tool, and it serves business and other educational purposes, as well.

Commodore has always played a major role in education. As we continue to excell in hardware, software and support, we will be better able to meet the present needs—and anticipate the future needs—of our educational users, so we can better live up to our motto: "Commodore—committed to excellence in education through technology."**C**



HOW TO GET FEDERAL BLOCK GRANT FUNDING TO BUY COMPUTERS FOR YOUR SCHOOL

BY DAN KUNZ
Program Manager for Educational Systems

Federal Block Grants to schools are actually a consolidation of many programs that were previously individually funded. These grants may be used by a school district as it sees fit. Many districts are choosing to use the funds to buy microcomputers and related software and provide inservice training.

As an aid to educators, we are providing the following information directly from *Block Grants and Staff Development* by Margo Johnson, Theodore Andrews and Roy Edelfelt, published in 1982 by the National Council of States on Inservice Education. (Copies of this publication may be obtained from the National Dissemination Center, Syracuse University, 123 Huntington Hall, Syracuse, NY 13210.)

We have also included a list of state education agency block grant coordinators, taken from this same publication. These state coordinators should be able to answer any questions your district may have concerning current use of block grant monies. Before you contact your coordinator, however, please be sure to consolidate your district's efforts. Then have a district-level official make the contact. That way you'll avoid confusion and overlap.

The following is taken directly from *Block Grants and Staff Development:*

The federal block grant legislation mandates a system for giving federal funds to state and local education agencies for educational improvement. Most of the funds go to local education agencies. The law says that local education agencies have complete discretion to spend the funds, within the broad provisions of the law. The law does not provide that the state education agency should influence how the local education agency spends its funds, but neither does the law prohibit the state education agency from doing so. . . .

1. What is the block grant legislation?

The Education Consolidation and Improvement Act of 1981 has three chapters. Chapters 1 and 3 deal with Title I of the Elementary and Secondary Education Act (ESEA) and general provisions, respectively. Chapter 2 is what is referred to as the "block grant legislation" or the "consolidation legislation."

2. What programs did the block grant legislation consolidate?

The block grant legislation consolidated six legislative authorizations for 50 distinct programs (many of which were never funded) into a single legislative authorization for a block grant to each of the 50 states, the District of Columbia and Puerto Rico (the latter two are included hereafter in the word "states") and, through the states, to local education agencies (LEAs). The six antecedent authorizations were:

- Titles II, III, IV, V, VIII and IX (except Part C) of the Elementary and Secondary Education Act of 1965
- the Alcohol and Drug Abuse Education Act
- Part A and Section 532 of Title V of the Higher Education Act of 1965
- the Follow Through Act, on a phased basis
- Section 3-a-1 of the National Science Foundation Act of 1950 related to precollege science teacher training
- the Career Education Incentive Act.

Note that not all federal education programs were consolidated in the block grant legislation. Among the major programs *not* consolidated are Title I, education for the handicapped, bilingual education, Indian education, vocational and adult education and impact aid.

Block Grant State Directory

Interviews with block-grants administrators in the 50 states and the District of Columbia indicate that most governors have appointed the advisory committees required by law and that some states have already developed formulas for distributing the money. The chart on these pages shows who each state's advisory-committee chairman is (if already selected); who is in charge of block grants for the state education agency; who will have final authority over the distribution formula; and what features the state's formula includes or is expected to include.

State	Chairman, state advisory committee on block grants	State education agency's block-grants coordinator	Final authority on distribution formula	Characteristics of formula for distributing block-grant funds to local education agencies
Alabama	Not available	Kennith S. Blankenship	State education agency	Not available
Alaska	Margaret Lowe (907) 276-8029	Raymond Minge Ron Bedard	State department of education	Weighted for sparsity
Arizona	Jewel Lewis and Thomas Ault	Thomas R. Reno	State education agency	5% extra for Title I students, 5% for sparsely populated districts, 5% for highest and lowest achievers on standardized test, 10% extra for districts that have adopted school-improvement plans
Arkansas	N.A.	Roland A. Carpenter	N.A.	N.A.
California	N.A.	Charles Cooke	State board, state superintendent, legislature, and governor	N.A.
Colorado	Walter Jackson (303) 668-3011	Arvin Blome	State board of education	83% based on enrollment, 16% for disadvantaged students, 1% for sparsely populated districts
Connecticut	Stephen B. Heintz (203) 566-4298	R. Douglas Dopp	State board of education, subject to approval by legislature	Will be weighted for educationally and economically disadvantaged children
Delaware	Henry C. Harper, Superintendent, Appoquinimink school district, Odessa	Randall L. Broyles	State board of education	70% based on enrollment, 10% on disadvantaged children, 10% on handicapped, 10% on gifted and talented
District of Columbia	N.A.	William Millar	Board of education	Will take into account high-cost programs
Florida	Leonard Britton, Superintendent, Dade County Schools	Marshall Frinks	Commissioner of education	Weighted for handicapped pupils, cost of living in district, students eligible for compensatory education
Georgia	C.L. Cain (404) 742-8292	Paul S. Goethe	State board of education	N.A.
Hawaii	William Brown (808) 525-8765	Ichiro Fukumoto	State board of education	Weighted to account for high-cost children, improvement of basic skills, special projects
Idaho	Linda Dewey P.O. Box 2638 Boise 83701	George H. Hunt	State department of education	N.A.
Illinois	Don Muirhead, Decatur	Bud Grossner	State board of education, subject to reappropriation by legislature	N.A.
Indiana	Jim Neal, Noblesville *Daily Ledger*	William F. Miller	Governor	85% on population, 10% on high-cost children (as measured by eligibility for free lunch), 5% for sparsity
Iowa	Anne Jorgenson (319) 477-5131	James E. Mitchell	State board of public instruction	75% based on enrollment, 20% on children eligible for Title I; 5% for limited-English-proficient students
Kansas	H.B. Scott (316) 862-5304	Warren J. Bell	State board	88.1% based on enrollment, 2.1% on handicapped and gifted, 9.3% on low-income pupils, 0.4% on sparsity
Kentucky	Fannie Gue (502) 366-0373	Alice McDonald	State superintendent of public instruction	Weighted for sparsity, disadvantaged pupils, high-tax/low-yield districts
Louisiana	N.A.	Leon R. Tarver Jr.	State board of elementary and secondary education	Will consider school-age population, disadvantaged and urban students, number of private schools to be served
Maine	Joyce Roach (207) 757-8407	Vendean V. Vafiades	State department of education, subject to approval by legislature	60% based on enrollment, 40% for economically and educationally disadvantaged
Maryland	David Daneker (301) 396-6890	Richard Petre	State board of education	80% based on enrollment, 20% on Title I and handicapped students
Massachusetts	N.A.	James Case	State board of education, subject to approval of legislature	N.A.
Michigan	Joseph Pollack (Interim) (313) 762-1249	Robert McKerr	State of board education, subject to reappropriation by legislature	N.A.
Minnesota	N.A.	Gregory J. Waddick	State commissioner and state board	N.A.
Mississippi	N.A.	Bob C. McCord	N.A.	N.A.

State	Chairman	Coordinator	Final authority	Characteristics of formula
Missouri	N.A.	Otis G. Baker	Subject to reappropriation by legislature	Weightings expected for low-income and handicapped children
Montana	John Frankino (406) 442-5761	Patrick Feeley	State superintendent	Based on attendance
Nebraska	N.A.	Larry Vontz	State board of education	N.A.
Nevada	N.A.	James P. Costa	State department of education	Will take into consideration low-income and economically depressed areas and rural isolation
New Hampshire	N.A.	Robert L. Brunelle	State board of education	N.A.
New Jersey	N.A.	Marvin S. Habas	State board of education	N.A.
New Mexico	N.A.	Susan Brown	State board of education	70% based on enrollment, 30% for sparsity and rural isolation
New York	Andrew Virgilio State University of New York at Brockport	Robert J. Maurer	Board of regents, subject to reappropriation by legislature	Weightings for districts' fiscal capacity (property value and income per capita), sparsity, special-education needs, limited-English-proficient students. Next year, each district is guaranteed at least 20% of the federal aid it received in 1980-81.
North Carolina	Ed Holmes (919) 542-3232	Weaver B. Rogers Jr.	State board of education	N.A.
North Dakota	Jerry Tjaden (701) 347-4432	Joe Crawford	Department of public instruction	N.A.
Ohio	N.A.	Irene G. Bandy	State education agency	Will include weightings for disadvantaged children, sparsity, high-tax/low-yield districts, handicapped children
Oklahoma	Leonard Campbell (405) 722-5201	Jack Strahorn	State board of education	80% based on average daily membership; weightings expected for low-income and handicapped students and for small districts
Oregon	William K. Worrell (503) 963-4171	Marshall D. Herron	State board of education	70% based on average daily attendance; remainder to be weighted for low-income, handicapped, rural, gifted, and limited-English-proficient students, and for programs for desegregation or cultural enrichment
Pennsylvania	John Hersey (717) 533-5514	Jo Ann Weinberger	State education agency, subject to reappropriation by legislature	60% based on enrollment, 35% based on poverty, 5% on sparsity
Rhode Island	James DiPrete (401) 828-7800	Chris O'Neil	Board of regents for elementary and secondary education	N.A.
South Carolina	Warren Mersereau (803) 242-6300	Jack Seurynck	State department of education	N.A.
South Dakota	N.A.	James O. Hansen	State board of education	Will take into account small schools and sparsely populated districts
Tennessee	N.A.	Carol Furtwengler	State commissioner of education	N.A.
Texas	Willis Tate, Southern Methodist University, Dallas	William N. Kirby	State board of education	Double weight given for Title I-eligible, handicapped, and limited-English-proficient students
Utah	Eldon Tolman (801) 266-4411	Bernarr S. Furse	State board of education	Same as for state aid to local school districts
Vermont	N.A.	Gerald Asselin	State board of education	55% based on enrollment, 45% based on high-cost factors, including high local tax effort with low yield, number of low-income students
Virginia	N.A.	William C. Cochran	State board of education	N.A.
Washington	N.A.	Mona Bailey	N.A.	Weightings expected for low-income students, lowest and highest achievers
West Virginia	N.A.	Carolyn Skidmore	N.A.	N.A.
Wisconsin	The Rev. John Hanley (414) 476-2101	Brian Roherty	State superintendent of public instruction	N.A.
Wyoming	John Rankine (307) 347-6355	Tom Morris	State board of education	Weighted for rural isolation, gifted and talented students, local tax burden

Source: *Education Week*, March 24, 1982. Reprinted by permission.

Jake Smith

GLOSSARY

Turning gobbledygook into language you can use

This glossary is provided to acquaint the novice with working definitions of words related to computers. It is not designed to be exhaustive nor definitive. Definitions have been taken, and in some cases reworded, from the following primary sources:

Educational Testing Service—Glossary of Computer Terms—Marty Schneideman, Princeton, N.J. 1981.

Data Phase Systems—Glossary of Computer Terms—Kansas City, MO, 1981.

Jef Raskins Brief Dictionary of Computerese/Glossary—Jef Raskin—1978.

At least four other dictionary/glossary type documents, the sources of which are currently unknown.

Reprinted from the December/January 1982 issue of *Interact*, the publication of the New Jersey State Department of Education. Revisions and additions by Mike Grabel and Dan Kunz at Commodore.

A

Address: The particular number associated with each memory location.
Algorithm: A step-by-step procedure by which instructions are given to a computer.
Alphameric: Letter characters only.
Alphanumeric: A character; alphabetic, numeric or symbolic.
Analog: A method of measurement that uses a continuously variable physical quantity, such as length, width or resistance, to represent values.
APL/A Programming Language: A computer language used primarily for mathematical applications. Uses special symbols to reduce the size of programs.
Array: A list of items any of which may be found by their position in the list.
ASCII: Often called U.S. ASCII. The standard code value from 0 to 128 that is assigned to numbers, letters and control characters for the purpose of communication between two computers.
Assembler: A program that converts assembly language to machine language.
Assembly Language: A language similar to machine language, but easier to write and understand.
Attack: The rate at which musical notes reach peak volume.

B

Back-Up Copy: A duplicate.
BASIC—Beginners All-Purpose Symbolic Instruction Code: A conversational computer language, using words, not symbols; frequently used in microcomputers.
Batch: Holding of similar jobs for processing at a later time. Opposite of interactive.
Baud Rate: Speed of data transfer from one computer to another in approximate bits per second.
Binary: 1's and 0's represent data. Similar to an on/off switch. The way a computer stores information.
Bit: A single binary digit. The smallest unit of information understood by a computer. Do not confuse this with a single character or letter. See BYTE.
Bug: An error. May exist in either hardware or software. Also called a glitch.
Buffer: An area in the computer used to temporarily hold information.
Byte: A group of bits. Represents one character such as a letter and is normally eight bits in length. Measured in units of approximately 1000 called K.

C

CAL—Computer Assisted Learning: A generic term used to refer to different ways of using the computer to help humans learn or to keep track of a human's learning.

- **CAI**—Computer Assisted Instruction.
- **CBI**—Computer Based Instruction.
- **CMA**—Computer Managed Instruction.

Cassette Interface: A slow process of using a regular cassette tape recorder and tape to transfer stored data to and from a computer.
Character: A single letter, number, symbol, or space. Requires one byte of storage.
Character Printer: A printer that prints one character at a time. Usually quite slow with variable quality.
Chip: A tiny piece of silicon embedded with hundreds or thousands of electronic circuits. See MICROPROCESSOR.
COBOL—Common Business Oriented Language: A computer language primarily used for business applications.
Cold Start: To begin to operate a computer that has just been turned on.
Column: A vertical sequence of characters, from top to bottom.
Command: A request or instruction to a computer.
Compatability: Two types: hardware to hardware and software to hardware. The ability to correctly match or connect two or more items.
Compiler: A translation device. Translates a computer language written by a programmer into a language that the machine understands.
Computer: An electronic device that can receive and then follow instructions to manipulate information. Changeable instructions, data storage and the ability to handle numbers, characters, and letters are characteristics. NOTE: Definition of the three basic levels of size in the computer family change rapidly as new hardware is developed.

- **Computer System:** All the hardware and software used in a complete computer operation or task.
- **Microcomputer:** A very small computer usually based on a microprocessor.
- **Minicomputer:** Larger and more powerful than a microcomputer.
- **Mainframe Computer:** The largest of the computer family.

Computer Education: Teaching about computers.

- **Computer Awareness:** A human's understanding of how computers are used in the everyday world.
- **Computer Literacy:** A human's ability to use computers in the everyday world.

Control Character: A character that has no graphic meaning but tells the computer to do something, e.g., cursor controls.
CP/M*: A common operating system on many computers that uses a Z80 microprocessor chip. (*CP/M is a registered trademark of Digital Research, Inc.)
CPU—Central Processing Unit: The brain. Serves as a switching network and performs all calculations. The internal operational control system.
CRT—Cathode Ray Tube: The display portion of a television, monitor, or video terminal.
Cursor: A movable spot on the CRT screen. Shows the programmer or operator where the pointer is.

D

Data: Information of any type.
Data base: Data, organized, stored, and cross referenced by a computer. Think of an organized file cabinet with a common concept that links the drawers together.
Debug: Locating and eliminating errors, bugs, or glitches.
Decay: The rate at which a musical note falls from peak volume.
Diagnostic Routine: A computer's self-testing program which can identify hardware and some software bugs.
Digital: A method of measurement that uses precise quantities to represent values, such as the presence or absence of voltage.
Disk (Disc): A device that looks like a phonograph record on which large amounts of data may be stored. Allows very fast random access of that data. May be hard or floppy, recordable on one or both sides, and single or dual density.

- **Disk drive**—A peripheral that reads and writes information on disks. Think of a record player that acts like a tape recorder.
- **Disk Operating System (DOS)**—See Operating System.
- **Diskette**—A floppy disk.
- **Floppy disk**—A flexible disk usually 5¼ or 8 inches in diameter.
- **Hard disk**—Large size disks that are not flexible.
- **Mini-floppy disk**—3-inch disks.

Display: A video screen. See CRT.
Documentation: A written description of a computer system. Allows one to understand the purpose and logic of someone else's program.
Dot-Matrix: Using printed dots in an array to represent letters.

E

Electronic Spreadsheet: Software that manipulates very time consuming interrelated arithmetic operations in a fast and simple way.
Execute: To run a computer program.
External Storage: Storing data via a peripheral device such as tape or disk.

F

File: A data set with some common delimiter. For example, think of a file drawer devoted to budget information.
Flow Charting: Using various shaped blocks connected by lines to show the sequence of operations in a computer program.
Format: To prepare a diskette for use.
FORTRAN - Formula Translator: A computer programming language primarily used for scientific or mathematical applications.
Friction Feed: Typewriter-like paper feed.

G

GIGO - Garbage In, Garbage Out: Errors that are put in will result in errors that come out.
Graphics: All non-alphanumeric displays generated by a computer.
Graphics Tablet: A peripheral that translates lines drawn on a board to some output device such as a CRT or printer/plotter.

H

Hard Copy: Output printed on paper.
High Resolution: Extremely clear video, much more so than normal television. Desirable for graphics.
Hardware: The physical parts of the computer system. Equipment.

I

IEEE-488: A standard interface used to connect peripherals to CBM and PET computers. See interface.
Input: The information or data that goes into a computer or its peripherals.
Interactive: Allows the user to communicate with the machine during operation. Opposite of batch.
Interface: A device that allows two other devices to communicate with each other, particularly a peripheral to a computer. Also used between persons and with persons and machines.
Interpreter: A program that translates a programmer's instructional language into the language the machine understands.
I/O - Input/Output: Devices, as well as the process for putting information into and taking information out of a computer. Printer, Keyboard, CRT, etc.

J

Joy sticks: Small peripheral devices that are used primarily for graphics and computer games.

Keyboard: Typewriter in the form of a terminal used to enter information into the computer.
K: Unit of measurement for memory size in computers. One K is approximately 1000 Bytes.

Language: A set of conventions that specify how to tell a computer what to do. High-level language is one that is considered conversational and is reasonably easy for a programmer to learn.
Letter Quality Printer: Typewriter-quality printing.
Line: A horizontal sequence of characters, from left to right.
Load: Inserting information into a computer.
LOGO: A new language using English. Designed primarily for children's use.

M

Machine Language: Instructions to the computer in binary code.
Magnetic Storage: Usually tape or disk type storage. Retained when the power is turned off. Can be destroyed by magnetic or electrical interference.
Megabyte: One million bytes
Memory: The portion of a computer that stores information, measured in bytes.

- **Main Memory** - RAM (Random Access Memory): Internal memory which is erased when power is turned off. Also ROM (Read Only Memory) which is permanent and internal to the machine.
- **RAM - Random Access Memory:** Internal to the computer and lost when the power is turned off (volatile memory). The computer's main internal memory and program storage system. The user can alter or manipulate this memory.
- **ROM - Read Only Memory:** Information stored in a memory chip that generally cannot be altered and is not affected by power loss. The user cannot alter or manipulate this memory.
- **Buffer Memory:** Temporary memory in the workspace.
- **Memory Location:** The smallest division of the memory map.
- **Memory Map:** A diagram to all memory locations that can be addressed by the computer.

Menu: A list of program options from which to choose; catalog.
Micro: Quite small.
Microprocessor: The computer's brain on a quarter-inch square silicone chip.
Modem: Modulator - Demodulator: An electronic device, or interface, which allows computers to communicate over telephone lines.
Monitor: A high resolution cathode ray tube. Think of a special television with a very sharp picture.

N

Network: To connect a number of computers to a main computer. In the classroom the main computer usually runs a printer and a disk drive.

O

Operating System: The internal programs that allow the computer to perform its functions.

- **Disk Operating System-DOS-** The internal programs that allow the computer to use a disk system.

Output: Information coming out of the computer or the end product of a program. Usually on paper or CRT.

P

Parallel: Transmission of data so all parts of a word are handled simultaneously, eight bits at a time.
Pascal: A new computer language similar to but more powerful than BASIC.
PEEK: A BASIC command that displays the value (a number from 0 to 255) of a specific memory location.
Peripheral: Any hardware that connects to computers.
PILOT: A programming language that allows educators to easily design lessons and tests for classroom use.
Pixels: Divisions on a display screen. The more pixels, the higher the screen's resolution.
Plotter: A peripheral with a microprocessor that does graphics.
POKE: A BASIC command used to put a one-byte value into a given memory location.
Printer: A peripheral that makes hard copy (paper) of computer output.
Program: Instructions given the computer in a "programming language" to perform a specific task.
Programmer: A person who writes and documents programs.
Programming Language: Languages that a computer understands such as BASIC, PASCAL, FORTRAN, COBOL, APL, PL/I, etc.

Q

Queue: A waiting line. The sequence of programs to be run by the computer.

R

RAM: See memory.
Random Access: When any portion of the stored information can be found and retrieved in a nonsequential search in approximately the same amount of time.
Record: A set of information that contains all data about one item. Files are made of records. Think of a file folder on one topic.
Resolution: The clarity of a video display.
Response time: The amount of time between an instruction or question to a computer and its response.
ROM: See memory.
RS-232: An industry-standard interface that can be used to connect peripherals to Commodore computers.

S

Scrolling: A data display technique for a CRT. After the screen is full the next line comes up from the bottom and the top line disappears from view.
Searching: The act of looking for a specific piece of data.
Security: Protection of both hardware and software.

- **Data Security** - Protection of data from unauthorized access, modification, or destruction.

Serial: Transmission of data one bit at a time.
Software: The programs that instruct the computer how to operate.

- **Courseware** - Software in a configuration similar to a course structure for instructional purposes.
- **Firmware** - Software permanently stored in a computer's memory chip. See ROM.

Speech Synthesis: The software and hardware that converts written information into audio signals that sound like human speech. A similar system exists for music.
Sprite: A graphic character that can be designed by the user and programmed on the Commodore 64.
Stack: An area in memory reserved to store information temporarily.
Storage: A place where data is located in the computer or its peripherals.

- **Mass Storage** - Data stored on tape or disk.
- **Storage Capacity** - The amount of bytes a storage device can hold measured in K.
- **Storage Device** - A peripheral that holds information. Usually a tape or disk.

Subroutine: A portion of a program that performs a specific subtask.
Syntax: The rules that specify exactly how an instruction can be written.
System: A mechanical and electronic operation that allows input, processing, and output of information.

T

Tape: Usually a magnetic cassette for data storage on a microcomputer.
Terminal: A peripheral consisting of a keyboard, videoscreen or paper printer, and a means of connecting them to the computer.

- **Dumb Terminal** - A terminal that has no self contained computational ability when disconnected from a computer.
- **Smart Terminal** - A terminal that is usually a stand alone computer when not connected to another computer.

Thermal Printer: A paper printer that uses heat sensitive paper.
Time Sharing: A method that allows two or more users to use the same computer simultaneously.
Tractor Feed: A printer that uses paper with holes in the side.

W

Warm Start: To restore the operation of a computer without turning it off.
Winchester: A disk technology that allows both compacting of information and extremely fast access to that information.
Word Processing: The use of a computer to produce documents. Think of a "smart typewriter."
Work Space: The information a programmer or operator sees on the CRT when working on a program before it is stored.

Z

Z: 80, 8080, 8085, 8088, 6502, 6800, 9900, F8000, 6809, etc.-Names of the most commonly used microprocessors or chips. **C**

Considering

Pre-Purchase Questions and Sources of Answers

This article originally appeared in *Interact* (formerly the New Jersey State Department of Education newspaper), December, 1982. It is reprinted with some modification by permission of the New Jersey State Department of Education.

You're an educator and you don't want to fall behind the times. You see the growth of technology in America and the boom in computer equipment and services. You realize it is important for your students to become computer literate and you also realize a computer can help you with your management tasks. But where do you start?

Comprehensive planning always pays off. In the case of purchasing computers, such planning begins with a set of questions to guide your thinking. Your answers to the questions listed here will help you clarify your needs, and thus help you get started toward making a decision about which system or systems would be best for you.

Why do you need a computer?

- Do you need a computer or want a computer?
- What are you planning to use it for?
- How much time will it be in use?
- Does its use fit into existing curriculum and management planning or does it stimulate such planning?
- How much time will computer use save you? How will this extra free time be used?
- How does time/cost savings compare with initial expense and ongoing support expenses?
- What benefits can be gained that may not be possible in another way?

Do you have any computer expertise?

- Are you conversant in the terminology?
- Are you able to make logical comparisons between systems?
- Are you willing to learn about this new field?
- Who will be responsible for the system and its operation?
- Does someone have the time to devote to making the system work correctly?
- Will necessary in-service be supported?

Sources

Now that you've clarified your thinking, here are some sources of information to help you narrow things down even more.

"1979-80 Microcomputer Report": Recommendations for schools as they begin the process of selecting microcomputers. The 90-page book contains checklists and suggestions for evaluation and purchase of equipment. It may be ordered from the Minnesota Educational Computing Consortium, 2520 Broadway, St. Paul, MN, 55113.

"Topics—Computer Education for Elementary and Secondary Schools": The final report of the American Computing Machinery, Elementary and Secondary School Subcommittee (ES3) report. Order ACM publication No. 812810, from the ACM Order Department, PO Box 64145, Baltimore, MD, 21264.

"Microcomputers: Their Selection and Application in Education": This is a special 134-page journal containing 11 articles by leaders in the field. Order from Association for Educational Data Systems (AEDS), 1201 16th St., NW, Washington, DC, 20036.

"Application of Microprocessors in the Schools": a study that surveys the use of microcomputers in the six New England states and New York. The results are available from the Technical Education Research Center, 8 Eliot St., Cambridge, MA, 02138.

"Oregon Council for Computer Education": Film Bibliography listing 425 films, available from the Association for Computing Machinery, 1133 Avenue of the Americas, New York, NY, 10036.

"Issues Related to the Implementation of Computer Technology in Schools—a Cross-Sectional Study": A preliminary report presented to the National Institute of Education Conference on Issues Related to the Implementation of Computer Technology in Schools. Available from: Bank Street College of Education, Children's Electronic Laboratory, 610 W. 112th St., New York, NY, 10025.

Computers?

What approach should you take?

- What system configuration will best meet your needs and provide for future expansion: e.g. stand-alone systems, distributed processing, languages.
- Have you discussed available systems and can make reasonable comparisons between them?
- How fast is what you are considering becoming obsolete?
- Can the system be expanded to fit your needs in the future without requiring major changes or lost time?

What equipment do you need?

- How many tasks can it perform?
- Is the equipment alterable to meet future needs?
- Is the speed of the machine fast enough for your needs?
- Are the machines directly accessible at all times or only accessible by phone or only part time?
- What is the reputation of the company?
- How fast is servicing available?
- Are the machines compatible with other brands that other schools or the state may have bought or are using?
- Are the machines easily compatible with the type of add-ons (peripherals) that you may wish in the future?
- What peripherals do you need now? How much is too much? Can you distinguish between want and need? Can you add these items later?
- Is the language that the machine uses easily learned by persons without a computer background?
- Is the system expandable to include more complex languages?

Is it properly budgeted for?

- Is in-service money available to train staff how to program the machine, or how to individualize instruction using the machine as a base? Also, is money available to train staff to evaluate the curriculum or software materials for bias?
- Are maintenance costs over a five-year period assured of passage in the district budget?
- If only a few microcomputers are purchased, is there a method of budgeting time for their use between interested parties?
- How reputable is the servicing agency?
- Are the types of machines to be purchased the best possible at the best cost for the type of application you need?

. . . and more sources

The following groups are sources for information on computer literacy.

Association for Computing Machinery, 1133 Avenue of the Americas, New York, NY, 10036.

Association for the Development of Computer-Based Instructional Systems (ADCIS), Computer Center, Western Washington State College, Dept. IM, Bellingham, WA, 98225.

Computers Using Educators, Don McKell, Independence HS, 1776 Education Park Drive, San Jose, CA, 95133.

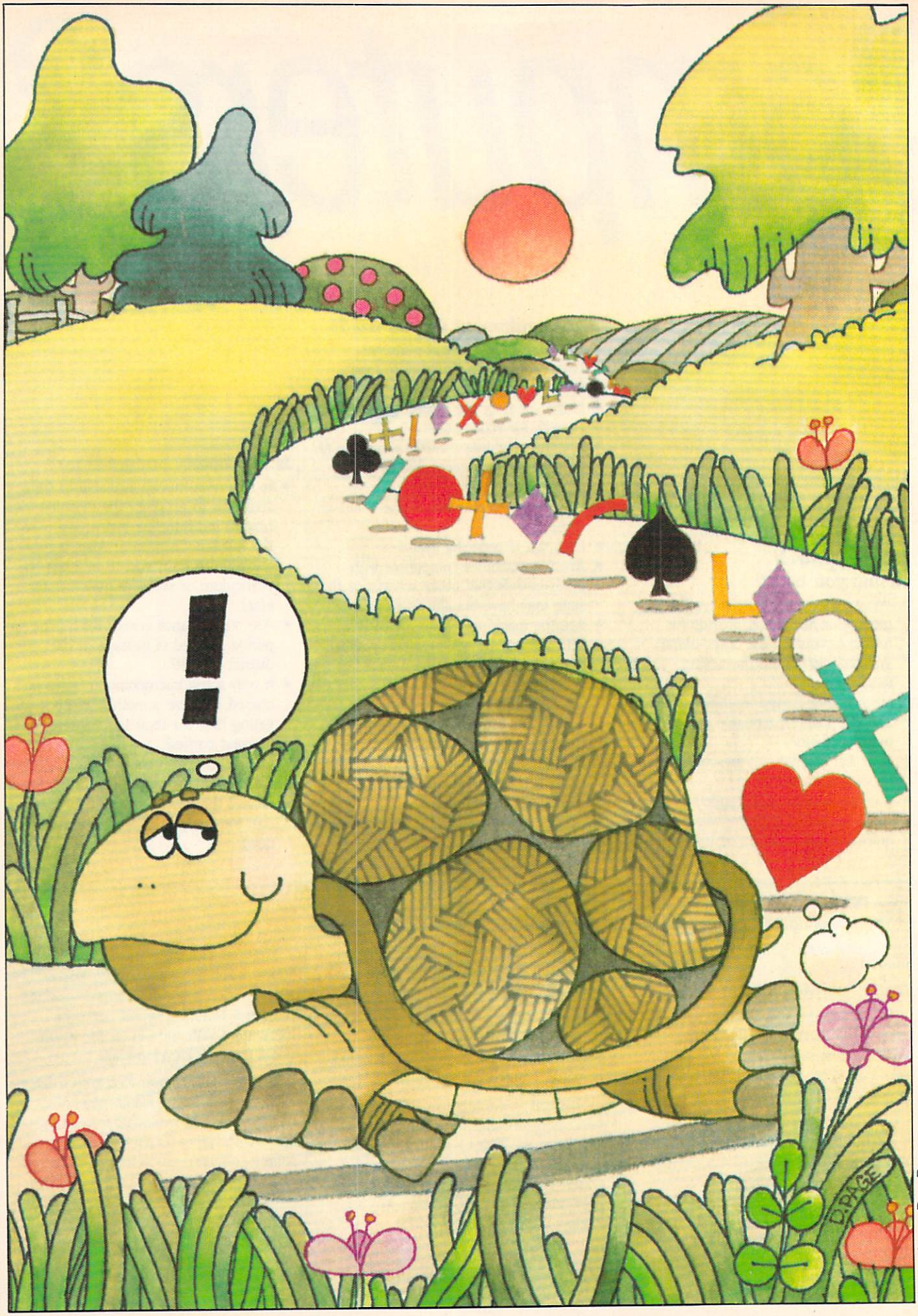
EPIE Institute, P.O. Box 839, Water Mill, NY 11976. Provide hardware and software reviews.

Human Resources Organization, 300 N. Washington St., Alexandria, VA 22314. Publishes Academic Computing Directory, by Robert J. Seidel.

Microsift Clearinghouse (Microcomputer Software and Information for Teachers), evaluates and disseminates software reviews of educational programs. Northwest Regional Educational Laboratory, 710 S.W. Second Ave., Portland, OR, 97204.

National Videodisc/Microcomputer Institute, Utah State University, Logan, UT 84322. Sponsors annual workshops and publishes technical report series in association with the Center for Institutional Product Development.

Technical Education Research Center (TERC), 8 Eliot St., Cambridge, MA, 02138. Operates a computer resource center, providing teacher training, consulting and information to educators. **C**



Illustration—Don Page

LOGO

A language for everyone

BY JIM BUSSEY

LOGO will soon be available for the Commodore 64 and, in addition to turtle graphics, will include word-list manipulation capabilities. It will be distributed through Commodore dealers.

In his book, *Mindstorms: Children, Computers and Powerful Ideas* (Basic Books, 1980), Seymour Papert, the developer of LOGO, explains the two fundamental ideas behind his work with children and computers:

". . . The first is that it is possible to design computers so that learning to communicate with them can be a natural process, more like learning French by living in France than like trying to learn it through the unnatural process of American foreign-language instruction in classrooms. Second, learning to communicate with a computer may change the way other learning takes place."

LOGO is a computer language developed over a 12-year period by Dr. Papert, his colleagues at M.I.T. and other researchers around the world. It was designed to combine discoveries in artificial intelligence with the learning theories of the educational psychologist, Jean Piaget, and is often used as a "first computer language" for children. The term LOGO comes from the Greek word for "thought", a fitting term for a language dedicated to teaching thinking through problem solving.

Although LOGO is often used as a programming language for young children, it is certainly not limited to that use. In fact, it can have applications for graduate students in computer science, as well. This is because, although it is very simple to learn, LOGO is limitless in its ability to explore complex mathematical relationships and ideas. As a result, it can be used to teach computer programming, computer literacy and for specific applications in many subjects, including mathematics, music, language arts and the sciences. However, its most noble purpose (pardon me if this sound too lofty) remains to teach people how to solve problems—even complex abstract problems—by drawing on their concrete experience of the world.

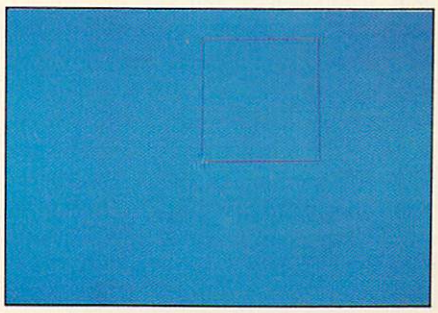
Piaget's research indicates that even very young children have theories about how the world works. Using this information, Papert sought to create an environment in which children could explore their theories of the world and use them as a basis for learning. He developed LOGO so children could use a computer to solve problems by drawing on their own personal experience—namely their understanding of their own physical movements.

LOGO does this through the use of a "turtle", a computer-controlled graphic character that has two characteristics—position and heading—each of which may be controlled independently. Children program the turtle to move on the screen by figuring out how they themselves move. In fact, by pretending to "play turtle" children can walk through a program and construct geometric figures that they can incorporate into more complex programs. Once they make the connection between their own movements and the movements of the turtle, the world of turtle graphics opens up an enormous playground for exploring ideas.

The beauty of LOGO is its open-ended nature. It does not lead to "right" or "wrong" answers. It simply provides a method for pursuing interesting problems—problems that can become increasingly difficult as the level of competence increases. The LOGO environment is ideal because it allows children to set their own goals and measure their own progress as they create programs. It also helps develop a high degree of critical thinking, especially when the children must debug and refine these programs.

Using the LOGO turtle, children build increasingly sophisticated programs by defining shapes, then using these shapes to define more complex shapes. For example, after a short trial-and-error period a student learns to draw a square:

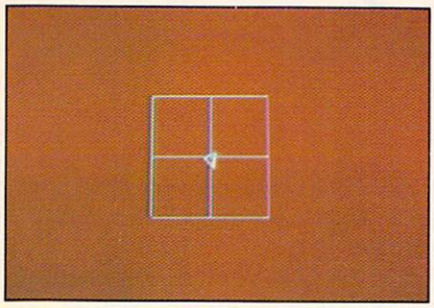
```
To Square
Forward 50
Right 90
Forward 50
Right 90
Forward 50
Right 90
Forward 50
```



The term "To" tells LOGO to enter this procedure called Square (it could have been called anything) into the language's list of terms that can be used in other programs. In other words the programmer is creating his or her own unique version of LOGO.

The procedure called Square can now be used to create a new procedure. For instance, combining four squares together creates a new shape, which you might label Window. The Window procedure would be defined like this:

```
To Window
Square
Square
Square
Square
```

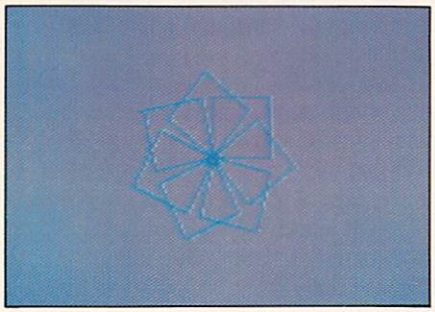


A simpler way of writing this program would be:

```
To Window
Repeat 4 {Square}
```

Then, using the procedure Window, you could:

```
To Flower
Repeat 20
{Window right 20}
```



Writing programs in LOGO becomes fun as students begin to see the potential for creating different shapes. Since there are no right or wrong shapes students can test their theories, receive immediate feedback and acquire a sense of their own accomplishments as they program and debug combinations of increasingly complex procedures.

This is necessarily a very brief synopsis of what LOGO is and what it can do. If you would like to gain a more thorough understanding of how LOGO is being used, please consult any or all of the following organizations, books and articles.

Organizations

Boston Computer Society,
LOGO Users Group
One Center Plaza
Boston, MA 02108

Center for Children and Technology
Bank Street College
610 West 112th St.,
New York, NY 10025

FOLLK (Friends of LISP, LOGO and Kids)
436 Araballo Drive
San Francisco, CA 94060

Friends of the Turtle
P.O.B. 5537
Eugene, OR 97405

Logo Times
99'er Magazine
Eugene, OR 97405

YPLA (Young Peoples' Logo Association)
1208 Hillsdale Drive
Richardson, TX 75081

Books

Abelson, Harold. *Logo.* Byte Publications, 1982

Abelson, Harold and diSessa, Andrea. *Turtle Geometry.* MIT Press, 1980

Boden, Margaret. *Artificial Intelligence and Natural Man.* Basic Books, 1977

Goldenberg, E. Paul. *Logo For Handicapped Students.* University Park Press, 1979

Muller, James H. *The Turtle's Sourcebook.* Young Peoples' Logo Association, 1983

Papert, Seymour. *Mindstorms: Children, Computers and Powerful Ideas.* Basic Books, 1980

Articles

Two magazines recently devoted whole issues to LOGO:

BYTE, July, 1982

The Computing Teacher, November, 1982

Author Unknown. "Are you a Turtle?" *Computing Teacher,* March 1982, p. 9. Describes Young People's LOGO Association, their parents and teachers in the enjoyment of learning about computers.

Blank, George. "Tourist's guide to the Cybernetic Tower of Babel (languages)." *Creative Computing,* Nov. 1981, pp. 94-103. Strengths and weaknesses of ten computer languages and lists the same short program in all ten languages.

Brown, Chris. "Mindstorms: children, computers and powerful ideas." *80 Microcomputing,* Feb. 1981, p. 14. A review for a book by Seymour Papert about what can and might be done with computers in the field of education.

Carter, Richard. "Logo and the great debate." *Kilobaud Microcomputing,* Sept. 1981, pp. 48-51. Using the computer to teach traditional curricula and teaching the students how to program computers.

Eyster, Richard. "Seymour Papert and the Logo universe." *Geographic Location: United States,* Dec. 1981, pp. 70-74. Describes the Logo child-oriented language and the content of Seymour Papert's lectures at a two-week workshop.

Heller, Dorothy. "User-friendly languages of the future." *Interface Age,* Dec. 1981, pp. 78-82. Discusses the problem of creating a computer language that is easily understood by people.

Heller, Dorothy. "Young People's Logo Association promotes learning." *InfoWorld,* Dec. 1981, p. 16. Reports on the Association's activities to provide

young people, their parents and teachers with an opportunity to learn about computers and computer languages.

Hunter, Beverly. "CZ03 and the Alien Key (programming problem)." *Electronic Learning*, Mar./Apr. 1982, pp. 52-53. A program in a language that resembles Logo and asks students to debug the program and find errors.

Kelman, Peter. "Journey through mathland: an interview with Seymour Papert." *Classroom Computer News*, Mar./Apr. 1981, pp. 10-11+. Presents an interview with the creator of LOGO, Seymour Papert.

Koetke, Walter. "Educator's recess: something for everyone in Minnesota." *Kilobaud Microcomputing*, Aug. 1981, p. 18-19. Reports on the annual meeting of the American Educational Data Systems (AEDS) held in Minnesota.

Lawler, Robert. "Logo Ideas." *Creative Computing*, April 1982, pp. 138-139. Gives short problem-solving exercises using Logo, this one for making a circle.

MacDonald, Doug. "Recursion: solving age-old mysteries." *Kilobaud Microcomputing*, Dec. 1981, pp. 104-108. How recursion in computer programs can be used to solve problems and puzzles.

Mace, Scott. "Papert Keynotes Computer Faire." *InfoWorld*, Mar. 1982, pp. 1 on. Seymour Papert, creator of the Logo language, will be the keynote speaker at the 7th annual West Coast Computer Faire.

Markoff, John. "Logo overturns old computer-education models." *InfoWorld*, Dec. 1981, p. 23. Discusses the Logo language and the support of Seymour Papert's work to develop Logo at MIT, by Texas Instruments.

McClees, Jock. "Where do Turtles Come From?" *Kilobaud Microcomputing*, Mar. 1982, pp. 99-100. Discusses the origins of the term "turtle" as it applies to robot research and describes the work done by the MIT Artificial Intelligence Lab.

Moursund, David. "Traditional computer assisted learning." *Computing Teacher*, Jan. 1982, pp. 3-4. The development of CAI and PLATO has become the standard against which microcomputer CAI systems are measured.

Muller, James. "Logo for the Imagination." *InfoWorld*, Mar. 1982, p. 19. VIEWPOINT column suggests that many articles have not communicated very well the power of the language of Logo.

Nelson, Harold. "Dynaturtle (turtle graphics and Newtonian physics)." *Kilobaud Microcomputing*, Mar. 1982, pp. 104-105. Describes the dynamic turtle developed by Andy DiSessa to help children and graduate students learn about classical physics.

Nelson, Harold. "Logo for personal computers." *BYTE*, June 1981, pp. 36-44. Reports on two versions of the Logo programming language now available for microcomputers.

Nelson, Harold. "Logo: Not Just for Kids." *Kilobaud Microcomputing*, Mar. 1982, pp. 96-107. Discusses the Logo programming language developed by Seymour Papert at M.I.T.

Nelson, Harold; Friedman, Rich. "Seymour Papert: spearheading the computer revolution." *OnComputing*, Summer 1981, pp. 10-12. Profiles Seymour Papert and his work in the area of computers and education.

Rousseau, Joseph; Smith, Stephen. "Whither goes the turtle?" *Kilobaud Microcomputing*, Sept. 1981, pp. 52-55. Discusses the work of Jean Piaget and Piaget's theories.

Sandifur, Kathleen. "Graphics fundamentals." *BYTE*, Oct. 1981, pp. 284-300. Includes graphics concepts and four subroutines of the Hewlett-Packard Graphics/1000 software package.

Swaine, Mike. "Micro languages—a matter of style." *InfoWorld*, Dec. 1981, p. 18. A brief description of the computer languages that are found on microcomputers.

Swaine, Michael. "Young people race to the future on the shoulders of turtles." *InfoWorld*, Nov. 1981, p. 28. The use of the Logo and Pilot programming languages which use a triangle to draw graphics on a video display.

Thornburg, David. "Friends of the turtle." *Compute!*, Jan. 1982, pp. 76-78. An introduction to turtle geometry and turtle graphics that are available in several computer languages.

Thornburg, David. "LOGO and PILOT languages." *Compute!*, Mar. 1981, pp. 14-18. COMPUTERS AND SOCIETY column discusses the emergence of highly interactive languages like LOGO and PILOT.

Weir, Sylvia; Watt, Daniel. "LOGO: a computer environment for learning-disabled students." *Computing Teacher*, Feb. 1981, pp. 11-19. Describes the provision of an innovative computer-based learning environment for learning disabled students in grades 5-8.

Weir, Sylvia. "Logo and the exceptional child." *Kilobaud Microcomputing*, Sept. 1981, pp. 76-84. Describes how the Logo programming language is used with students who are educationally handicapped.

Wise, Deborah. "1982 to be the year of the Turtle?" *InfoWorld*, Jan. 1982, p. 15. Demonstration of turtle graphics done by Seymour Papert for a program in the Sunrise Semester TV series.

Wolz, Ursula; Williams, David; Bird, Pristen. "Text editing practice; Calculator activities; Problem-solving." *Electronic Learning*, Nov./Dec. 1981, pp. 76-78. TECHNOLOGY IN THE CLASSROOM covers: teach Logo text editing; math operations vs. the simple algebraic logic system; problem-solving activities. **C**

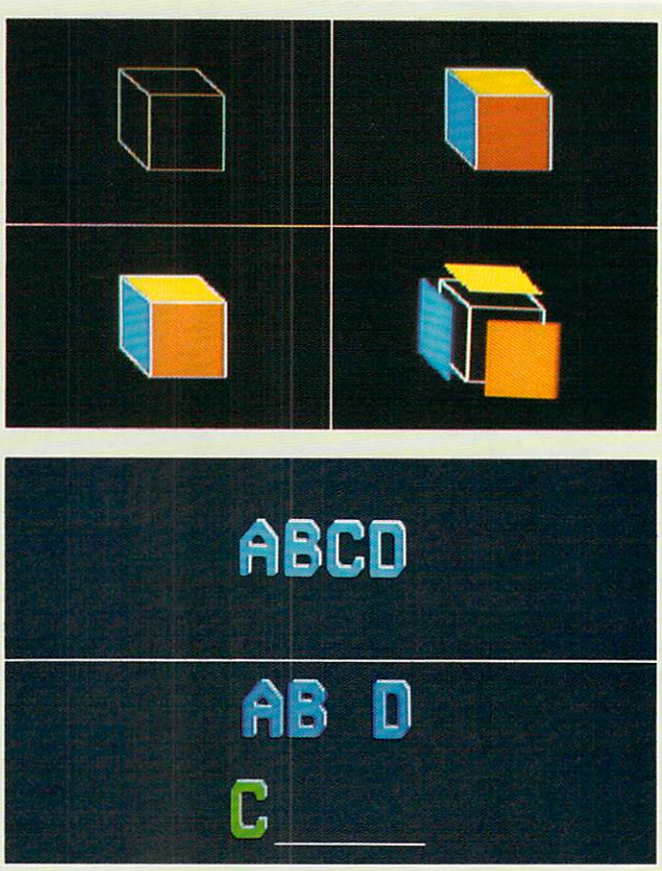
PILOT for the Commodore 64

BY APRIL M. KOPPENHAVER
AND STEPHEN MURRI

Today we are faced with sophisticated students who demand challenging computer applications and exciting graphics. PILOT for the Commodore 64 combines the unique features of the Commodore 64 with the power of the PILOT language, thus producing the most powerful tool available for the creation of quality educational software. PILOT for the Commodore 64 was written by Larry Kheriaty and George Gerhold, who originally authored COMMON PILOT for Western Washington University. The Commodore version of PILOT includes the abilities to define your own characters, create colorful movable objects called sprites, and to create music and a variety of sound effects.

PILOT for the Commodore 64 is a language that allows teachers to create their own Computer Assisted Instructional (CAI). With PILOT, educators can write their own educational software, combining fun, entertainment, and learning. As Dr. Dan Kunz, Commodore's Educational Program Manager, says, "PILOT frees the creative teacher from the problems associated with programming and courseware development. PILOT allows teachers to develop their own educational courseware specifically related to their own instructional strategies." Programs previously written in COMMON PILOT can be run with PILOT for the Commodore 64 without converting any code. Also, a "run only" version of PILOT, PILOTR, is included on the PILOT for the Commodore 64 disk. PILOTR is identical to PILOT except that programs can only be loaded and run (i.e., the user cannot create, edit, print, or save PILOT programs when running PILOTR). PILOTR is especially useful when teaching students with previously written PILOT programs.

PILOT is a full-scale computer language containing a complete and powerful instruction set. This provides the flexibility to program virtually any application. There is also a provision for integrating your own machine language subroutines. Using subroutines, external devices such as video-tape or laser-disk can be linked to the PILOT system.



Use PILOT's graphics commands to draw lines.

Next, sprites are combined to form the panels of the cube.

Sprite animation is used to create movement on the screen.

The labels on the panels show PILOT's ability to combine high-resolution graphics and text on the same screen.

Eight sprites make up these four letters, created using Commodore PILOT. Each letter uses two sprites overlaid to create a three-dimensional effect.

Here the two "C" sprites have changed location and color.

All screens by Stephen Murri. Special thanks to Carol Carr for PILOT technical assistance.

The PILOT system consists of four modes:

EDIT—To create and edit PILOT programs

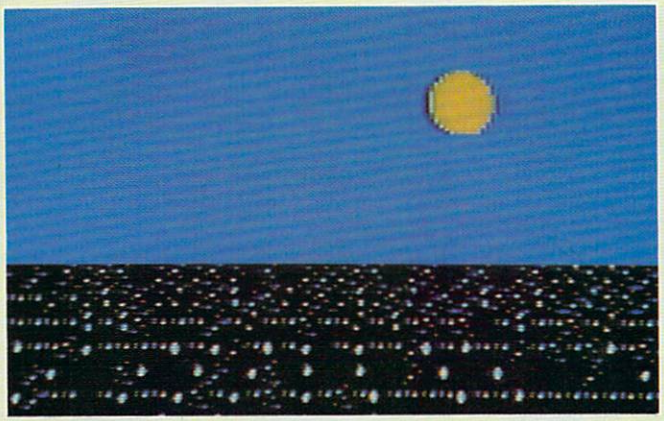
RUN—For program execution

IMMEDIATE—To enter individual instructions and view the results on the screen

COMMAND—For loading, storing, and printing programs

A maximum of *two keystrokes* is required to access the different PILOT modes. Press RUN/STOP to enter the COMMAND mode before accessing any of the other three modes. To enter the EDIT, RUN, or IMMEDIATE mode, simply input the mode's first letter.

Every learning situation requires some form of two-way communication. The PILOT system was designed to create programs that interact with students in a non-threatening, friendly manner. PILOT programs have the flexibility to determine what the user is *trying* to say, regardless of the spelling or sentence structure. Also, PILOT programs are advanced enough to determine if the user is close to an answer; thus, PILOT can provide a series of conversational *hints* to direct the student to the correct answer. By interacting with the computer through a program created with PILOT, the student is *learning* as he or she goes along. This is in direct contrast to the



Using Commodore PILOT, we first split the screen to create the effect of a "horizon". Then programmable characters are defined and placed on the screen to create the city lights effect. The moon is made up of two sprites.

Combine sprites using PILOT to create a planet and use programmable characters for the stars.

conventional one-way quiz, in which answers are either right or wrong.

PILOT is also beneficial to the teacher and offers substantial recordkeeping capabilities. A PILOT program can keep track of the number of times it takes a student to reach the correct answer, as well as maintain records of each student's score to monitor their progress.

You may have perceived PILOT to be only a text-based language . . . this is not so. Graphics are essential parts of effective instructional programs and PILOT includes several features to accomplish this. Text and graphics can be displayed simultaneously in any combination of sixteen colors. Graphic images can be created through use of plot, draw and fill features. You can even split the screen in a variety of locations to create "window" areas for specific purposes.

Thus far, we have discussed the features of PILOT in a very general way. Now let's take a look at the features which make PILOT for the Commodore 64 the most powerful version of PILOT available.

PILOT for the Commodore 64 allows you to easily program your own characters. This feature provides enormous flexibility in the creation of graphic displays. Programmable characters are especially important for those applications that require specialized notation. This includes language, science, and mathematics.

Your graphic capabilities are further enhanced with the ability to easily create and use sprites. Sprites are graphic images that you define and can move anywhere on the screen. Especially suited for video graphics and arcade-type animation, a maximum of eight sprites can be displayed on the screen at any given time.

Sprites can be single-colored or multicolored and can be expanded in the horizontal and/or vertical directions. Sprites can be combined to create large, colorful displays. Also, to give a three dimensional effect to an image, sprites can be assigned priorities which enable them to pass in front of or behind either each other or background data. Included on the PILOT disk is a "sprite editor" program which allows you to easily create and view sprites before using them in your PILOT program.

PILOT on the Commodore 64 allows you to create an immense variety of sounds and music by providing full access to the 6581 Sound Interface Device (SID). Sound is thoroughly explained in the PILOT manual, which includes music note values, charts, and a register map. Also included is a discussion of bits, bytes, and registers, which must be understood in order to use this feature. The PILOT package also contains a sample "sound editor" program that allows you to experiment with various sounds that may be fun to include in your programs. The sound feature is strictly an enhancement and is not required to effectively use the PILOT program.

In summary, there is no other language more suited to the educational environment than PILOT for the Commodore 64. The interactive features of PILOT, in combination with the unique features of the Commodore 64, make this package the most complete and effective instructional aid available.

For a more detailed technical discussion with references to actual lines of code, please see "Understanding the Pilot Language," which will appear in the next issue of *Commodore*. **C**

BY LEN LINDSAY
Author of *The COMAL Handbook*

In Len Lindsay's opinion, COMAL is the ideal language to teach beginning programmers. Whether they agree with Len's ideas or not, educators who offer courses in programming will find much useful information here to help them clarify their goals.

My five year-old daughter, Rhianon, has her own computer in her bedroom. She likes to play some of the simple games and type on the keyboard to spell words on the screen. This fall she will go to school to learn reading, writing and arithmetic. She will also learn about computers.

As more and more primary and secondary schools begin to include computer studies in their curricula, more educators are faced with the responsibility of selecting the proper computers, programming languages and software to best meet their school's needs. I would like to offer some guidelines here to help educators make a responsible choice if they plan to teach programming as part of their computer studies programs. Just buying a dozen computers and using the language that comes free with them is not necessarily a responsible method. Just a few of the many points that should be considered are listed below.

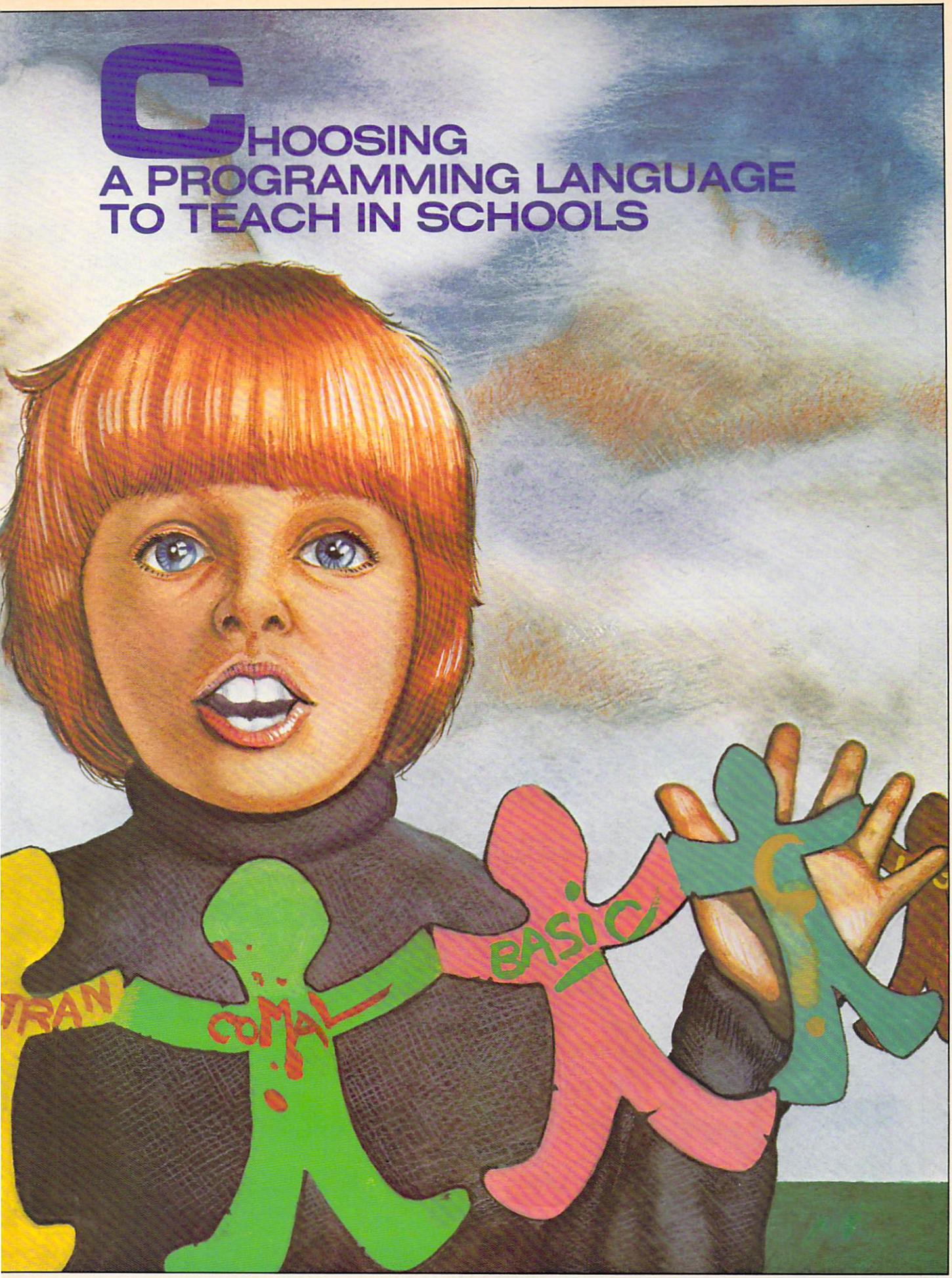
1. The computer system used should be affordable for two reasons. First, some students may wish to get a similar computer to use at home. Second, it is preferable to have one computer for each student, so all students can simultaneously try an exercise presented by the teacher. The cost of two dozen computers can be too high for the school if an expensive computer is used. Special networking systems are also available that allow one disk drive and printer to be shared by several students, helping to keep costs down and reduce the number of diskettes needed. It may also be an advantage if the computer chosen can run several languages designed for use by beginners.
2. The programming language should be easy to learn and use. Some students may not be naturally adept at programming, and the language should not scare them away. Try to avoid a language, for instance, that requires many steps just to run a simple program to print the student's name on the screen. Try to find a language with a built-in editor and command language, allowing everything to be done from within the language.
3. The programming language should adequately prepare the more advanced students for future programming courses, which most likely will include a structured language with modular programming.
4. The programming language should support general educational objectives. This includes solving problems and communicating ideas.
5. A program listing should be able to be read by each student in the class, not just the one who wrote it. Each student can then learn from the work of the others.
6. The programming language should have a clear definition. This means it should not vary significantly from computer to computer. Its instructions should be reasonably consistent, regardless of what computer you use.
7. There should be a choice of textbooks or teaching aids available to support the language.

A look at the possible choices

As you probably have gathered, I think the programming language, rather than the computer used, is the most important consideration for educators. Therefore, it seems reasonable to choose the language first, then find an affordable computer system that implements it. Better yet, choose a system that includes all of the languages designed for beginning programmers.

There are hundreds of programming languages. Each language has its own special strengths and weaknesses. To aid your search, let's review the characteristics of some of the most popular languages being taught today.

CHOOSING A PROGRAMMING LANGUAGE TO TEACH IN SCHOOLS



Illustration—Jack Freas

Chart of Languages and Points to Consider

* = Very Good + = Good – = Inadequate

Points to Consider	BASIC	PASCAL	C	COBOL	FORTRAN	FORTH	PILOT	LOGO	COMAL
1) System Cost	*	–	–	–	–	+	+	+	*
2) Easy to Learn/Use	*	+	–	+	–	–	*	*	*
3) Prepare for Future	–	*	*	+	–	*	–	–	*
4) General Education	–	*	*	*	+	+	*	*	*
5) Readable Programs	–	+	+	+	+	–	+	*	*
6) Well Defined	+	*	*	*	*	*	+	+	*
7) Textbooks/Aids	*	*	+	*	*	*	+	*	*

Each language in the chart is briefly explained below.

BASIC

BASIC is the language most often included with a microcomputer system. But because it is included with the system does not necessarily mean it is the best language to teach beginning programmers. True, BASIC is a general purpose language designed for beginners and has many teaching aids available. But it has no clear definition, so there are many differences from one BASIC to another. If it is written with great care, a BASIC program can be read by other students, but it is not reasonable to assume that students will take great care with their programs. Many will simply aim for a program that works, readable or not.

PASCAL

PASCAL is a popular advanced scientific programming language. PASCAL has a clear definition, but the most popular implementation does not follow it. It is a powerful language and supports general educational objectives. If a special utility is used to nicely list the programs with the structures properly indented, the programs can be easy to read. However, PASCAL is generally not a language for a first-time computer user. Beginners will most likely be scared away or confused by its details and rules. Thus, while it is a very good language for advanced programming, it will probably not work as a general purpose beginner's language.

C

This language is similar to PASCAL, and is most often associated with system programming and the UNIX operating system. It shares PASCAL's advantages as well as its disadvantages, but does not have as many rules, and makes extensive use of abbreviations. Due to its design, it functions better on a 16-bit rather than 8-bit system. It, too, would probably not be suitable for a beginner.

COBOL and FORTRAN

These languages are well established in the business and engineering fields. They are clearly defined powerful languages, but beginning students would undoubtedly be put off by their complexities. So neither COBOL nor FORTRAN

Buyer's Guide to COMAL Systems

COMAL-80 Interpreters

Versions available for Commodore computers:

1. CBM COMAL-80 version 0.12: Introductory COMAL.
   - A) Disk Loaded, runs on any PET/CBM with 32K and 4.0 ROMs.
   - B) Disk Loaded, runs on the Commodore 64.
2. CBM COMAL-80 version 1.02: Full COMAL with enhancements.
   - A) Disk Loaded, runs on the CBM 8096.
   - B) Disk Loaded, runs on a PET/CBM with Z-RAM board.
   - C) ROM board, installs in any PET/CBM except original PET.
3. CBM COMAL-80 version 2.00: Full COMAL with more enhancements.
   - A) Disk Loaded, runs on the CBM 8096.
   - B) Cartridge for the Commodore 64.
   - C) ROM update to ROM board (2C).
   - D) Deluxe ROM/RAM board for any PET/CBM except original PET.
4. 8 VIC 20 computers with time sharing disk drive and printer. COMAL and BASIC available.

Available from:

*Reston Publishing sells a disk with versions 1A and 2A plus procedure library and sample programs for $15. Also available with COMAL HANDBOOK for $29.95.

*COMAL CATALYST sells a COMAL Starter Kit with versions 1A and 2A plus procedure library, sample programs, binder, and newsletter subscription for $45.

*Ellis Horwood sells a disk with version 1A plus sample programs, price unknown. Also available with BEGINNING COMAL book, price unknown. Also version 2C ROM Board for 195 British Pounds.

*COMAL Interest Group sells a User Group Disk with versions 1A and 2A plus sample programs for $12.95. Also version 2C ROM Board with COMAL HANDBOOK for $350.00. Also distributes Reston Publishing and Ellis Horwood books and disks.

*Micro Computer Systems Ltd. sells the VIC system (4).

*Versions not mentioned in the list above have uncertain distribution arrangements. Next issue I hope to include a more up-to-date availability list.

COMAL Books

1. **COMAL Handbook** by Len Lindsay, 334 pgs. Spiral bound, $18.95 or with disk $29.95. Reston Publishing. A reference book for the COMAL programmer. Sample procedures and programs using CBM COMAL.
2. **Beginning COMAL** by Borge Christensen, 332 pgs. Soft bound, 10 English Pounds (also available with disk, price unknown). Ellis Horwood Ltd. A tutorial for the first time computer user introducing programming with COMAL. CBM COMAL used for examples.
3. **Structured Programming with COMAL** by Roy Atherton. 266 pages. Hard bound $50; soft bound $25. Ellis Horwood Ltd. Text on structured programming using COMAL. Examples using CBM COMAL and RC COMAL.
4. **Foundations in Computer Studies with COMAL** by John Kelly. To be published January 1983 by Education Company of Ireland, price not listed.
5. **Microcomputers in Education** published by Ellis Horwood has 2 chapters on COMAL: COMAL—an Educational Alternative, and Software Standards in BASIC and COMAL. Price $49.95, hard bound.

would be a wise choice for a first language.

FORTH

FORTH is becoming a popular language with professional programmers and seems to be a good choice for real-time process control systems. It has a well defined kernal and is very fast. It is possible to find one package that includes an editor, compiler, and interpreter. But FORTH, a hybrid between assembly language and the other high level languages, is not meant to be a language for beginners.

PILOT and LOGO

Each of these languages appears to be acceptable to beginners. Each is more or less defined and both are highly regarded for their use of turtle graphics. PILOT is useful for text manipulation, LOGO for symbol manipulation. PILOT, however, is generally intended to help teachers design programs for the students to run, rather than to teach students programming. Although teaching materials are available and the languages support general educational objectives, they are somewhat lacking in the power needed for some applications and it is questionable if they adequately prepare students for future programming courses. Even so, they each could have a place in a computer studies curriculum.

COMAL

Although COMAL may be a new language to American schools, it has been around since 1974, and already is the official language taught in all schools in both Denmark and Ireland. It is very easy to learn and use and supports general educational objectives. It is structured, so it adequately prepares students for future programming studies. Program listings are very readable, with all structures automatically indented. It has a clear definition, the COMAL KERNAL, and includes most of the power of PASCAL. Best of all, it is affordable—CBM COMAL versions 0.12 and 1.02 are in the public domain (read that FREE). Two new versions will run on a Commodore 64 computer.

Summary

BASIC, PILOT, LOGO, and COMAL are easy to learn and use, and thus each would be a possible choice for beginners and non-professional programmers. To give students an advantage in future studies it would probably be advisable to have students exposed to these several different programming languages. That way they can learn there can be more than one acceptable way to solve a problem. Just as a person who can speak three foreign languages in addition to English has an advantage in many situations, students who can use a variety of computer languages gain valuable programming flexibility.

A reasonable course of action would be to teach one good general purpose language first (COMAL would be a good choice), and present additional languages to all interested students after that (BASIC, PILOT, and LOGO). Most computer systems will support several languages, so this would be possible without having to buy different computer systems for each language. Here is where Commodore computers shine above the others. All four easy-to-learn-and-use languages are available for the PET, CBM, and the Commodore 64, which presents a cost-effective solution to the problem of choosing a programming language.

Newsletters/Magazines

1. **COMAL CATALYST.** 6 issues per year newsletter, $20. (or as part of $45 COMAL Starter Kit).
2. **COMAL BULLETIN.** 6 issues per year newsletter, 13.5 British Pounds. Published by Ellis Horwood Ltd.
3. **Riomhiris na Scol,** 6 issues per year journal, 20 Irish Pounds.

Users Groups

1. **COMAL Users Group, USA:** Contact Len Lindsay, 5501 Groveland Terrace, Madison, WI 53716
2. **COMAL Users Group, England:** Contact John Collins, 4 Grimthorte House, Percival St., London, EC1V OB5, U.K.
3. **COMAL Users Group, Alberta, Canada:** Contact

Recommendations

It looks like CBM COMAL version 2.00 on cartridge with a Commodore 64 computer will be a "best buy" (additional 1541 disk drive would be desirable over a cassette unit). Get *BEGINNING COMAL* for your tutorial and *COMAL HANDBOOK* as your reference guide. Each student should have his or her own copy. Several copies of *STRUCTURED PROGRAMMING WITH COMAL* and *FOUNDATIONS IN COMPUTER STUDIES WITH COMAL* should be in your school library. Subscribe to the *COMAL CATALYST* (one subscription for each teacher, plus several for your library for students to check out). If you have a disk drive, get a disk along with the *COMAL HANDBOOK*, it will be worth it for the "procedure library" on the disk. A utilities disk from the *COMAL CATALYST* or COMAL Interest Group would also be recommended.

Next issue COMAL will be described in detail. If you can't wait until then, send $2 to *COMAL CATALYST* for a sample copy, which has plenty of information about COMAL. Or prepare yourself even more and get a copy of each of the books recommended above. You should have no problem getting your school library to get copies of the books (just show them this article).

Resource List

**COMAL Catalyst,* Publisher, Gerald Hasty, 101 Convention Center Drive, Suite 900, Las Vegas, NV 89109. Tel: 702-737-5670

*COMAL Interest Group, 505 Conklin Place, Madison, WI 53703

*COMAL Users Group, 5501 Groveland Ter., Madison, WI 53716. Tel: (608) 222-4432 (include Self-Addressed Stamped Envelope if reply requested).

*COMAL Users Group, John Collins, 4 Grimthorte House, Percival Street, London EC1V OB5, U.K.

*Databank Ireland Ltd., Nagor House, Dundrum, Dublin 14 Ireland. Tel: 951020

*Educational Company of Ireland Ltd., Ballymount Road, Dublin 12 Ireland

*Ellis Horwood Limited, Market Cross House, Cooper Street, Chichester, West Sussex, PO19 1EB, U.K.

*Gemini Microcomputers, Oakfield Corner, Sycamore Road, Amersham, Bucks HP6 5EQ, U.K.

*Instrutek, Christiansholmsgade, DK 8700, Horsens, Denmark. Tel: 05 61 1100

*Len Lindsay, 5501 Groveland Ter., Madison, WI 53716. Tel: 608-222-4432

*Metanic Aps, Kongeveien 177, 2830 Virum, Denmark. Tel: (02) 858284

*Micro Computer Systems Ltd., 103 O'Connell Street, Limerick, Ireland. Tel: 061-46755

*RC (RegneCentralen), Lautrubjerrg 1-5, DK 2750, Ballerup, Denmark. Tel: (02) 658000

*Reston Publishing, 11480 Sunset Hills Road, Reston, VA 22090. Tel: (703) 437-8900

Riomhiris na Scol,* Colaiste an Spioraid Naomh, Bishopstown, Cork, Ireland **C

Illustration—Robert Neumann

COMPUTERS IN SCHOOLS: A TEXAS PERSPECTIVE

BY SANDRA K. PRATSCHER

Introduction

by William F. Kernahan Executive Vice President Regency Educational Systems

The schools in Texas, as the article to follow indicates, are actively implementing microcomputers at all levels of education. Such rapid implementation of any new device or idea is bound to change. This is especially true as the microcomputer enters classrooms that are staffed by educators not yet trained in the technology and who have little experience with either the hardware or its related software.

The very real danger is that the microcomputer, a device that holds unlimited promise for education, could easily slide into either misuse or non-use simply because of lack of adequate support for the person on the firing line—the teacher.

In Texas, the state is attempting to keep the "microcomputer revolution" from being a failure by designing a support system that will help the local school districts cope with this new technology. The article that follows, "Computers In Schools: A Texas Perspective", explains in detail the steps being taken by the Texas Education Agency.

In order for them to be successful, the TEA truly needs the assistance of manufacturers like Commodore. Recognizing this, we at Regency made a request of David Rosenwald, Director of Education Marketing for Commodore, to make an equipment grant to the Texas Education Agency and the twenty Regional Education Centers located throughout the state. The purpose of the grant is to place Commodore 64's in each location to assist the leadership at the state and regional levels to accomplish two goals—to train teachers and evaluate software.

Commodore agreed and each of the twenty-one locations in Texas will receive the following:

Commodore 64
Commodore Color Monitor
C-1541 Disk Drive
C-1530 Datassette
C-1600 VICMODEM
"Amazing 656" Public Domain Software

In turn, Regency has agreed to train personnel in each of the locations in the proper and effective use of the equipment. We feel this is the best example of the good that can happen when a manufacturer, dealer and institutional end-user work together to address a common need.

Something very exciting is happening in Texas schools today.

- In one Texas high school, a student was discovered hiding in the computer math class at 2 a.m. His reason? He was working on a programming project and wanted to spend extra time on the computer.
- In a Texas elementary classroom, a computer-using teacher reports that her only complaint with the new computers in her classroom is that she has a difficult time getting youngsters to stop working with them and move on to their noncomputer-based lessons.
- In one regional education service center, computer-related workshops fill before any other training sessions offered. In fact, many teachers are put on waiting lists for the computer sessions that the ESC offers.

The microcomputer has enrolled in Texas schools, and its presence has created a great deal of excitment, en-

thusiasm, and sometimes confusion across the state.

Buzz words like computer-based instruction, computer literacy, word processing, and computer-assisted instruction are fast becoming a part of the vocabulary of many Texas educators. Planning groups and user groups are being organized, and school personnel look to these new instructional devices as a way to bring relevant, effective instruction to students.

In the 1982 Texas Education Agency (TEA) study of accreditation of textbook preliminary results pointed out the following:

- Of the campuses surveyed, 40 percent of the principals said they were using computers in instruction on their campuses.
- Ten percent of the teachers said they were using computers to support instruction.
- More than 60 percent of the surveyed districts support computer literacy requirements for teachers.

Clearly, there is a great deal of computer-based activity in the state, and indications are that the rapid proliferation of this tool for instruction will continue. In fact, a recent report of the Texas Legislative Committee on Public Education states that nationwide annual expenditures on computer-based educational materials will rise to $500 million by 1985, an increase of 300 percent over current levels. Indeed, the activity in Texas is consistent with what is happening on a national and even international basis.

In one Texas high school, a student was discovered hiding in the computer math class at 2 a.m. His reason? He was working on a programming project and wanted to spend extra time on the computer.

TEA Activity

The level of activity in computer-based instruction across the state is also reflected at the state education agency. Two years ago, the need for leadership from the state level was recognized when Grace Grimes, deputy commissioner for professional development and support, established an ad hoc committee on computers in education. Its purpose was to develop short- and long-range objectives for the agency and propose actions and placement of responsibilities to reach these objectives.

In April 1981, the committee wrote its preliminary report, which outlined problems and needs in computer-based education and proposed several recommendations for action. Through that report, many actions have been initiated, and several statewide efforts are under way.

One area of concern that the committee identified was the problem in selecting computer systems for instruction. The importance of careful and thoughtful planning for computer-based instruction before hardware purchase is acknowledged by many. Yet, few documents exist to help schools plan for effective implementation of this new tool.

To address this need, the ad hoc committee wrote a guide, which is now available to Texas schools. (This guide, as well as other documents related to computer-based instruction, may be obtained by contacting Sandy Pratscher, Education Specialist for Instructional Computing, TEA, 201 East Eleventh, Austin, TX 78701).

Computer-Related Competencies

Most Texas teachers, like those across the nation, obtained their degrees B.C. (that's "before computers"). They require a great deal of computer-based training before effective use can be made of the computer for instruction. Happily, for Texas students, teachers are becoming increasingly interested in acquiring computer literacy. Dr. George Culp, assistant director of the computation center at the University of Texas at Austin, reports a tremendous growth in the number of students registering for his educational computing classes. This semester he has more than 100 students enrolled in one evening course.

The need for inservice and preservice education for teachers is evident, and though many Texas teachers are seeking that training, the scope and degree of training necessary is not always well defined. To help clearly identify those computer-related competencies needed by Texas teachers, TEA is working on its second document. This booklet will list the teacher competencies necessary for effective use of computers in instruction. This list is being developed with the help of a grassroots advisory committee, composed of computer-using teachers and other educators currently involved in teacher training efforts.

Once that list is completed, workshop designs and strategies will be developed as a way to guide schools, service centers, and other institutions of learning in the delivery of this training.

The need for training exists not only for teachers, but also for Texas Education Agency staff. To that end, TEA has established a microcomputer resource center in which professionals may receive training, review software, and use existing systems to help manage their own work.

One problem in dealing with a state the size of Texas is keeping informed on current levels of activities. Compound that problem with the rapid proliferation of computer use in education, and the result is a Texas-sized problem, indeed. In order to stay in touch with current computer-activity levels in the state, the

University of Texas at Austin, in cooperation with TEA staff, is in the process of collecting data about computer use in Texas schools.

Agency staff are also requesting that schools actively involved in computer-based learning send this author a short summary of their activities. It is our hope that a data base of this information (by ADA, program type, numbers of students served, grade levels, etc.) can be maintained, so that school personnel can communicate with one another about common problems and solutions. (If your school is actively involved in computer-based instruction, please contact the author at TEA.)

Statewide Network for Courseware Evaluation and Training

Recognizing that the effective use of the computer in instruction requires good quality software that has been reviewed, the Texas Education Agency has funded a statewide network for courseware evaluation. This project is managed through the Region IV Education Service Center in Houston and involves the evaluation of several hundred courseware packages by hundreds of computer-using educators in local independent school districts across the state. The results of the courseware evaluations are placed in each of the twenty intermediate agencies in Texas, so those districts planning to purchase software can have access to the evaluations.

Each of the twenty education service centers across the state has designated a consultant, who is responsible for directing the courseware evaluation effort. These representatives of the education service centers also meet periodically for training in computer-based instruction. These training sessions not only enhance the consultants' ability to evaluate courseware, but also serve as a way of increasing their expertise in computer-based instruction, so they can deliver required training to teachers across the state.

In each of these training sessions, the representatives from the education service centers are sharing information about the training they are doing in their specific regions. They are also provided with training materials, so training can be duplicated across the state. In order to facilitate both the courseware evaluation and training efforts each center has established a microcomputer resource center similar to the one at the Texas Education Agency. These facilities are used for courseware evaluation and training for educators from each regional area of the state. The goal is to have each of the regional service centers play an important role in the future, as technology centers in education, so training and courseware evaluation efforts can be maintained.

The project coordinator for the Texas Education Computer Cooperative Courseware Evaluation Project is Vicki Smith at the Region IV Education Service Center in Houston. She can be contacted for additional information at: Region IV Education Service Center, 7200 W. Tidwell, Houston, TX 77001.

Computer Literacy Objectives

House Bill 246, a sweeping revision of essential curriculum in the state of Texas, has provided the Texas Education Agency staff a unique opportunity to provide curriculum information regarding computer literacy.

At initial public hearings for House Bill 246 across the state, participants were asked whether they thought computer literacy should be incorporated into the existing curriculum. Overwhelmingly, the answer to the question was yes. Yet few people questioned seemed to have a clear definition of computer literacy.

In an effort to begin to give direction to computer literacy as a curriculum component, TEA staff has begun writing a series of strands of computer literacy objectives and some optional models for delivery. The strands of objectives identified are:

- Computer-related terminology and computer use;
- The history and development of computing devices;
- Using the computer as a tool;
- Communicating instructions to the computer;
- Computer-related occupations and careers.

These objectives will be reviewed in public hearings along with other House Bill 246 objectives, and, based on feedback from reviewers, they will be revised and sent to Texas schools.

Legislative Activities

The importance of computer technology in public education has not been overlooked by the state's lawmakers. A report on changing technology has recently been released. This report is the result of the work of the subcommittee on changing technology and represents many months of study and careful consideration. Subcommittee members are William W. Blanton, chairman, representative from Carrollton; Mike Richards, senator from Houston; and Dr. Calvin E. Gross, superintendent from from Alamo Heights in San Antonio.

The report contains not only a description of technology in Texas schools but also a series of recommendations to the central education agency regarding computer-based instruction. The recommendations include:

- An emphasis on the importance of science, mathematics, and technology instruction;
- Provisions to encourage access to computer literacy for all children in Texas;
- A review of teacher certification requirements and inservice training to reflect the importance of the technology in teaching and learning;
- Funding priorities for computer-based instruction and teacher training;
- Leadership for and coordination of statewide efforts by the central education agency.

National Cooperation

Texas is also involved in a nationwide cooperative effort to share information and support regarding computer-based instruction. Project "Best" is a project funded through the Department of Education. Its purpose is to facilitate communication among state departments of education, particularly in the use of technology to support basic skills instruction. The project makes available to participating departments teleconferences, electronic bulletin boards, and electronic mail services to help support needed communication.

The importance of the computer in Texas schools is being recognized by many as one of the most significant and impactful developments to occur in many years. Certainly, it has educators across the state talking, and, more noteworthy, it has caused many of us to begin to plan for ways we can use technology for students.

We have discovered a tool that shows a great deal of promise, both for improving the way of delivering instruction to students, and as a means of more effectively preparing our students to live in a highly technological environment.

C

Sandra Pratscher is an education specialist for instructional computing at the Texas Educaton Agency and vice-president of the Texas Computer Education Association.

How to SHOP

Educational administrators have been trying to thrive and survive with the latest rage of the computer age—the microcomputer. The earlier generations of these electronic marvels were appropriate only for instructional use, primarily because of their small data storage capacity. But some microcomputers now have the capacity of many older full-size computers. Thus they are now useful for performing certain administrative functions in education. Keep in mind, however, that not every task is best performed by a computer. So, before making any decisions, carefully outline your needs and take a little time to become computer literate. It could save you money, time and agony later.

There are four situations that generally justify the use of a computer:

1. When massive amounts of data are to be processed through well-defined operations.
2. When processing is going to be highly repetitive.
3. When processes are to be repeated many times under a variety of conditions.
4. When speed of processing is of great importance.

When you begin looking for software to automate your functions, consider general applications software first, rather than software designed specifically for education administration. General applications programs are of three basic types: data management systems, electronic spread sheet systems and word processing systems.

Illustration—Jack Freas

for Administrative Software

Data Management Systems

A data base is the computer equivalent of a filing cabinet full of information. You create your own data base and store information in it. You can retrieve information to either refresh your memory or generate reports. A good data manager will let you generate reports, without programming, in the format most useful to you. It should make sense to you even when you know nothing about computers or programming.

Some characteristics you should look for when you shop for a data manager are:

1. User-oriented operating procedures.
2. Procedures that initiate each routine from a menu (list) of selections.
3. Reasonable requirements for the amount of main memory and external storage you need in order to use the program.
4. Flexibility in setting up a format for entering and retrieving information.
5. Specialized formats for entering dates, monetary quantities and decimal specifications.
6. Storage procedures that store each field of data only once.
7. File maintenance procedures for adding, deleting, updating, inspecting and scanning records.
8. Sorting procedures in ascending and descending order for multiple data fields.
9. Expandability for field sizes, record sizes and file sizes.
10. Arithmetic capability with computed fields.
11. Options to generate reports either on the video display or printed on paper (hard copy), and to break out reports by column specification, totals by field, final totals, and by columnar or other vertical headings.
12. Access to different parts of each record by people with different levels of security authorization.
13. Access to the files created by the data management system through other BASIC programs.

Data management systems are often unappreciated by microcomputer buyers, even though they are very useful tools. To determine the many potential uses of such a system you should probably try one out. In a school setting a good data manager can be useful in instructional management, inventory and property records, media center management and student records.

Electronic Spreadsheet Systems

Any problem that can be solved with a calculator, a pencil and a sheet of paper can be solved with an electronic spreadsheet system—only faster and more accurately. These spreadsheets are also excellent tools for things like budgetary forecasting.

The electronic spreadsheet is a matrix of columns and rows. The intersections of these columns and rows define thousands of positions into which you can enter a number, an alphabetic title or a formula to be calculated. If you wish, for example, you can make your electronic spreadsheet look just like your school's monthly budget/expenditure report. The rows could be defined as combination function-object budget entities and the columns could be defined as (1) allocated budget amounts, (2) expenditures for the month, (3) expenditures for the year-to-date, (4) outstanding encumbrances or commitments and (5) unencumbered and uncommitted balances that can be appropriated.

Now, suppose you want to see what would happen to your entire budget if changes are made to any function-object entities on this spreadsheet. Just make the changes and the computer will automatically recalculate all the other numbers that are affected and generate an entirely different budget report. If these same calculations were done manually, hours of time would have to be invested to obtain the same results.

A typical spreadsheet has about 63 columns and 254 rows with which to work. Within these parameters you should be able to create just about any format you desire and perform just about whatever calculation or manipulation of figures you need. The video screen becomes the window through which you view the parts of the spreadsheet you need to see. You should also be able to split the screen horizontally or vertically to view two parts of the spreadsheet simultaneously, and scroll the screen horizontally or vertically to view any one section of the sheet. You might say electronic spreadsheets turn a sea of data into data you can see.

Word Processing Systems

When you "process words" on a microcomputer you do things like compose text on the video screen, store it on a floppy disk, recall it from memory later, edit it and produce it in final form on paper. Word processing software is designed to make all these procedures easy to perform on your computer.

The computer offers a radical change from the usual manual mode of word processing. Using the manual method all corrections and modifications must be done on paper. A succession of different versions means a succession of paper copies. Correcting one little typing mistake could mean producing an entire new page. If you use computerized word processing, however, you can do all correcting, editing and modifying in the computer's memory before you commit anything to paper. After all the changes are incorporated, the production of an error-free page then only takes a minute or so on your printer.

Proposals, contracts, forms, reports and some letters can often contain a great deal of standardized text or "boilerplate," that requires only minor changes to fit individual circumstances. Word processing allows you to perform quick surgery on the boilerplate—cut out inapplicable words, phrases, sentences and paragraphs and substitute others—so there is no need to retype the entire document in order to individualize it. When the changes are made you can print out an original on paper. The skeleton boilerplate remains stored on a floppy disk for the next use.

The simplest word processing systems allow you to type text, correct typographical errors, store the document as it appears on your display and then automatically type it out in its correct form. Good word processors often include the following features:

1. Adding or deleting text anywhere in the document.
2. Correcting text errors.
3. Moving sections of text to other locations (cut and paste).
4. Searching for, deleting and/or replacing designated key words, sentences or paragraphs anywhere in the text.
5. Composing letters by selecting any number of standard paragraphs from a master file of paragraphs.
6. Creating and justifying left and right margins.
7. Centering of lines between left and right margins.
8. Centering of text vertically on the page.
9. Tabulating and indenting.
10. Automatic formatting of lists and tables.
11. Document page formatting: length, number and spacing of lines.
12. Automatic placement of header and footer on a page.
13. Automatic pagination and dating.
14. Video viewing of text as it will later appear on paper.
15. Creation of mailing labels.

General Considerations

When you are selecting an applications software package you should also consider the following general criteria.

General Questions

1. Does the software do what it claims to do?
2. Can it be returned for a full refund within thirty days? If not what other satisfaction can a dissatisfied purchaser receive?
3. Does the company make adequate provisions for back-up copies? The best plan is to have a back-up copy available on site, but the company should at least agree to send you one in overnight mail.
4. If the software is to be used at multiple sites, will the company sell multiple copies at a discount? Terms on volume purchases are definitely negotiable.
5. Is any user training provided as part of the purchase price? The more complicated the function being performed, the more important user training, support and service become.
6. Are support and service available from the supplier through a toll-free number?
7. Does the software provide for some form of security, especially for financial and confidential student data? Several levels of security might be desirable and could be incorporated into the software at little extra cost.
8. Is the program integrated with any other application program? The data you use in one program may be needed in another.
9. Is the software flexible enough to meet your school or district needs, and yet not so complex as to be confusing?

Documentation Questions

1. Is the documentation (manuals, instruction booklets, etc.) provided with the software complete, clear, understandable, well-organized and indexed? Do not underestimate the value of good documentation. On the other hand, do not let the appearance of good documentation mislead you into believing the software is good.
2. Does the company provide updates to the documentation and software periodically and at little or no extra cost? A good company reviews its software periodically and should provide revisions to the user at little or no extra cost. Revisions that correct errors in previous editions should be provided at no cost.
3. Does the documentation provide a way for the user to calculate the number of records that can be physically stored on one floppy or hard disk? You should be able to determine whether a floppy or hard disk configuration would be able to best handle the functions being considered.

Input Questions

1. Are input fields well-defined and self-prompting? When the software is requesting data, no confusion should exist about what kind of data is to be entered, where it is to be entered, and at what time.
2. Are potential errors at the time of input well-diagnosed and described in an understandable way? The software should not signal an error down in the machine-language jungle if all you did was type in a letter where you were supposed to type in a number or if you made some other reasonable input error.
3. If the data could be used for several different programs, can it be done without manually re-entering the same data?

Processing Questions

1. Is the software menu-driven? And if so, is the order of selection on the menu the same as the sequence of operation of the programs in normal use? A menu-driven software package is composed of a number of separate programs to perform different tasks. The user selects which of these tasks are to be performed from a list referred to as a menu.
2. When the software is running, does it give the user feedback about what part of the processing is taking place? During the operation of the software, pauses of varying lengths may occur while the microcomputer is processing data. During these pauses, some kind of feedback should be given to the user on the video to indicate that processing is proceeding normally.

Output Questions

1. Does the design and format of information produced by the software meet district, state, and/or federal requirements so that this information will not have to be entered onto other forms by hand?
2. Upon output of the information, does the user have the option to have it printed on paper or displayed on a video screen? If the output can be reviewed on a video before it is printed, the time required to run the printer, as well as the paper consumed, can be saved while it is reviewed for errors.
3. Are reports formatted in an easy-to-read manner using appropriate abbreviations, spacing, and print size?

If you find that general applications software cannot be adapted to fit a specific administrative need, a package designed especially for educational administration may be your best solution. A variety of programs exist in areas such as intructional management, student record-keeping, grade reporting, attendance monitoring, inventory records and school budgeting. A list of some of these programs is included in the following article. **C**

Software

Designed Specifically for Educational Administration

This list, compiled by Commodore education specialist Pat Walkington and educator Pat Kuhn, is not meant to be a be-all and end-all reference. It does, however, provide you with a good idea of the kinds of administrative software available for Commodore computers.

Name and Address of Company	Program	Micro	Memory	Drive	Language	Program Capacity	Documentation	Licensing Back-Up Policy	Comments	Cost
ATTENDANCE										
American Peripherials 122 Bangor St. Lindenhurst, NY 11757	Micro Attendance Package	8032	32K	8050	BASIC	1700	60 pages	not copy protected	dealer phone-in consultant	$800
CERF 1441 N. 27th La. Phoenix, AZ 85009	BU-2 Package	8032	32K	8050 Hard	BASIC	900 floppy disk 3000-5000 hard disk	112 pages very basic	copy protected back-ups provided with program	dealer phone-in consultant	$2,500
Evans Newton Inc. 7745 E. Redfield Rd. Scottsdale, AZ 85260	Project Attend	8032	32K	8032	BASIC		198 pages very basic	copy protected user made back-ups leased indefinitely to user	dealer phone-in consultant	$4,995
Microphys 2048 Ford Street Brooklyn, NY 11229	Attendance System	8032	32K	8050	BASIC		yes	copy protected		$600
Rediker Software (413) 566-3495	Attendance	8032 4032	32K	8050 4040	BASIC	2,450 students 820 students	yes	protected ROM		$850
BUDGET										
COMM*DATA Box 325 Milford, MI 48042	Financial Report	4032 8032	32K	4040 8050	BASIC	depends upon drive	yes	copy protected		
VISI CORP. 2895 Zanker Dr. San Jose, CA 95134	VisiCalc	4032 8032	32K	4040 8030	BASIC	depends upon drive	yes	copy protected	easy electronic worksheet	
Personal Software 1330 Bordeaux Dr. Sunnyvale, CA 94086	VisiCalc	8032	32K	4040 8050	BASIC	depends upon drive	yes	copy protected		$200
CMS Software 2204 Camp David Mesquite, TX 75149	General Ledger	4032	32K	4040 8050	BASIC	depends upon drive	yes	copy protected		
Info Designs 6905 Telegraph Rd. Birmingham, MI 48010	Accounts Rec. & Pay. General Ledger	Commodore 64	64K	1541	Compiled BASIC	100-350 records	yes	protected		$199
M.A.G. P.O. Box 346 Athens, GA 30603	Fund Accounting	8032	32K	8050	BASIC	pre-set	yes	security key		$2,000

Name and Address of Company	Program	Micro	Memory	Drive	Language	Program Capacity	Documentation	Licensing Back-Up Policy	Comments	Cost
BUDGET (continued)										
Creative Software 230 Caribbean Dr. Sunnyvale, CA 94086	General Ledger	4032 8032	32K	4040 8050	BASIC	depends upon drive used	yes	copy protected	multi-purpose	$200
Sheridan College 1430 Trafalgar Rd. Oakville, Ontario Canada	Divisional Budget	4032 8032	32K	4040 8050	BASIC	depends upon drive	yes	copy protected back-ups used		
Micro Learning Ware P.O. Box 2134 Mankato, MN 56001	Accounting Package	4032 8032	32K	4040 8050	BASIC	depends upon drive used	yes	copy protected back-ups	easy to use	$24.95
Pacific Coast Software 3220 S. Brea Canyon Rd. Diamond Bar, CA 91765	CalcPack	Commodore 64								
INSTRUCTIONAL MANAGEMENT										
Evans Newton Inc. 7745 E. Redfield Rd. Scottsdale, AZ 85260	Project Test Project BASIC	4032 8032	32K	4040 8050	BASIC	4040 80 students 8050 320 students 3000 objectives	yes	copy protected user made back-ups	card reader opt.	$1995
Holt, Rinehart & Winston 383 Madison Ave. New York, NY	Class Management System	4032 8032	32K	4040 8050	BASIC	depends upon drive used	yes	copy protected back-ups		
Sheridan College 1430 Trafalgar Rd. Oakville, Ontario Canada	Class Mark Tracking	4032	32K	4040	BASIC	depends upon drive used	yes	copy protected back-ups		$100
Prescription Learning Corp.	In-Lab Management	4032	32K	4040	BASIC	4040 400 students per disk	yes	copy protected back-ups	This is a lab package purchase program. Includes hardware, software, maintenance and in-service consultant.	
Micro Computer Industries 1520 East Mulberry, Suite 110 Ft. Collins, CO 80524	Create-a-Base	8032	32K	4040 8050	BASIC	depends upon drive used	yes	copy protected	easy to use data base	$360
Merlan Scientific 247 Armstrong Ave. Georgetown, Ontario Canada	Class Manager	4016 4032	16K 32K	4040 8050	BASIC	depends upon drive used	yes	copy protected back-ups		$75
Competitive Software 21650 Maple Glen Dr. Edwardsburg, MI 49112	Schoolpac-1	old 8K 4016 4032	8K 32K	4040 8050	BASIC	depends upon drive used	yes	copy protected		$35
Educational Activities P.O. Box 392 Freeport, NY 11520	Classroom Management	4032 8032	32K	4040 8050	BASIC	depends upon drive used	yes	copy protected	for library and audio-visual equipment	$98
Rediker Software (413) 566-3495				4040 8050		depends upon drive used			produces classroom report cards for 1800 students	
Commodore Business Machines 1200 Wilson Dr. West Chester, PA 19380 *available soon	EasyLesson* EasyQuiz*	Commodore 64		1541					keeps track of credits, produces class rank by GPA.	
INVENTORY										
Creative Software 230 Caribbean Dr. Sunnyvale, CA 94086	Inventory	4032 8032	32K	4040 8050	BASIC	depends upon drive used	yes	copy protected back-ups	easy to use	$200

Name and Address of Company	Program	Micro	Memory	Drive	Language	Program Capacity	Documentation	Licensing Back-Up Policy	Comments	Cost
Min. Comp. Systems Limited 5666 Stanley St. Halifax, Nova Scotia Canada Po3K26	Inventory Control Package	4032	32K	4040 8050	BASIC	depends upon drive used	yes	copy protected back-ups	multi-purpose	$170
Computer House Division 1407 Clinton Rd. Jackson, MI 49202	Inventory	4032	32K	4040 8050	BASIC	depends upon drive used	yes	copy protected		$95
B.P.I. 3423 Guadalupe Austin, TX	Inventory Control	8032	32K	8050	BASIC	depends upon drive used	yes	protected by hardware device		$400
Southern Solutions P.O. Box P McKinney, TX 75069	Information not available									
SCHEDULING										
Mount Castor Industries 368 Sharp Street Amherst, MA 01002	The Scheduler	8032	32K	8050	BASIC	1500 floppy disk	112 pages very basic yes	copy protected back-ups with program	dealer phone-in consultant	$495
CERF 1441 N. 27th Lane Phoenix, AZ 85009	Scheduling Program	8032	32K	8050 + Hard Disk	BASIC	2000 dual disk	70 pages very basic yes	copy protected back-ups with program	dealer phone-in consultant	$2000
Microphys 2048 Ford St. Brooklyn, NY 11229	Scheduling Update System	8032	32K	8050	BASIC		yes			$500
STUDENT RECORDKEEPING AND STUDENT PROGRESS										
A B Computer 252 Bethlehem Pike Colmar, PA 18915	Flexfile	4032 8032	32K	4040 8050	BASIC	4040: 1000 records 8050 about 2800 records	yes	prints files with report writer or a mail label routine—easy to use data base	$75	$110
Canadian Micro Distributors, Ltd. 365 Main Street Milton, Ontario L9T1P7	Manager	8032	32K	8050	BASIC		yes			$250
Evans Newton 7745 E. Redfield Rd. Scottsdale, AZ 85260	Project BASIC IEP TIME	4032 8032	32K	4040 8050	BASIC	4040 80 students 750 objectives 8050 320 students 3000 objectives	yes	Card Reader Opt.		$1995
CFI Computer Solutions 875 West End Ave. New York, NY 10025	Ascert	8032	32K	8050	Compiled BASIC	2,000 records multiple disk	yes			$495
OTHER										
A B Computer 252 Bethlehem Pike Colmar, PA 18915	Papermate Flexfile	4016 4032 8032	16K 32K	4040 8050	BASIC	4040: 1000 records 8050: 2800 records	yes	copy protected back-ups user made	Papermate functions with 16K VIC, with any printer & cassette or disk	$75 $110
Professional Software Inc. 51 Fremont St. Needham, MA 02194	WordPro 3, 4, 5	8032	32K	4040 8050	BASIC	depends upon drive used	yes	copy protected uses ROM chip	excellent word processing programs	$375
Commodore Dealers	WORDCRAFT 80	8032	32K	4040	BASIC		yes	copy protected		$395
Computer House Division 1407 Clinton Rd. Jackson, MI 49202	Mailing List	8032	32K	4040	BASIC	depends upon drive used	yes	copy protected		$80
Microphys 2048 Ford St. Brooklyn, NY 11229	Mailing List Telephone Directory	4032 3032	32K	4040	BASIC	depends upon drive used	yes	copy protected		

SUBJECT-ORIENTED EDUCATIONAL SOFTWARE for COMMODORE COMPUTERS

COMPILED BY COMMODORE EDUCATION SPECIALISTS

Although there are many ways to use computers in the classroom, most schools right now seem to be using them mainly for Computer Assisted Instruction (CAI). The usual format of CAI is drill-and-practice, although programs may vary widely in their approach to that format. This list gives you an idea of where you can purchase various kinds of CAI software for Commodore computers. The first part lists vendors by subject and grade-level categories. The second part provides each vendor's address.

Who Supplies What for Whom

Mathematics

Primary Grades

American Peripherals
Comaldor Software
Compu-Mania
Courseware Magazine
Cow Bay Computing
Educational Activities
JMH Software
K-12 Micromedia
Kidstuff
Micro-Ed, Inc.
Microcomputer Workshops
Micrograms, Inc.
Milliken Publishing Co.
Queue
Right On Programs
Robbinsdale Project
Scholastic Materials
Teachers Pet Software
Teaching Tools

Intermediate Grades

American Peripherals
Comaldor Software
Compu-Mania
Courseware Magazine
JMH Software
K-12 Micromedia
McGraw-Hill
Micro-Ed, Inc.
Microcomputer Workshops
Micrograms, Inc.
Midwest Visual
Milliken Publishing Co.
Program Design, Inc.
Queue
Right On Programs
Robbinsdale Project
Scholastic Materials
Sterling Swift
Teachers Pet Software
Teaching Tools (NC)

Junior/Senior High

American Peripherals
Brain Box
Briley Software
Educational Activities
K-12 Micromedia
Micrograms
Microphys
Midwest Visual
Robbinsdale Project

Reading

Primary Grades

American Peripherals
Courseware Magazine
JMH Software
Learning Tree Software
Micro-Ed, Inc.
Micrograms
Midwest Visual
Queue
Robbinsdale Project
Teaching Tools

Intermediate Grades

American Peripherals
Brain Box
Courseware Magazine
Educational Activities
K-12 Micromedia
Micro-Ed, Inc.
Midwest Visual
Program Design, Inc.
Queue
Robbinsdale Project
Wiz Kids, Inc.

Junior/Senior High

American Peripherals
Brain Box
Educational Activities
K-12 Micromedia
Micrograms, Inc.
Midwest Visual
Robbinsdale Project

Language Arts

Primary Grades

American Peripherals
Comaldor Software
Courseware Magazine
JMH Software
K-12 Micromedia
Micro-Ed, Inc.
Micrograms, Inc.
Queue
Robbinsdale Project
Scholastic Materials
SLED Software
Teaching Tools (California)

Intermediate Grades

American Peripherals
Brain Box
Comaldor Software
Courseware Magazine
Educational Activities
JMH Software
K-12 Micromedia
Micro-Ed, Inc.
Micrograms, Inc.
Midwest Visual
Program Design, Inc.
Queue
Right On Programs
Robbinsdale Project
Scholastic Materials
SLED Software
Teaching Tools (California)
Wiz Kids

Junior/Senior High

American Peripherals
Brain Box
Education Activities
K-12 Micromedia
Krell Software
Micrograms, Inc.
Microphys
Midwest Visual
Robbinsdale Project

Who Supplies What For Whom (continued)

Science

Primary/Intermediate

Brain Box
Compu-Mania
Educational Activities
Micrograms, Inc.
Right On Programs
Robbinsdale Project
Scholastic Materials

Junior/Senior High

American Peripherals
Brain Box
Conduit
Creative Computing
Educational Activities
K-12 Micromedia
Merlan Micro Series
Micrograms, Inc.
Microphys
Midwest Visual
Pasco Scientific
Queue
Right On Programs
Robbinsdale Project

Social Studies

Primary/Intermediate Grades

American Peripherals
Comaldor Software
Compu-Mania
Creative Computing
Island Software
Micro-Ed, Inc.
Micrograms, Inc.
Right On Programs
Robbinsdale Project
Scholastic Materials

Junior/Senior High

American Peripherals
Brain Box
C.E.R.F.
Comaldor Software
Compu-Mania
Creative Computing
Island Software
Micro-Ed, Inc.
Micrograms, Inc.
Right On Programs
Robbinsdale Project

Administrative Management

A.B. Computers
Canadian Micro Distributors
Computer Marketing Services
Personal Software
Professional Software

Library Management

Right On Programs

Networking

A.B. Computers
Canadian Micro Distributors
Computer Marketing Services
Commodore Business Machines
Copperstate Cash Register
Cyberia, Inc.
Interface Dynamics
Neeco
Questar International
Skyles Electric

Utilities

Commodore Business Machines
Computer House Division
Madison Computer
Palo Alto ICS
Professional Software

Student Record Keeping

American Peripherals
C.E.R.F.
Comaldor Software
Educational Activities
Evans Newton, Inc.
Info Designs
Merlan Scientific
Micrograms
Microphys
Midwest Visual

Test Quiz Generation

American Peripherals
C.E.R.F.
Chadsworth Card Reader
Comaldor Software
Compu-Mania
Evans Newton, Inc.
Micrograms, Inc.
Queue

Courseware Authoring Packages

Academic Computing Association
Greenwood Software

Where to Find Suppliers

A-1 Computer Service
7103 W. Clearwater
Kennewick, Washington 99336
(509) 783-4980

AB Computers
252 Bethlehem Pike
Colmar, PA 18915
(215) 822-7727

Academic Computing Association
P.O. Box 27561
Phoenix, AZ 85061
(602) 244-8355

American Peripherals
122 Bangor Street
Lindenhurst, NY 11757
(516) 226-5849

BrainBank
70 East 10th Street
New York, NY 10003
(212) 777-7035

Briley Software
P.O. Box 2913
Livermore, CA 94550
(415) 455-9139

C.E.R.F.
1441 N. 27th Lane
Phoenix, AZ
(602) 269-2411

Canadian Micro Distributors Ltd.
365 Main Street
Milton, Ontario L9T 1P7
(416) 878-7277

Comaldor
P.O. Box 356/PS 0
Toronto, Ontario M4A 2N9
(416) 751-7481

Comm Data Computer House
Box 325
Milford, MI 48042
None Listed

Comm*Data Systems
P.O. Box 325
Milford, MI 48042
(313) 685-0113

Commodore Business Machines
1200 Wilson Drive
West Chester, PA 19380

Competitive Software
21650 Maple Glen Drive
Edwardsburg,
MI 49112
(616) 699-7115

Compu Mania
18285 Hwy 108
Jamestown, CA 95327
None Listed

Computer House Division
1407 Clinton Road
Jackson, MI 49202
(517) 782-2132

Computer Marketing Services
300 W. Marlton Pike
Cherry Hill, NJ 08002
(609) 795-9480

Conduit
P.O. Box 338
Iowa City, IA 52244
(319) 353-5789

Copperstate Cash Register
3125 E. McDowell Road
Phoenix, AZ 85008
(602) 244-9391

Courseware Magazine
N. Milbrook, Suite 222
Fresno, CA 93726
(209) 227-4341

Cow Bay Computing
P.O. Box 515
Manhasset, NY 11030
None Listed

Creative Computing
P.O. Box 789-M
Morristown, NJ 07960
(201) 540-0445

Cyberia
2330 Lincoln Way
Ames, IA 50010
(515) 272-7634

David Blake Systems
14563 Bexhill Court
Chesterfield, MO 63017
(314) 532-0980

Educational Activities
P.O. Box 392
Freeport, NY 11520
(516) 223-4666

Educational Connection
1508 Coffee Road, Suite J
Modesto, CA 95355
(209) 576-1611

Evans Newton Incorporated
17335 E. Acoma Drive, Ste. 102
Scottsdale, AZ 85260
(602) 998-2777

Focus Media
135 Nassau Blvd.
Garden City, NJ 07950
Not Listed

Greenwood Software
1214 Washington
The Dalles, OR 97058
None Listed

Info Designs
6905 Telegraph Road
Birmingham, MI 48010
(313) 540-4010

Interface Dynamics
1494 Solano Ave.
Albany, CA 94706
None Listed

Island Software
P.O. Box 300
Lake Grove, NY 11755
None Listed

JMH Software of Minnesota
4850 Wellington Lane
Minneapolis, MN 55442
(612) 559-9208

K-12 Micromedia
P.O. Box 17
Valley Cottage, NY 10989
None Listed

Kidstuff
P.O. Box 345
Dedham, MA 02026
(617) 277-2456

Kinetic Designs
401 Monument Road #171
Jacksonville, FL 32211
(904) 725-0435

Krell Software
21 Milbrook Drive
Stonybrook, NY 11790
None Listed

Learning Tree Software
P.O. Box 246
King Park, NY 11754
(516) 462-6316

Merlan Scientific
247 Armstrong Avenue
Georgetown, Ontario
None Listed

Micro Learningware
P.O. Box 2134
Mankato, MN 56001
(212) 646-0140

Micro-Ed
P.O. Box 24165
Minneapolis, MN 55424
(612) 926-2292

Microcomputer Workshops
103 Puritan Drive
Port Chester, NY 10573
(914) 273-2209

Micrograms, Inc.
P.O. Box 2146
Loves Park, IL 61130
(815) 965-2464

Microphys
2048 Ford Street
Brooklyn, NY 11229
(212) 646-0140

Midwest Software
P.O. Box 214
Farmington, MI 48024
None Listed

Midwest Visual
6500 North Hamlin
Chicago, IL 60645
None Listed

Milliken Publishing Co.
1100 Research Blvd.
St. Louis, MO 63132
(314) 991-4220

Mount Castor Industries
386 Shays Street
Amherst, MA 01002
(413) 253-3634

NEECO
679 Highland Ave.
Needham, MA 02194
Not Listed

Osborne/McGraw-Hill
2600 Tenth St.
Berkeley, CA
(415) 548-2805

Paperclip
71 McCaul Street
Toronto, Ontario M5T 2X1
(416) 596-1405

Pasco Scientific
1933 Republic Avenue
San Leandro, CA 94577
None Listed

Personal Software
1330 Bordeaux Drive
Sunnyvale, CA 94086
(408) 745-7841

Program Design
11 Idar Court/Dept. 110
Greenwich, CT 06830
(203) 661-8799

Programs for Learning
P.O. Box 954
New Milford, CT 06776
None Listed

Queue
5 Chapelhill Drive
Fairfield, CT 06423
None Listed

Questar International
7270 Woodbine Avenue
Markham, Ontario
(416) 490-8044

Right On Programs
P.O. Box 977
Huntington, NY 11743
None Listed

Robbinsdale Project
4148 Winnetka
Minnesota, MN 55427
None Listed

Scholastic Inc.
50 West 44th Street
New York, NY 10036
None Listed

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

SLED Software
P.O. Box 16322
Minneapolis, MN 55416
(612) 926-1735

Sterling Swift Publishing Co.
1600 Fortview Road
Austin, TX 78704
(512) 444-7570

Steven M. Thorpe Co.
1229 West 4000 North
Pleasant View,
UT 84404
None Listed

Sunburst Communications
Room VF 414,
39 Washington Ave.
Pleasantville, NY 10570
None Listed

Tamarack Software
Water Street
Darby, MT 59829
(406) 821-4596

Taylormade Software
P.O. Box 5574
8053 E. Avon Lane
Lincoln, NE 68505
(402) 464-9051

Teachers Pet
1517 Holly Street
Berkeley, CA 94703
None Listed

Teaching Tools
P.O. Box 50065
Palo Alto,
CA 94303-0065
(415) 493-3477

Teaching Tools
P.O. Box 122679
Research Triangle Park,
NC 27709
(919) 851-2374

Tycom Associates
68 Velma Avenue
Pittsfield, MA 01201
(413) 442-9771

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

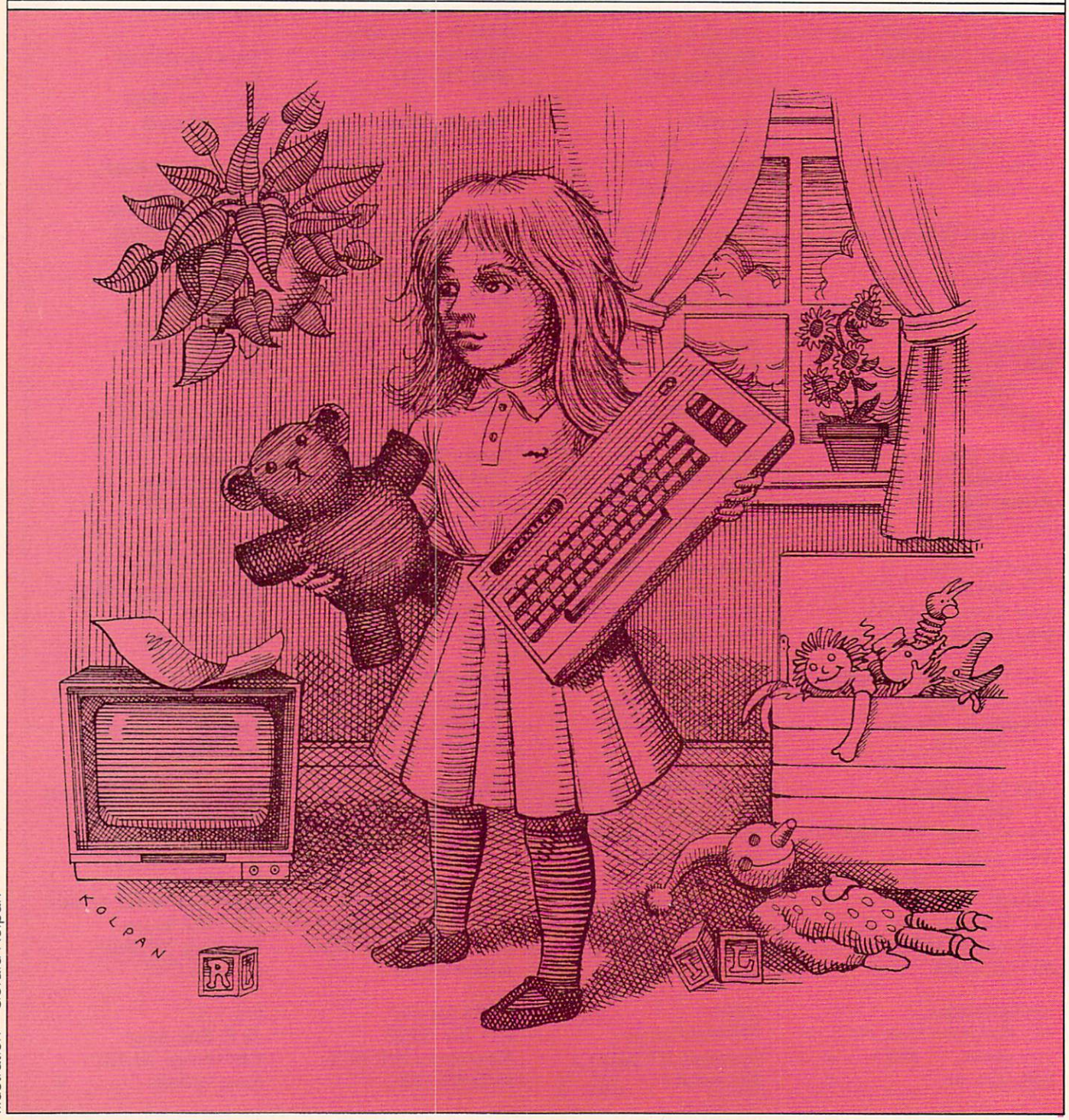
United Software of America
750 Third Avenue
New York, NY 10017
(212) 682-0347

Wiz Kids
76 E. Park Street
East Orange, NJ 07017
None Available **C**

you and your computer

The conclusion of Doris' Commodore 64 users guide for children. Parts 1 and 2 appeared in the March issue of *Commodore*.

BY DORIS DICKENSON



Illustration—Gerald Kolpan

part 3 Number Know-How

Your computer likes to work with numbers. You are going to find out some of the things it can do.

Turn on your computer and monitor (or T.V.)

A. Addition

1. Type 5 + 7 =
   Press RETURN
2. Type 4 + 6 =
   Press RETURN

Are you disappointed that it worked like a typewriter and only printed what you typed?
Don't be! Try this.

3. Type this—PRINT 5 + 7 (Notice that you can forget about the = sign)
   Press RETURN
4. Type PRINT 4 + 6
   Press RETURN

Surprise! The word PRINT without the quotation marks around the numbers, told the computer to do the addition for you.

5. Try adding larger numbers.
6. Try adding more than two numbers at a time.

How's that for speedy addition!

B. Subtraction

1. Type PRINT 15 – 5
   RETURN

Isn't that easy? It works the same as addition.

2. Practice some other subtraction problems.

C. Multiplication

Before you start, you need to know that the * key takes the place of X (the times sign) that you write in your problems.

1. Type PRINT 5 * 7
   RETURN

How's that for speed!

2. Practice multiplying some larger numbers, but don't forget to start with PRINT.
3. Try multiplying more than two numbers together.

D. Division

There is no division mark on the computer keyboard, either. The / underneath the question mark tells the computer to divide.

1. Type PRINT 45 / 5
   RETURN

Wasn't that easy?
Try dividing some other times-table facts.

2. Now see what happens when a number cannot be divided evenly.
   Type PRINT 31 / 7
   RETURN

The remainder in division, on the computer, will always be written as a decimal fraction. Sometimes it will be a very long number. You could shorten it by rounding off.

3. Practice more division problems.

E. Exponents

1. Type PRINT 12*12*12*12*12
   RETURN

When you want to multiply a number *by itself* a certain number of times, *you* would write it with an exponent (the number of times written after and above the number) such as 12^5. The computer uses the ↑ key (up arrow) to write what you mean.

2. Type PRINT 12 ↑ 5
   RETURN
3. Practice multiplying other numbers by themselves. Remember to use the ↑ key.

F. Mixed operations

Now that you can add, subtract, multiply, and divide, you can put them all together.

1. Type PRINT 3 + 5 − 7 + 2
   RETURN

2. Try putting some larger numbers together, but *don't* use any commas in the numbers. You will only confuse the computer.

3. Order of operations
   Get a paper and pencil.
   Write down 20 + 8 / 2 and work out the problem.
   Was your answer 14?
   Or was it 24?
   Or was it both, depending on what you did first?

   Type PRINT 20 + 8 / 2
   RETURN
   On the computer it will *always* be 24 because—
   A computer will work across a row from left to right and will perform operations in the following order:
   a. any exponentiation (numbers times themselves) from left to right,
   b. any multiplication and division, from left to right,
   c. any addition and subtraction, from left to right.
   Now try these problems and see if you can guess what the answer will be before you let the computer figure it out. Remember to type PRINT and RETURN each time you do the problem on the computer.

4. Practice:
   a. 33 / 3 − 5
   b. 12 − 3 * 2 + 5
   c. 313 − 2 + 10 / 2
   d. 6 * 4 − 8 / 2 + 2
   There *is* a way you can get the computer to do things differently. If you put an operation in parentheses ( ), the computer will do that operation first, before it does anything else.
   Type PRINT (20 + 8) / 2
   RETURN
   What did you get for an answer? Look back at what you got the first time you tried this problem.
   Look back at Practice 2, above.
   Type PRINT (12 − 3) * 2 + 5
   RETURN
   Now type PRINT (12 − 3) * (2 + 5)

G. Printing problems

You can combine what you learned in this section with what you learned about PRINT statements in Part 2.

1. Suppose you want the computer to print out the problem as well as work out the answer. Do you think you could do that without help? Try this. Have the computer write this problem and answer it:
   5 X 9 =

2. Need some help? Try this
   Type PRINT "5 * 9 =";5 * 9
   RETURN
   Remember—
   " " quotation marks let the computer print what they enclose
   ; semicolon causes the next part of a statement to be printed right after the previous part (remember how your name looked when followed by a semicolon?)
   the computer will work out a problem after a PRINT statement

3. Practice
   Have the computer *print out* and answer a variety of arithmetic problems.

Computer License Test
Test 3
Number Know-How

You must be able to have the computer print out the following problems and figure out the answers:

1. 16 + 58 =
2. 8 + 29 + 33 =

3. 15 − 6 =
4. 321 − 158 =

5. 6 X 47 =
6. 139 X 26 =

7. 74 ÷ 13 =
8. 56 ÷ 7 =

9. (33 − 2) * 3 / 2 =
10. 16 − 4 * 5 + 7 =

Student name: _____

Test date: _____

_____

_____

Date passed (100%) _____

Approved by _____

part 4

Datassette Instructions

Everyday when you wake up, you know you remember many things that you learned and did before that day.

Every time you turn off the computer, it "forgets" all the things you told it or did with it. When you turn it on the next time, you have to give it new instructions all over again.

You can imagine what a waste of time and effort this is!

One way of saving your computer programs is to record them on tape in a machine that works like a tape recorder. It is called a Datassette.

If you follow these directions carefully, you can save each day's practice and programs to work with at some future time.

A. To SAVE a program on tape

1. After you type in a program, you will have to think of a name for it. You can call it anything, but you cannot use over 16 letters or spaces.
   When you have finished your work on it for the day, you may want to save it. Everything between where you are at that point and the last NEW will be saved on the tape.
   Note: Have your teacher/tutor make sure that the Datassette is plugged in correctly. All buttons should be in the UP position (recorder off)

Practice on the Datassette by typing these programs and following the directions carefully.

Clear computer screen.

2. Type
   NEW
   10 PRINT "THIS IS A TEST PROGRAM"
   20 PRINT "WE WILL SAVE OUR PROGRAMS"
   30 PRINT "FOLLOW DIRECTIONS"
   (We will call our program "TEST")
   SAVE "TEST"
   (remember to press RETURN)

3. The computer will answer with:
   PRESS PLAY AND RECORD ON TAPE
   You do just that! Press down both the RECORD (REC) and PLAY keys on the Datassette. Make sure they stay locked down.

4. The screen will go blank and turn the color of the border of the normal screen.
   After the program is saved, the screen will return to normal and the computer will say:
   O.K.
   SAVING TEST
   READY
   Press the STOP button on the Datassette.

5. Type in this next program
   NEW
   10 PRINT "I LIKE TO USE THE COMPUTER"
   20 PRINT "I TRY TO USE IT EVERYDAY"
   30 PRINT "PRACTICE MAKES PERFECT"
   SAVE "PRACTICE" (don't forget RETURN)
   Press PLAY and RECORD when the computer tells you to.
   Press the STOP button when the program is saved.

6. Now type in this program
   NEW
   10 PRINT "COMPUTER GAMES ARE FUN"
   20 PRINT "WE CAN LEARN FROM GAMES"
   30 PRINT "WE WANT MORE GAMES"
   SAVE "GAMES"
   Press PLAY and RECORD when the computer tells you to. Press the STOP button when the program is saved.

B. TO LOAD a program into the computer from tape clear your screen and type NEW. This will clear the memory inside the computer. Type LIST (and RETURN). This will show you that there is no program inside the computer.

1. *Press down the REWIND (REW) button on the Datassette.*
   Make sure the button is locked down. This rewinds the tape back to the beginning. Once the tape stops, press the STOP button.

2. Type
   LOAD "TEST" (RETURN)
   (If you forget the program name, just type LOAD and the first program will be loaded into the computer.)

3. The computer will answer:
   PRESS PLAY ON TAPE
   Do that!
   The screen will turn blank and the color of the border of the normal screen. When the program is found, the screen will say:
   FOUND "TEST"
   Press the C= key. The computer will respond with:
   LOADING
   READY
   Press the STOP button on the Datassette.

4. Now type LIST to make sure that your program is really in the computer.

5. Press REWIND on the Datassette. Type
   LOAD "PRACTICE" (RETURN)
   Press PLAY when the computer tells you to. The computer will tell you:
   SEARCHING FOR "PRACTICE"
   FOUND "TEST"
   Press the C= button and the screen will go blank again. The next time it stops, it will say:
   SEARCHING FOR "PRACTICE"
   FOUND "TEST"
   FOUND "PRACTICE"
   Now press the C= button once more. This time the computer will respond with:
   LOADING
   READY
   Press the STOP button on the Datassette. Now type LIST to make sure that the right program is really in the computer.

6. Press REWIND on the Datassette. Type
   LOAD "GAMES" (RETURN)
   Press PLAY when the computer tells you to.
   The computer will tell you:
   SEARCHING FOR "GAMES"
   FOUND "TEST"
   Press C=. The computer will say:
   SEARCHING FOR "GAMES"
   FOUND "TEST"
   FOUND "PRACTICE"
   Press C=. The computer will say:
   SEARCHING FOR "GAMES"
   FOUND "TEST"
   FOUND "PRACTICE"
   FOUND "GAMES"
   Press C= to load the program into the computer. When the computer is READY press the STOP button on the Datassette, and type LIST to make sure that the correct program is in the computer.

7. Rewind the tape and practice loading one or the other of these programs several times.

8. Just to see what happens, one of the times that you type LOAD, ask for the program without " " quotation marks around the name of the program. See how fussy the computer is?

C. You should be ready to practice on your own. Type in some programs of your own and try saving and loading them.

Computer License Test

Test 4

"I Remember You"—
Datassette Instructions

Write three short 3-line programs below. Give a name to each one.

(1)

(2)

(3)

Demonstrate to your teacher/tutor that you can save your programs on a Datassette tape and then load them back into the computer as your teacher/tutor asks you to.

Student name: _____

Test date: _____

_____

_____

Date passed (100%) _____

Approved by _____

What does **COMMODORE** have that Apple, Radio Shack and IBM don't?

backpack™

BATTERY BACKUP SYSTEM

For CBM™/PET® 2000, 4000, 8000, and 9000 series computers and CBM 4040/8050 dual disk drives. Installs within the cabinets of the computer and disk drive. Recharges continually from the machine's own power supply and automatically supplies 30 minutes (max.) of reserve power during outages. Also eliminates surges and spikes. User installable.

OFFICIALLY APPROVED BY COMMODORE

In Canada call: Van-Hoy Group (604) 542-1138 or (604) 545-0794
In United Kingdom call: Wego Computers (0883) 49235

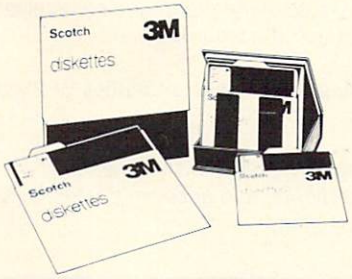
ETC

ETCETERA OF CSC CORPORATION
APEX, NORTH CAROLINA, U.S.A.
(919) 362-4200

SOLD ONLY BY INTELLIGENT COMPUTER DEALERS • DEALER INQUIRIES WELCOME

Scotch® Diskettes

record reliability



at the lowest price!

Call our Modem Hotline (anytime) – 619-268-4488 for exclusive monthly specials. Our free catalog contains more than 600 fantastic values.

ABC Data Products

(formerly ABM)

8868 CLAIREMONT MESA BLVD.
SAN DIEGO, CALIFORNIA 92123

ORDERS ONLY	ITT TELEX	INFORMATION
800-850-1555	4992217	619-268-3537

NEW! FOR YOUR CBM 4032, 8032, 8096 OR COMMODORE 64

PROOFREADING

SOFTWARE

NOW A 50,000 WORD DICTIONARY

CBM 8050 VERSION

Now you can rapidly eliminate misspellings from your word processing text. Spellmaster (TM) **is delivered with a 35,000 word dictionary,** allowing the user to add up to 15,000 words (CBM 8050 version). **Direct Screen Editing of Mistakes.** Words "suspected" to be incorrect are displayed "reverse video" on the screen; simply correct the mistakes and resave your corrected file. **Add Words to Dictionary with a Single Keystroke!** Spellmaster makes it easy to Permanently Add any correctly spelled work in your text to your User Dictionary. **100% Machine Language Speed** allows a large word processor textfile to be Proofread in 2 minutes or less. **Uses dictionary of LITERAL WORDS,** not imprecise rootword approach. **Specialized medical and legal dictionaries are available. Menu-Driven and User Friendly. Compatible with Commodore 1541, 2040, 4040 and 8050 Disc Drives.**

"HIGHLY RECOMMENDED"
Jim Strasma —
Micro Magazine Dec. '82

"SPELLMASTER IS AN EXCELLENT MACHINE LANGUAGE PROGRAM"
Robert Baker —
Micro Computing Magazine
Jan. '83

SPECIAL $89 CBM 64 VERSION

SPELLMASTER

by Dwight Huff & Joe Spatafora

OFFICIALLY APPROVED BY COMMODORE

TM

SPELLMASTER SYSTEMS SOFTWARE 1400 - 66th St. No. Suite 485 St. Petersburg, Florida (813) 347-6733 33710

For Management Only

* * * MAGIS-PLUS * * *

The Superior Management Tool and Micro Accounting Software

MAGIS-Plus is a fully integrated system and includes a General Ledger, Accounts Receivable, Accounts Payable, Payroll, Inventory—all on one program disk, requiring only one data disk.

MAGIS is presently being used in the floor covering industry, picture frame shops, florist shops, graphic art studios, furniture manufacturers, general merchandise stores, home decorating centers, computer centers, bakeries, auto parts stores and many other small business establishments.

Feature	Benefit
A Sales Journal (cash and credit)	updates all affected files including General Ledger, prints invoices
An Expense System (cash and credit)	updates all affected files in General Ledger and subsidiary ledgers
An Accounts Receivable System	generates statements, produces aging and delinquency reports, records payments to affected files
An Accounts Payable System	records payment of bills, prints checks, tracks delinquencies, handles anticipated purchase discounts
A Perpetual Inventory System	produces physical inventory sheet and reorder reports
An Inventory Transaction System	records purchase orders and merchandise received and updates all affected accounts
A Payroll System	computes payroll, prints checks, produces W-2's and quarterly reports
A General Ledger System	produces trial balance, financial statements, comparative statements, and accommodates direct General Ledger postings

MAGIS™

Merging computer technology with the practical worlds of business and government

OFFICIALLY APPROVED BY COMMODORE

The Management Accountability Group, Inc.
P. O. Box 346
Athens, GA 30603
(404) 353-8090

MAGIS is a registered trademark of The Management Accountability Group, Inc.

The VIC Magician

Writing Games In BASIC
Part 4 . . . Using the PEEK Command

by Michael S. Tomczyk

Our gamewriting series continues. This installment in our series focuses on the PEEK command, a "magical" command that lets you look into any of the computer's memory locations and PEEK at the contents.

By now you're pretty familiar with the POKE command. Well, PEEK is similar to the POKE command. The difference is that the POKE command lets you put a number value into a VIC memory location (for example, when you POKE a musical note value into one of the VIC's four "speaker" locations or when you POKE a graphic symbol onto the television screen).

The PEEK command lets you find out which number value currently resides in a memory location . . . and since a screen location is actually a memory location, you can PEEK into one of these locations to see which value is stored there, and thus detect which letter, number or graphic symbol is displayed on a particular spot on your TV screen. You can also PEEK at a special memory location to see which key is being pressed by the user. More on this in a moment—but for now just remember that there are two primary reasons for using the PEEK command in game animation:

1. To "peek" at a screen location to see which symbol is displayed there . . . for example, when you want to detect when two objects touch or "collide."
2. To "peek" at the keyboard to see which keys are being pressed. This lets you define games controlled by the VIC keyboard . . . it's also an easy way to check to see if the user is typing the correct number in a number game.

Controlling Games From the Keyboard

You can use the PEEK command to detect which key is being held down on the keyboard. This is very useful when you want to write a game requiring keyboard controls, or when you want to use the keyboard like a game "joystick."

There is a special memory location inside the VIC—location 197—which always holds the value of the key being held down. This information is described briefly on page 179 of the *VIC 20 Programmer's Reference Guide* . . . but how can you use this information in a game? Simple. The first thing you need to know is the PEEK key code of the 65 VIC keys you can use in a game (Note: there are 66 keys but SHIFT LOCK doesn't count as a PEEK key . . . also, in case you're wondering, the PEEK key codes are different from the POKE codes on page 141-142 of your VIC user's guide).

To see the PEEK codes for any key you press, type in this short program:

```
10 PRINTPEEK (197)
20 GOTO10
```

Now type the word RUN and press the RETURN key. What happens? The screen fills up with the number 64 . . . that's because location 197 always contains the number 64 when no key is being held down. Now, leave the program running and hold down the key with the letter "J" on it. Hey—the number on the screen changes to 20 as long as you hold the key down. That's because the PEEK code of the J key is 20. Try the letter "I" . . . value 12. Now the letter L . . . 21. And the letter M . . . 36. These four keys (I, J, L, M) are important because they are commonly used to control games from the keyboard when moving objects in four direc-

tions, up, down, left and right.

To help you find the PEEK values of these and other keys in the future, the PEEK values are printed here:

#	key	#	key	#	key	#	key
0	1	16	none	32	space	48	Q
1	3	17	A	33	Z	48	E
2	5	18	D	34	C	50	T
3	7	19	G	35	B	51	U
4	9	20	J	36	M	52	O
5	+	21	L	37	•	53	@
6	£	22	;	38	none	54	↑
7	DEL	23	[graphic]	39	f1	55	f5
8	←	24	STOP	40	none	56	2
9	W	25	none	41	S	57	4
10	R	26	X	42	F	58	6
11	Y	27	V	43	H	59	8
12	I	28	N	44	K	60	0
13	P	29	,	45	:	61	—
14	*	30	/	46	=	62	HOME
15	RETURN	31	[graphic]	47	f3	63	f7

Now let's proceed to the next step . . . using this information to control an object on your television screen! Hold down the RUN/STOP key and press RESTORE at the same time, then type the word NEW and press RESTORE at the same time, then type the word NEW and press RETURN. The NEW command erases the previous program.

Moving An Object Using the Keyboard

This program lets you move a ball around the screen using the I, J, L and M keys. Note that there are no limits in this program so you can actually move the ball off the screen and back onto the screen. Enter the program exactly as shown:

```
10 PRINT [SHIFT] [CLR/HOME] ;:P=7910:PC=P+30720
20 POKEP,81:POKEPC,5
30 IFPEEK(197)=12THENP=P-22:PC=PC-22:
   POKEP+22,32
40 IFPEEK(197)=20THENP=P-1:PC=PC-1:
   POKEP-1,32
50 IFPEEK(197)=21THENP=P+1:PC=PC+1:
   POKEP-1,32
60 IFPEEK(197)=36THENP=P+22:PC=PC+22:
   POKEP-22,32
70 GOTO20
```

Type RUN and press RETURN. Now type the I, J, L and M keys. Isn't this a neat program? Almost a game, huh? Here's a quick explanation:

Line 10: First we clear the screen. Then we define the variable "P" as screen location 7910, in the center of the screen, and PC is defined as the matching color setting, which is always the screen location plus 30720.

Line 20: Here we POKE a ball character into our P location which is the same as saying POKE7910,81. And POKEPC,5 is the same as saying, POKE7910+30720,5. We looked on page 140-143 of the VIC owner's manual to see that 81 is the POKE value of the ball symbol and number 5 is the color number for green.

Line 30: Follow closely on this one because this is the "key" to moving objects on the screen. The first part translates, IF the PEEK value in memory location 197 is 12 (the "I" key), THEN we redefine P as P−22 which is one screen location upward on your TV screen. We also redefine the color setting as PC−22 so it matches the new screen location. Finally, we erase the previous symbol from the screen by POKEing a space (POKE value 32) into our previous screen location which is now P plus 22 (P+22). The previous location is P+22 because the NEW value of P is P−22. Got it? Again . . . we read the keyboard to see if the "I" key is being held down. IF it is, and only if the I is being pressed, THEN we redefine P and PC as one location higher on the screen so pressing the I key appears to "move" our ball symbol up when you press it. Finally, we erase the ball from its previous position because otherwise the first ball would stay on the screen.

Lines 40-60: The same principles apply as in line 30. Line 40 is looking for the "J" key and if it's pressed, the new screen location becomes P−1 which is one location to the left on the screen. Line 50 looks for the "L" key and if that key is pressed then P is redefined as P+1 which is one space to the right. And Line 70 checks for the "M" key and if that key is pressed, the new P position is P+22 which is one space down. Note that you could also have diagonal movements . . . for example P+23 is a right downward diagonal movement.

Line 70: Finally, we GOTO 20 because line 20 is the line where we POKE the new P and PC position. Going back every time to line 20 is an efficient programming technique . . . otherwise we would have to POKE a new ball and color position in every IF . . . THEN statement on lines 30-60. So the program structure looks like this:

1. Clear the screen and define screen and color position of P which is initially defined as 7910.
2. POKE a green ball into the position defined as P.
3. Check the I, J, L and M keys and redefine P and PC accordingly (i.e. up, left, right, down).
4. Go back to line 20, which POKEs the green ball into a new position IF one of these four keys was pressed.

Let's keep going . . . how about if we POKE an object on the screen in a stationary position . . . say, a diamond? Next, let's say that whenever the ball you control from the keyboard contacts the diamond, the diamond turns into a hollow ball!

What do we have to do to our program to make this happen? First, we need to find out the POKE codes of the diamond and hollow ball. Turn to page 140-141 in your VIC owner's guide and you'll see that the POKE code for the diamond is 90 and the code for the hollow ball (circle) is 87. The next thing we want to do is POKE a hollow ball somewhere on the screen.

Hold down the RUN/STOP key and press RETURN, then type the word LIST and press RETURN to see your program. Cursor down to the bottom of the screen and type these lines:

```
15 H=90:HP=7732:HC=HP+30720
25 POKEHP,H:POKEHC,2
35 IFPEEK(P-22)=90THENH=87
```

Now type RUN and press RETURN. You should see a green ball in the middle of your screen and a diamond near the top slightly to the left. Now, using the J, K, L and M keys, move the green ball above the diamond, then come down directly on top of it. Now approach it from the LEFT, then from the RIGHT, then go down directly below it and approach it from the bottom up . . . presto!

When you hit the diamond from the BOTTOM side, it magically turns into a hollow ball. How did we do this? In a nutshell here's what we did . . .

Line 15: We defined a variable H as the number 90, which is a diamond. HP became the screen position of the diamond and HC was the matching color setting.

Line 25: If you substitute the numbers from line 15 for the "H" variables we use here, you can see that this line really means: POKE7732,90:POKE7732+30720,2.

Line 35: Here, we use the PEEK command to look at the location directly above the position of the ball. IF the POKE value of that position is 90 (a diamond) THEN we change the value of H to 87, which is a hollow ball. Notice that P−22 will always be the screen location one space above the ball. Note also that we check for POKE values of symbols when

PEEKing at a screen location (because that's the value the screen location contains), but we check PEEK key values when PEEKing at the keyboard to see which key is pressed (because location 197 contains a PEEK key value which is different from a the POKE value of a symbol).

A Word About Game Design

Before we begin to design our first game, we need a game design or blueprint to work from. This design may take the form of a written outline, graphic storyboard, or programming flowchart. The programming approach can take several forms but the two most common approaches involve either designing the entire game in as much detail as possible before you start programming or writing the smallest possible nucleus program and building the game by adding enhancements like music, sound effects, etc.

Designing a game is almost as important as programming it and in fact most professional game companies have game designers as well as programmers. In the past, many games were designed by programmers working in isolation. Today, however, the best computer games are designed by a coalition of a programmer or a group of programmers, a graphic artist and a musician. Of course, a single programmer can design games solo but to be successful the game should still look like it had professional graphics and music input. That's becoming increasingly important as computer games in general become more sophisticated. Commodore computers make gamewriting pretty easy even for beginners because of their built-in graphics, music and sound effects capabilities, so owning a Commodore computer gives you a good headstart if you're into gamewriting or want to be.

Designing Your First Keyboard Game

We're going to write a very simple keyboard game, using the principles we've learned in this article . . . plus what was covered in previous articles on random numbers in games (see the last VIC Magician). Don't worry if you didn't catch the earlier articles. We'll explain the various parts of the program in detail. Okay . . . ready? Let's write a game!

We'll start with an outline of the basic elements of the game. Later we may introduce some optional features (sound effects, scoring, etc.) but those don't have to be included until after we get the nucleus program working. This lets us work with a shorter program to begin with, which makes it easier to program and debug.

1. Green balloons appear one at a time, at RANDOM in the top row of the screen (locations 7680 to 7701). The ball symbol (POKE value 81) will be used.
2. The balloons fall straight down one at a time to the bottom of the screen, at which time a new balloon appears at random on the top row, and falls.
3. An up-arrow symbol (POKE value 30) can be moved back and forth on the bottom row of the screen (locations 8164 to 8185).
4. The player controls the up-arrow symbol by pressing the J key to go left and the L key to go right.
5. The object of the game is to move the arrow under the "balloon" to puncture it before it falls.

Arrows and Balloons

Let's call our game, "Arrows and Balloons" and start off by writing a very lean "skeleton" program, which we can expand later. Here it is . . . but before you type it in, this time take a quick look at the program commands. Note that we're going to try to use some mnemonic variables that describe what they represent. For example, BP means Ball Position, AP means Arrow Position.

```
10 PRINT"□"
20 AP=8175
30 IFPEEK(197)=20THENAP=AP-1
40 IFPEEK(197)=21THENAP=AP+1
50 IFAP>8185THENAP=8185
55 IFAP<8164THENAP=8164
60 IFBP=0THENBP=INT(22*RND(1))+7680
70 POKEBP,81:POKEBP+30720,5:POKEBP-22,32
80 POKEAP,30:POKEAP+30720,0:POKEAP-1,32:
   POKEAP+1,32
90 BP=BP+22:IFBP>8206THENBP=0
100 GOTO30
```

Enter this program, type the word RUN and press RETURN. Now you can use the "J" and "L" keys to control the movement of the arrow. The program is working now but of course it doesn't have any "bells and whistles" like sound effects or "explosions" when the balloon is zapped. So you're asking yourself, why does it look so confusing? Let's see what all these letters and numbers mean . . . they're not really as complicated as they look. Here's the explanation:

Line 10: Clears the screen.

Line 20: AP sets the beginning arrow position at 8175 in the middle of the bottom row.

Line 30: This is our familiar keyboard check, which looks to see if the "J" key is being pressed. If the J key is pressed, AP is redefined as AP-1 . . . notice that we haven't POKEd any symbol into this location yet . . . this happens farther down in line 80.

Line 40: This is the same as line 30, except here we use the "L" key to redefine AP as AP+1 which moves the arrow one position to the right . . . that is, when the program cycles through to line 80 and actually POKEs the symbol.

Line 50: This line sets the left and right limits of the arrow position . . . it says if you try to move the arrow to a screen location which is more than 8185 (far left position on the bottom row), AP (arrow position) will stay equal to 8185. The second part of the line says IF AP is less than 8164 THEN the arrow position stays equal to 8164 and won't go any lower. This line is important because if you don't set these limits the arrow can be moved off the screen.

Line 60: Look at the second part of this line first. The part that reads "BP=INT(22*RND(1)) + 7680" sets the ball position at a random screen location in the top row of the screen. We know it's the top row because the limits are 7680 to 7680 + 21 (7701) which covers the screen locations in the top row. This means the ball will appear at ONE random position in the top row. Now look at the first part of the line that reads "IF BP=0 THEN". To understand this, you have to know that random variables (in this case, the value of the variable BP are reset to zero when the program cycles through to the end . . . so by putting this in the line, we're telling the computer that whenever the program cycles all the way through, BP is reset to zero and when this happens, a new random number will be selected by the computer, which is what we want. Note that line 60 doesn't POKE anything on the screen . . . it merely selects a random position in the top row that we can POKE a symbol into, as we do in the next line . . .

Line 70: Here's where we POKE the ball (POKE number 81) into the random position BP which we set up in line 60. We also POKE the matching color location with the number 5 which colors the ball green. Then we erase the previous ball by POKEing a space (POKE number 32) into the location directly above the current position of the ball. This is important when we move the ball down the screen, otherwise we would get a vertical string of balls on the screen.

Line 80: Here's where we POKE the arrow symbol (POKE number 30) into the screen location represented by AP (which was determined first in line 20 . . . AP=8175. If the J or L key is pressed down, AP is defined as one position to the left or right in lines 30 and 40). AP−1 and AP+1 are both POKEd with an empty space (32) which erases the previous arrow when you move the arrow to a new position.

Line 90: TP=TP+22 increases the ball one space down the screen. The last part of this line sets the limit for the ball . . . in other words, when the ball drops beyond this limit, TP is reset to zero. This relates to line 60, where we told the computer "if TP is zero, pick another starting point for the balloons to appear."

Summary

So now you've written your first game! It involved a bit of POKEing and PEEKing but the result is interesting and may lead you to experiment by adding some extra features. In our next installment we'll examine some ways to add sound effects, and to put a little more ANIMATION into the game we just created. **C**

WAITE TRAINING FOR YOUR COMPUTER.

SAMS BOOKS™

Mitch Waite and Sams bring you the Sams/Waite Primer series, a better way to learn computing!

Learn the details of CP/M®—how it works and how you can use it—with our best-selling CP/M PRIMER, No. 21791, at $14.95.

BASIC PROGRAMMING PRIMER (2nd Edition) now helps make you an expert BASIC programmer no matter which micro you own! No. 22014, $17.95.

Do business graphics, animation, and more with help from COMPUTER GRAPHICS PRIMER, another Sams/Waite best-seller. No. 21650, $14.95.

PASCAL PRIMER gives you a pain-free introduction to PASCAL (No. 21783, $16.95) while MICROCOMPUTER PRIMER, 2nd Edition, talks technology so you can understand it (No. 21653, $14.50).

NOW THERE'S A NEW SAMS/WAITE PRIMER! UNIX® PRIMER PLUS introduces you to the powerful UNIX operating system with the only clearly written, friendly guide on the market. No. 22028, $19.95 (tentative).

SPECIAL WAITE NEWS FOR CP/M USERS!

CP/M BIBLE: THE AUTHORITATIVE REFERENCE GUIDE TO CP/M gives you fast, handbook-style access to all CP/M conventions, keywords, commands, utilities, and more. Covers CP/M-86, MP/M-80 and 86, and CP/Net, too. No. 22015, $18.95 (tentative).

SOUL OF CP/M: USING AND MODIFYING CP/M's INTERNAL FEATURES teaches you how to modify BIOS, use CP/M system calls and more. No. 22030, $14.95 (tentative).

Get all the muscle out of your computer with Sams "Waite Training" books. To buy Sams books, visit your nearest dealer, or call 800-428-3696 or 317-298-5566 today. Reference AD328.

SAMS BOOKS AND SOFTWARE

HOWARD W. SAMS & CO., INC.
4300 West 62nd Street • P.O. Box 7092 • Indianapolis, IN 46206

Offer good in USA only and expires 8/31/83. Prices subject to change without notice. In Canada, contact Lenbrook Industries, Ltd., Scarborough, Ontario.
CP/M is registered trademark of Digital Research, Inc. UNIX is a trademark of Bell Laboratories.

Programmer's Notebook:
More Fun With Prime Numbers

by Neil Harris

This is one article I didn't think I'd write. After all, with six different programs on prime numbers in my previous article (in the December-January issue), surely all the possibilities were covered.

Of course not. There is an old rule in programming: a program is never finished. Some astute readers wrote to me explaining some new techniques and some mistakes in my reasoning, so it's back to the drawing board.

First of all, I explained the reasoning that proves there is an infinite number of prime numbers. Assuming that there is a finite number of primes, multiply all the primes together and add one to the result. The result is indivisible by any of the primes, therefore there is no limit to the quantity of prime numbers.

I took this one step further. As far as I tested, when the prime numbers are multiplied together with one added to the result, the result was prime. 2*3+1=7, 2*3*5+1=31, etc. I then wrote some programs that resulted in ever-larger results, and invited comments on whether they were, in fact, prime.

Wrong again! As Mark Zimmermann and Bruce Adams were quick to point out, the method fails at the fifth number, 2*3*5*7*11*13+1=30,031=59*509. It also fails at the next number, 510,511=10*97*277. Most of the large numbers printed out by Programs 4, 5, and 6 of the article have to be suspect. I'm a programmer, not a mathematician.

Now for the part that's less excusable. Bruce Adams went on to show several mistakes in the programs as written. First off, line 20 in Program 2 should read, FOR L=4 TO SQR(N) STEP 2. The text of the article discusses the change although the program as shown doesn't incorporate it. The program should read:

Program 1

```
 0 REM SLOW PRIMER, VERSION
   2.1, BY—  NEIL HARRIS
10 N =3
20 FOR L=3 TO SQR(N) STEP 2
30 Q=N/L: IF Q=INT(Q) THEN 60
40 NEXT L
50 PRINT N,
60 N=N+2: GOTO 20
```

Fortunately for my ego, Bruce then claimed that there was another problem which, when I checked it, proved not to exist. He claimed that the SQR function in line 20 is calculated each cycle through the loop. Not so! MicroSoft BASIC only calculates the loop parameters when the loop is first executed. To test this, I used the real-time clock in the computer and added:

Program 2 . . . add these lines to Program 1

```
 0 REM SLOW PRIMER TIMER,
   2.1, BY→ NEIL HARRIS
 5 PRINT CHR$(147);: TI$="000000"
60 N=N+2: IF N  200 THEN 20
70 PRINT TI
```

Program 3 . . . change this line from Program 2

```
20 S=SQR(N): FOR L=3 TO S
   STEP 2
```

Program 2 takes 563 jiffies (60ths of a second) while Program 3 takes 573, on my Commodore 64. The extra 10 jiffies is the time it takes to set up the variable S. These times are without the spaces, which does speed things up.

Just when I was starting to feel good again, Bruce really went to town on me, this time for a quirk in the math routines in BASIC, which cause problems I really should have anticipated. You see, the math routines lose exact precision when dealing with the last few decimal places. This is caused by rounding errors when converting from the internal number format, which is binary, to the external format in decimal.

For example, let's take the square root of 26569, which we all know to be 163. When you type PRINT SQR(26569) in your computer, it responds with 162.999976. The error is only .000024, or a small fraction of a percent, but it is enough to confuse my program. The loop in line 20 will never get up to the correct number in this case. The program would evaluate 26569 as prime, incorrectly. A solution for this would be to add 1 to the SQR in line 20, as follows:

```
CORRECTION 1 TO PROGRAM 2:
20 FOR L=3 TO SQR(N) +1 STEP 2
```

Now comes the sticky part. Line 30 compares N/L to INT(N/L). A rounding problem in the binary-to-decimal conversion routine can be a killer here. Adams here suggests checking in line 30 for a remainder less than .1, instead of checking for a whole number after division.

This does work, as shown below.

Program 4

```
 0 REM SLOW PRIMER, V 2.2,
   BY → NEIL HARRIS & BRUCE
   ADAMS
10 N=3
20 FOR L=3 TO SQR(N) +1
   STEP 2
30 IF N-L*INT(N/L)<.1 THEN 60
40 NEXT L
50 PRINT N,
60 N=N+2: GOTO 20
```

Adams also suggests that the method shown in Program 3 of the previous article is prohibitively wasteful of memory, since the largest number an integer array can hold is 65535. In fact, the largest number that can be held is 32767 (because you can also go to −32768), which is indeed a small prime, and that a floating point array that can hold larger numbers is even more wasteful of memory. My answer is simply that it's true, but nobody's come up with a more efficient way.

My next letter came from Edmund Ricchezza, a stalwart member of the PET User Group of the Philadelphia Area Computer Society (PACS). He added two pieces of information. First, he supplied me with some suffixes of numbers even larger than before for Program 6 (which spells out each number by name, i.e., thirty-thousand thirty-one, etc.). After OCTillion comes NONillion, DECillion, UNDECillion, DUODECillion, TREDICillion, QUATTUORDECillion, QUINDECillion, SEXADECillion, SEPTENDECillion, OCTODECillion, NOVENDECillion, and VIGINTillion. Whew! The letters that are capitalized can be added as DATA after line 230, and the loop in line 7 should be expanded to FOR L=1 TO 20.

Next, Ed supplied me with a program even faster than my so-called Fast Primer program.

Program 5

```
 0 REM FASTER PRIMER, BY
   EDMUND RICCHEZZA
 5 DIM A(1000), B(200)
10 S=SQR(1000)
20 TI$="000000"
30 FOR B=2 TO 1000
40 IF A(B)  0 THEN 110
50 C=C+1
60 B(C)=B
70 IF B  S THEN 110
80 FOR X=B TO 1000 STEP B
90 A(X)= -1
100 NEXT
110 NEXT
120 PRINT
130 FOR X=1 TO C
140 PRINT B(X),
150 NEXT
160 PRINT
170 PRINT TI$
```

Finally, I got an interesting letter from Robert Frens, who compared the search for ever-larger primes with goldfish-swallowing or flagpole-sitting. He also provided an entirely different approach to discovering primes. He used the old "sieve" approach, which has the advantage of never dividing, but the disadvantage of being even more wasteful of memory than the other approaches. This method places all numbers into an array, then starts with the first prime and eliminates numbers at that interval. In other words, it starts at 2 and eliminates every second number, then starts at 3 and eliminates every third number, and so on.

Program 6

```
 0 REM PRIME SIEVE, BY ROBERT
   FRENS
10 N=32: M=N*N: P=1: DIM P%(M)
20 FOR J=2 TO M: P%(J)=J: NEXT
30 P=P+1: IF P%(P)=0 THEN 30
40 FOR J=P+P TO M STEP P
50 IF P  N THEN 70
60 P%(J)=0: NEXT: GOTO 30
70 FOR J=2 TO M
80 IF P%(J)  0 THEN PRINT P%(J),
90 NEXT
```

This program works almost five times faster than my program did. To solve for larger primes, simply change the value of N in line 10 to a larger number. Robert Frens, my hat is off to you.

One last comment, for those of you with VIC 20's who want to fit these programs into your more limited memories. Just decrease the number in the DIM statements in line 10 of the Fast Primer program, the Prime Multiplier programs, or Ed Ricchezza's programs.

Next issue, on to a new topic (unless you ever-vigilant readers come up with more goodies for me)! **C**

Win A FREE Commodore 64™ Computer

Can You Beat PARATROOPER?

10 Prize Winners

A New Commodore 64 will be awarded to the person with the highest score. The winning entry must contain a photograph of the highest score of the game.

PARATROOPER *By Nic Dudzik*

REALISTIC SMOOTH ACTION AND TRUE HI-RES GRAPHICS

JOYSTICK OR KEYBOARD

1st Prize
CBM 64

2nd Prize
Disk Drive

3rd Prize
$200 worth of Software

4th - 10th Prize
$100 worth of Software

Runs on the Standard VIC 20™

PARATROOPER a High Resolution game that doesn't let you make any mistakes. You are in command. Helicopters fill the sky, (and we mean fill the sky!), dropping paratroopers. Your mission is to keep 3 paratroopers from hitting the ground on either side of your gun. But that's just the beginning. You score by hitting the helicopters or the paratroopers, but if you miss a shot it subtracts from your score. Therefore, you must make every shot count to make a high score! IT HAS FOUR FAST ACTION LEVELS TO CHALLENGE THE BEST PLAYER. The High Resolution graphics helicoptors are fantastic. They look exactly like helicopters! The paratroopers are super realistic. Their chutes open and then they drift down to earth. If this weren't enough the sounds are fantastic. There are helicopter blades whirring and you can hear the howitzer pumping shells. This game really show off the sound and graphic capabilities of you VIC. PARATROOPER IS OUR #1 SELLING ARCADE GAME, you've got to see this game to believe it. **$19.95**

Order your copy today or see your dealer. Will you get the top score?

RULES

1. All entries must be mailed, as postmarks are required to determine the earliest winning entry. In the event of a tie the postmark will determine the winner.
2. Deadline for entries is July 31, 1983.
3. Proof of purchase must be provided. Return your entry with package front and proof of purchase slip and photo.
4. **ENTRIES MUST BE MAILED TO:** PARATROOPER CONTEST, P.O. Box 388, Lake Havasu City, AZ 86403
5. Game contest void where prohibited.
6. Contest begins April 2, 1983. All entries must be postmarked by July 31, 1983. Contest ends July 31, 1983.
7. Winners will be notified by mail. Public notice of winners will be printed in this and other computer magazines.
8. Only one entry per person please — all duplicates will be discarded.
9. There will be 10 winners for a 1st, 2nd & 3rd prize. Prizes for 4th - 10th will be for software.
10. Contest is limited to U.S.A., Properties and Canada only.

ComputerMat . P.O. Box 1664X . Lake Havasu City, Arizona 86403

The Rule of 78's

by John B. Roberts

This is a little simple program that computes the payoff balance on an add-on type loan. (This is called the Rule of 78's payoff balance—a mysterious term.)

Of course the variable "AN" is the old "A angle N," a geometric progression from my old algebra days, which is used to calculate the payment from interest, number of periods, and is simply divided into "P" (the principle) to get the payment "R."

"PO" is the payoff balance, "F" is the finance charge, "LP" is the last payment (in case it is a few cents different from a normal payment—the original program did not include this). "IN" is the interest that was charged as part of the payoff balance, and "RE" is the rebate of the interest you get for paying off the loan ahead of time.

The subroutine 4 through 8 is from the *VIC 20 Programmer's Reference Guide*, and as far as I can tell, this is the only way of generating a pseudo "PRINT USING" statement.

This program might be of interest to some of the smaller finance companies out there. I called several here in Tulsa, a couple of years back, and no one knew the equations to use for this. They simply look the numbers up in a book, is what I was told. Someone might like to know how these numbers are actually generated—if they had a computer, they could get these numbers directly and move up to the computerized modern world! C

```
1 PRINT"■■■RULE OF 78'S■■■■■PROGRAM■■"
2 INPUT"INT./YEAR";I:I=I/1200:PRINT
3 GOTO10
4 Z=Z*100:Z=Z+.5:Z=INT(Z):Z=Z/100
5 Z$=STR$(Z):Q=LEN(Z$)
6 FORL=QTO1STEP-1
7 IFMID$(Z$,L,1)<>"."THENNEXTL:Z$=Z$+".00"
  :GOTO9
8 IFL=Q-1THENZ$=Z$+"0"
9 RETURN
10 INPUT"NUMBER OF MONTHS";N:PRINT
11 PRINT"AMOUNT BOORROWED■"
12 INPUT"$";P
13 Z=P
14 GOSUB4
15 P=Z
16 P$=Z$
17 PRINT"■■AMOUNT BORROWED■"
18 PRINT"$";P$;"■"
19 AN=(1-(1+I)↑-N)/I
20 R=P/AN
21 Z=R
22 GOSUB4
23 R=Z
24 R$=Z$
25 PRINT"PAYMENT $";R$;"■"
26 F=R*N-P
27 Z=F
28 GOSUB4
29 F=Z
30 F$=Z$
31 LP=P+F-(N-1)*R
32 Z=LP
33 GOSUB4
34 LP=Z
35 LP$=Z$
36 PRINT"FINANCE CHARGE■"
37 PRINT"$";F$;"■"
38 PRINT"SELECT TRANSACTION:■"
39 PRINT"■■■PAYOFF BALANCE■"
40 PRINT"■■■COMPLETE SCHEDULE■"
41 GETA$:IFA$=""THEN41
42 IFA$="A"THEN45
43 IFA$="B"THEN76
44 GOTO41
45 REM PAYOFF BALANCE
46 PRINT"■■PAYOFF BALANCE■"
47 PRINT"NUMBER OF PAYMENTS WAS■"
48 PRINTN;"■"
```

```
49 PRINT"PAYOFF AT PAYMENT■"
50 INPUT"NUMBER";A
51 GOSUB53
52 END
53 REM CALC. PAYOFF
54 REM CALC. INT.
55 REM CALC. REBATE
56 PO=R*N-(A-1)*R-(N-A)*((N-A)+1)*(F/
   (N*(N+1)))
57 Z=PO
58 GOSUB4
59 PO=Z
60 PO$=Z$
61 IFA=NTHENPO$=LP$
62 IN=2*F/(N*(N+1))*((N-A)+1)
63 Z=IN
64 GOSUB4
65 IN=Z
66 IN$=Z$
67 RE=(N-A)*((N-A+1)*F) /(N*(N+1))
68 Z=RE
69 GOSUB4
70 RE=Z
71 RE$=Z$
72 PRINT"■PAYOFF = $ ";PO$;"■"
73 PRINT"INTEREST = $ ";IN$;"■"
74 PRINT"REBATE = $ ";RE$;"■"
75 RETURN
76 REM COMPLLETE SCHEDULE
77 PRINT"■":J=0
78 J=J+1:A=A+1
79 PRINT"■PAYMENT NUMBER";A
80 GOSUB53
81 IFA=NTHENEND
82 IFJ<>2THEN78
83 PRINT"ENTER C TO ■CONTINUE■"
84 GETA$:IFA$=""THEN84
85 IFA$="C"THEN77
86 GOTO84
```

Saga Of A Printer Terminal And The RS-232C Interface

By Bill Trice

I was fortunate enough to acquire a standard RS-232 printer terminal at a very reasonable price. So, naturally, I figured all I had to do was purchase the VIC-1011A RS-232C interface for my VIC 20 and hook it up to the printer. I found out this wasn't as simple a procedure as one might expect.

I purchased a standard RS-232 cable and inter-connected the printer and the VIC. When I couldn't transfer any data at all, I dug into both the printer manual and the RS-232C interface manual. What I found was that both units think they are data terminals. As a result, transmitted data coming out of the interface goes down the same line as transmitted data coming out of the printer. The same is true of received data.

I also noticed that the "request to send" and the "clear to send" for both units were butted against each other. So I figured, why not swap them in one of the connectors? According to the connector pin-out diagram on page 13 of the interface manual, transmitted data is on pin 2 of the 25-pin EIA connector and received data is on pin 3. So I swapped them in one of the connectors on my RS-232 cable. Likewise, I swapped pin 4 (request to send) and pin 5 (clear to send). I thought that pin 6 (data set ready) and pin 20 (data terminal ready) might need to be swapped also, but as it turned out, it was not necessary. When I plugged the modified cable in, BINGO! I heard the printer start printing on request from the VIC.

I still had a small problem, however. When I sent a program listing to the printer, I got no linefeeds at the end of each line. Page 4 of the interface manual clued me into the answer to this dilemma. If the logical file number used in the OPEN statement is less than 128, then no linefeed is generated after a carriage return. When I changed the logical file number to 128, I found myself in business.

My printer will go 1200 baud, but don't plan on listings at that speed. CBM BASIC doesn't have any carriage return delays built into it, so the printer can't react fast enough to get the carriage back for the next line. 1200 baud is feasible when printing from within a program, however, by stalling a delay subroutine in the program and doing a GOSUB to it after every PRINT#128 statement. You will find that this eats up a lot of RAM, so you may prefer to stay at 300 baud or slower.

Also, be sure to OPEN the printer file before using any variable or DIM statements in your programs. I found this out the hard way after bombing a couple of programs. You see, when you OPEN an RS-232 channel, an automatic CLR is performed, allocating 512 bytes at the top of memory. If 512 bytes of space are not available at the time of the OPEN statement, your program is destroyed. The best way to avoid this is to make your OPEN statement the first one in the program. The OPEN statement takes the following form:

```
OPEN128,2,3,CHR$(6+32)+CHR$(32+128)
```

This OPENs the RS-232 channel (2). I still don't understand the 3 that follows the 2, but that's what was used in the sample BASIC program on page 11 of the interface manual. The CHR$(6+32) sets the baud rate at 300 and seven-bit word length. Set this to CHR$(8+32) for 1200 baud. The CHR$(32+128) sets up for no parity check. I finally simplified the OPEN statement to:

```
OPEN128,2,3,CHR$(38)+CHR$(160)
```

After you OPEN the printer file in your program, simply use PRINT#128 statements when you want to go to the printer. Don't forget to CLOSE 128 before the end of your program. To list your programs, type the OPEN statement directly, followed by CMD128 and LIST:

```
OPEN128,2,3,CHR$(38)+CHR$(160):CMD128:LIST:
PRINT#128:CLOSE 128
```

Before you CLOSE the printer file, be sure to do a PRINT#128.

I hope this helps out those of you who had problems using the RS-232C interface with a printer and those of you thinking about going this route. Happy printing. **C**

Formatting and Error Checking:
A Time-Saving Algorithm

By Anthony J. LaMartina

The demonstration program on page 88 shows how to format columnar numbers so the decimal points always align. There are several ways to accomplish this task. The initial program listing is a simple, easy to understand way to align the decimal points for column numbers rounded to two decimal places. Further enhancement of the initial program demonstrates error trapping.

Lines 10 and 30 input our data for mathematical manipulation. In this case, we're inputting A and B to be divided, with the result being C.

Lines 20 and 40 take the string input and convert it to a numeric variable equal to the value of the string (A=VAL(A$)). Our inputs are entered as string variables to facilitate the program expansion discussed later.

As you can see, A$ is entered at column 1, line 1. Input B$ is then positioned at column 21(spc20), line 1(" ").

Line 50 rounds the result, to two decimal places via the DEFFNA statement, and line 60 does the mathematical computation.

Now the fun begins!

Line 70 converts our rounded result, (C) to a string representation. Line 80 looks at the rightmost two characters of our C$ string and assigns them to string variable X$.

Line 90 then looks at the leftmost character of string X$ to see if it is a decimal point (remember in a string the decimal point is also considered a character), meaning our answer has only one decimal place because PET BASIC automatically truncates trailing zeros (1.55+55=3.1, not 3.10). If this condition is true then we add "0" to C$, thus giving our answer two decimal places.

Line 100 operates similarly to line 90. If our answer C is equal to the integer of C (no decimal places, 1.5+1.5=3 not 3.0 nor 3.00) then we add ".00" to C$, again giving us an answer carried out to two decimal places.

The key to this little routine is line 110. Since our answers (properly formatted through lines 70-100) need to be printed in a column with the decimal points aligned, we use SPC and LEN to our advantage here.

When we print our answer, we space over 60 columns minus the length of our answer. This will align the last character of our answer, and since all our answers have been formatted to two decimal places the decimal points will automatically align, too! This makes for ease of reading and is more esthetically pleasing to the eye also.

Line 120 brings us back to line 10 for multiple entries and viewing of the formatted output column.

Now, what about the program expansion I spoke of? This expansion involves error checking to eliminate the unwanted "? REDO FROM START" that generates and ruins your formatted display. Also we'll see how to avoid the "? DIVISION BY ZERO" error that may also occur.

The addition of the following lines after each input will branch away from the error message "?REDO FROM START" prior to its display. (This is why we input a string rather than a number.)

```
12 k=len(a$)
14 for j=ltok
16 if asc(mid$(a$,j,l))<45 or asc(mid$(a$,j,l))>57 then j=k:
   goto1000
18 next
```

Repeat these same lines as lines 32, 34, 36 and 38. Change the goto in line 38 to 1050.

Briefly, what these additional lines accomplish is this. We input a string of length K. If our string (or any single character of our string) has an ASCII value of less than 45 or greater than 57 then branch out to an error alert routine. ASCII values in this range allow for numeric entries, fractional numeric entries, and negative numeric entries. All non-numeric entries will cause an error branch. Branch to what? Well, let's give our operators an audible alert to an erroneous entry and a chance to try, try again! We'll use the following lines as an "error branch to" and an audible alert subroutine. First, we'll alert the operator to the error via the built in chime present in PET/CBM 40/80 columns. (Those of you with the 2001 series will need an amplifier hooked up to the CB2 output.)

In the preceding line 16 we used a goto 1000 for an error branch. Line 1000 directs us to gosub5000, so let's take a look at 5000+. Line 5000 sets up a 3X loop. Line 5010 pokes location 59467 with a value of 16. This sets the shift register oscillator to a free-running condition with a squarewave output. All this means is it turns the music on! Pokes 59464 and 59466 set the tone and timbre of our musical note output.

Line 5020 gives a time delay, while line 5030 changes the tone and timbre and line 5040 gives us another time delay.

Line 5050 sends us back to line 5000 to do it all again. After 3 repetitions, line 5050 pokes all three poked locations to zero, turning off the music port and thus we return to line 1000 going to line 1010.

The values at locations 59464 and 59466 can be any value from 2 through 254 inclusive. By varying these values and the amount of time delay, you can program your own alert or any musical ditty! (See CB2 Music Programming in the third edition of the *PET Personal Computer Guide* by Osborne, Strasma and Strasma.

Line 5060 could be removed and placed at line 1005. This line merely positions the cursor for what is about to happen in line 1010.

Using the CHR$ (22) command, erase line from cursor to end, we erase our erroneous entry. PET BASIC forces an automatic carriage return after any print command, therefore it is necessary to re-position our cursor up a line or two. We then goto line 20 or 30 depending, and re-input at the same

space and column position we started from!

You can further error-check by trapping errors such as too long a string input:

55 if len(a$)>7 then 1000
115 if len(b$)>7 then 1050
or trapping a divide by zero error
135 if b$="0" then c$=".00":goto200
You will need an error erasure line for each error-checking routine goto after input, since the goto changes in each case based on its associated input. The error erasure routine (lines 1000-1020 and 1050-1070) could not be made into a subroutine, because the return would bring you back to the subsequent line from which you came. We need to re-input; thus lines 10 and 30 are our target lines and only a goto, not a gosub will work here. Also you will notice on the complete demonstration program that a$ and b$ input have three shifted spaces and three left cursors for further input protection. Thus, if just "return" is pressed without an entry, the program will branch to the error-alert routine.

You could dress up this program by putting in column headings and tabular borders, as well as change the formula to fit your needs.

So, in summary, we've learned to format and align decimal numeric data, how to error check for correct numeric entry, and how to erase erroneous entries and give the operator another chance. All with a few short routines and a couple of tricks from our bag of programming magic!

Formatting Error Checking

The program listings provided here are for a CBM computer. PET, VIC and 64 users should substitute a [ ] (CLR/HOME) character for the [ ] and [ ] (cursor up) for [ ]. The audible alarm routine (lines 1000-5070) in the first program will work on the CBM only. Anybody want to modify the POKEs to make it work on the VIC and 64?

```
5 REM FORMATTING DEMO W/ERROR CHECKING, AUDIBLE ALERT
7 REM AND INVALID ENTRY ERASURE
10 PRINT"[ ]"
20 INPUT"    [ ]";A$
30 K=LEN(A$)
40 FORJ=1TOK
50 IFASC(MID$(A$,J,1))<45ORASC(MID$(A$,J,1))>57THENJ=K:
   GOTO1000
55 IFLEN(A$)>7THEN1000
60 NEXT
70 A=VAL(A$)
80 PRINTSPC(20)"[ ]";:INPUT"    [ ]";B$
90 K=LEN(B$)
100 FORJ=1TOK
110 IFASC(MID$(B$,J,1))<45ORASC(MID$(B$,J,1))>57THENJ=K:
    GOTO1050
115 IFLEN(B$)>7THEN1050
120 NEXT
130 B=VAL(B$)
135 IFB$="0"THENC$=".00":GOTO200
140 DEFFNA(C)=INT(C*100+.5)/100
150 C=A/B
160 C$=STR$(FNA(C))
170 X$=RIGHT$(C$,2)
180 IFLEFT$(X$,1)="."THENC$=C$+"0"
190 IFC=INT(C)THENC$=C$+".00"
200 PRINTSPC(60-LEN(C$))"[ ]"C$
210 GOTO20
999 REM INVALID ENTRY ERASURE
```

```
1000 GOSUB5000:REM AUDIBLE ALERT
1010 PRINTCHR$(22)
1020 PRINT"■■":GOTO20
1050 GOSUB5000:REM AUDIBLE ALERT
1060 PRINTTAB(10);CHR$(22)
1070 PRINT"■":GOTO70
4999 REM AUDIBLE ALERT SUBROUTINE
5000 FORI=1TO3
5010 POKE59467,16:POKE59464,50:POKE59466,100
5020 FORT=1TO100:NEXTT
5030 POKE59464,75:POKE59466,175
5040 FORT=1TO100:NEXTT
5050 NEXTI:POKE59467,0:POKE59464,0:POKE59466,0
5060 PRINT"■■"
5070 RETURN
```

```
5 PRINT"■"
10 INPUT" ";A$
20 A=VAL(A$)
30 PRINTSPC(20)"■";:INPUT" ";B$
40 B=VAL(B$)
50 DEFFNA(C)=INT(C*100+.5)/100
60 C=A/B
70 C$=STR$(FNA(C))
80 X$=RIGHT$(C$,2)
90 IFLEFT$(X$,1)="."THENC$=C$+"0"
100 IFC=INT(C)THENC$=C$+".00"
110 PRINTSPC(60-LEN(C$))"■"C$
120 GOTO10
130 END
```

The first in a three-part series by one of the authors of the VIC 20 User Guide.

Getting the Most Out Of (And Into) Your Disk System

-Part One-

by John Heilborn

Although the 1540 and 1541 disk drives are a real help in using the VIC 20 and Commodore 64 computers for advanced applications, occasionally the 142 files available on each diskette just aren't enough. This series of three articles will show you how to more than quadruple the number of random files you can put onto a single diskette, help you to understand how the disk system works at the same time and give you a mailing list program.

The Brain Behind the Brain

The "brains" behind the VIC 20 and Commodore 64 computers is a single device called a microprocessor. It is this device that performs all the logical functions of the computer and controls all the peripheral devices that can be connected to it (such as the disk drive, the printer and the modem). But the computer is not the only part of the Commodore system that uses a microprocessor. The disk drive also has a microprocessor in it. In fact, the 1540 and 1541 disk drives use the same kind of microprocessor that the VIC 20 does. By using the microprocessor in the disk drive for storing and retrieving data from the disk, the computer has more time to perform other tasks.

Operating Systems

To get a better idea of how this works, let's look at the way a computer runs. Initially, the computer can't understand anything at all. To make the computer able to accept instructions, the people who design the computer give it a program that tells it how to get commands from you and what to do with them.

The program that tells the computer how to operate is called (logically enough) the *operating system* (on the VIC 20 and Commodore 64, the operating system is built right in). In addition to the many duties it performs, the operating system tells the computer to use BASIC (the programming language) to interpret all your commands into a language it can understand—machine code. For a more detailed explanation of how this works, get a copy of "An Introduction to Microcomputers—Volume 0" by Adam Osborne. For the moment however, just accept that it does work.

Of course, if the microprocessor in the VIC 20 (or 64) has an operating system to tell it how to operate and a language that allows you to communicate with it, it would seem logical that the disk drive would have these too. Well, it does, and as you might expect, the program that tells the disk drive what to do is called the *disk operating system* (DOS). The disk drive has its own programming language just as the computer does and you can write programs for it, and it will execute them just like the computer does. However, since the disk drive is not usually called upon to perform tasks which are as complex as those performed by the computer, its language is much simpler. The following table summarizes some of the commands that the disk drive can accept. By using these instructions you can read or write to any part of a diskette.

The 1540/1541 Instruction Set

Command	Abbreviation	Description
Block-Read	B-R	Reads a section of the diskette and stores it in disk memory.
	U1	Performs the B-R operation but first moves the block pointer to 255 in order to read the entire block.
Block-Allocate	B-A	Marks a section of the diskette as occupied.

Block-Write	B-W	Writes data which has been stored in disk memory onto a section of the disk.
Block-Free	B-F	De-allocates a section of the diskette so data can be written there.

Tracks and Sectors: The Organization of a Disk

If you looked at a diskette outside of its square, black plastic jacket (don't take it out, you would probably damage it if you did) you find that it is a very thin flexible disk made of plastic similar to the material that cassette tapes are made of. The data is stored in an even thinner layer of magnetic metal oxide on the surface of the plastic and is written in a circular pattern that resembles the way music is put onto a phonograph record. There is a difference however. The grooves on a record are actually a single, spiral groove. On a diskette, the grooves (tracks) are actually separate rings. So, if the disk head stayed on the same track while the diskette rotated, it would end up back where it started. (On the other hand, a record needle would end up in the center of the record.) Each track is divided up into several sections called sectors. On the 1540 and 1541 disk drives, different tracks have different numbers of sectors ranging from twenty-one sectors on the outer tracks to only seventeen sectors on the inner tracks.

Blocks and Sectors

Before we start in on the actual disk commands, there is one more term you will need to understand. Commodore uses the term 'block' to refer to the data in a single sector of a disk. This is equal to 256 bytes of data. So when you perform a block-read for example, you'll be reading the data from a single sector of the diskette.

Reading from the Disk

To read from the disk, you'll need to OPEN two separate files. One is used to send commands to the disk drive and the other is used to transmit data to and from the disk.

The first file must be channel 15 because that is the only command channel. So to OPEN the command channel you would enter:

```
OPEN 15,8,15
```

(the 8 is used to access the disk drive)

Then you must OPEN a direct access channel for data communication: (For more information on direct access channels, read the *VIC 20 User Guide* available from Osborne/McGraw Hill).

```
OPEN 2,8,4,"#"
```

Now that you've set up your data channels, all you have to do is read the data using the block-read instruction:

```
PRINT#15, "U1:"4;0;T;S
```

(Note that T refers to the track being read and S refers to the sector).

Finally, you'll need to read the data out of the data buffer (it was transferred into a buffer when it was read, not to the screen). So to put the data onto the screen, use the following routine:

```
10 GET#2, A$
20 PRINT A$;
30 IF ST= 0 THEN 10
40 CLOSE 2
50 CLOSE15
60 END
```

The status register (ST) is used in the routine above to signal the program that the file has been totally read. As long as it is zero, there is more to read; when it is not zero, the file has been read and the program ends, closing all the data access channels.

In the next issue, you'll learn how to write data to a disk file and how to manage your own random-access file system. The last article in this series will describe a complete mailing list system with the capability of managing up to 600 names and addresses on a single diskette. C

John Heilborn is an editor with Osborne/McGraw-Hill. He has recently co-authored the VIC 20 User Guide *with Ran Talbott. The price is $14.95 and it's available from:*

Osborne/McGraw-Hill
2600 Tenth Street
Berkeley, CA 94710

Enhancing Your REMarks on PET/CBM

by John Taylor
Math Department, Healdsburg High School, Healdsburg, California

We all know how difficult it sometimes is to find those remark statements in a listing that has been on the shelf for a while.

Following you will find a short six line program that will enhance all remarks that appear in a program when it is listed on a printer (see Listing #1).

The use of this routine is very easy:

1. Append this routine to the end of an existing program (Load enhance program—list it—load program to be enhanced—return cursor through enhance program on screen to add it to your program) or
   Load this program before you start to type a new program.
2. Whenever you put in a remark, put a "PI" right after the REM (see Listing #2).
3. When you are ready to enhance, type RUN 50000. A counter will appear in the upper left corner of the screen to let you know something is happening.
4. Delete the enhance routine, if you wish, and save program.

The printer listing will now be as it is in Listing #3. (Excuse the excessive remarks, but you get the idea!)

If the program is listed to the screen, it will look normal, but the "PI's" will no longer be there. See screen dump in Listing #4.

Line 49999:
Flag so enhance program does not over run itself.

Line 50000:
Starts search from bottom of basic (Same as the street address of our school).

Line 50010:
Prints an "anti-anxiety" message to the screen.

Line 50020:
Looks for the "PI's" in line 49999 to terminate the program.

Line 50030:
Looks for a REM followed by a PI.

This is the guts of this routine. This line replaces the PI with a 01 in memory. 01 is an unused character for CRT output, but it is the enhancement character for Commodore printers [CHR$(1)]. When the printer is listing, it accepts the 01 and enhances until a return is found.

Cute, eh???

Hope you can spread the word. I use it all the time, so do my students. **C**

Listing 1.

```
REM ENHANCE ROUTINE

49999 ππππ
50000 FORX=1024TO50000
50010 PRINT"■■"X
50020 IFPEEK(X)=255ANDPEEK(X+1)=255THENEND
50030 IFPEEK(X)=143ANDPEEK(X+1)=255THENPOKEX+1,1
50040 NEXT
READY.
```

Listing 2.

```
PROGRAM BEFORE ENHANCEMENT

30000 REMπ >> REM ENHANCE DEMO <<
30010 :
39997 :
39998 REMπ >> INPUT SUBROUTINE  <<
39999 :
40000 PRINTCL$"□":N$="":POKE167,0
40003 REMπ > CURSOR ON <
40005 N$="":POKE167,0
40007 REMπ > GET LOCK-UP <
40010 GETA$:IFA$=""THEN40010
40012 REMπ > DELETE W/NO INPUT CHECK <
40013 IFN$=""ANDA$=CHR$(20)THEN40010
40014 REMπ > SH=1,SHIELD OFF; =0 ON <
40015 IFSH=1THENB$=A$
40016 REMπ > DELETE LAST CHARACTER <
40017 IFA$=CHR$(20)THENPRINT" ■■ ■";:N$=LEFT$(N$,LEN(N$)-1):GOTO40010
40019 REMπ > DETECT RETURN <
40020 IFA$=CHR$(13)THEN40035
40025 REMπ > BUILD INPUT STRING <
40030 PRINTB$;:N$=N$+A$:GOTO40010
40032 REMπ > CHECK FOR NULL INPUT <
40035 IFN$=""THEN40010
40040 PRINT:RETURN
40050 :
40060 :
49990 REMπ > REM ENHANCE ROUTINE <
49999 ππππ
50000 FORX=1024TO50000
50010 PRINT"■■"X
50020 IFPEEK(X)=255ANDPEEK(X+1)=255THENEND
50030 IFPEEK(X)=143ANDPEEK(X+1)=255THENPOKEX+1,1
50040 NEXT
```

Listing 3.

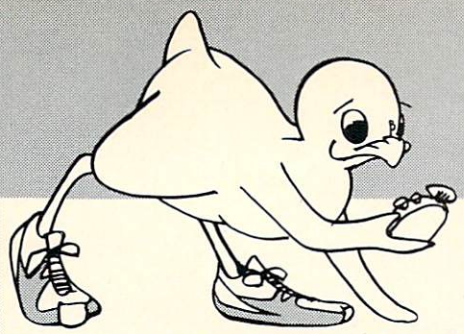
```
AFTER ENHANCEMENT

WITH ROUTINE DELETED

30000 REM  >>  REM ENHANCE DEMO  <<
30010 :
39997 :
39998 REM  >>  INPUT SUBROUTINE  <<
39999 :
40000 PRINTCL$"■":N$="":POKE167,0
40003 REM  >  CURSOR ON  <
40005 N$="":POKE167,0
40007 REM  >  GET LOCK-UP  <
40010 GETA$:IFA$=""THEN40010
40012 REM  >  DELETE W/NO INPUT CHECK  <
40013 IFN$=""ANDA$=CHR$(20)THEN40010
40014 REM  >  SH=1,SHIELD OFF; =0 ON  <
40015 IFSH=1THENB$=A$
40016 REM  >  DELETE LAST CHARACTER  <
40017 IFA$=CHR$(20)THENPRINT" ■■ ■";:N$=LEFT$(N$,LEN(N$)-1):GOTO40010
40019 REM  >  DETECT RETURN  <
40020 IFA$=CHR$(13)THEN40035
40025 REM  >  BUILD INPUT STRING  <
40030 PRINTB$;:N$=N$+A$:GOTO40010
40032 REM  >  CHECK FOR NULL INPUT  <
40035 IFN$=""THEN40010
40040 PRINT:RETURN
40050 :
READY.
```

Listing 4.

```
40000 PRINTCL$"":N$="":POKE167,0
40003 REM > CURSOR ON <
40005 N$="":POKE167,0
40007 REM > GET LOCK-UP <
40010 GETA$:IFA$=""THEN40010
40012 REM > DELETE W/NO INPUT CHECK <
40013 IFN$=""ANDA$=CHR$(20)THEN40010
40014 REM > SH=1,SHIELD OFF; =0 ON <
40015 IFSH=1THENB$=A$
40016 REM > DELETE LAST CHARACTER <
40017 IFA$=CHR$(20)THENPRINT"  ";:N$
=LEFT$(N$,LEN(N$)-1):GOTO40010
40019 REM > DETECT RETURN <
40020 IFA$=CHR$(13)THEN40035
40025 REM > BUILD INPUT STRING <
40030 PRINTB$;:N$=N$+A$:GOTO40010
40032 REM > CHECK FOR NULL INPUT <
40035 IFN$=""THEN40010
```



It's Time for TOTL SOFTWARE!

for the VIC 20™ and COMMODORE 64™
WORD PROCESSING AND MAILING LIST & LABEL
now available with **CHICKSPEED**

FAST PRINTING • LIGHTNING LOADS • SIMPLE COMMANDS

TOTL.TEXT 2.0 + CS VIC + 8K expansion	$25.00
TOTL.TEXT 2.5 + CS VIC + 16K expansion	$35.00
TOTL.TEXT 2.6 + CS Commodore 64	$40.00
TOTL.LABEL 2.1 + CS VIC + 16K expansion	$20.00
TOTL.LABEL 2.6 + CS Commodore 64	$20.00
TOTL TIME MANAGER 2.1 VIC + 8K expansion	$30.00
TOTL TIME MANAGER 2.6 Commodore 64	$35.00
time management, scheduling, reports	
RESEARCH ASSISTANT 2.0 VIC + 8K expansion	$30.00
RESEARCH ASSISTANT 2.0 Commodore 64	$35.00
key word cross-reference research tool	
TOTL.BUSINESS 3.0 VIC + 16K expansion	$85.00
TOTL.BUSINESS 3.6 Commodore 64	$95.00
business programs require disk and are shipped on disk	
One Megabyte Fuzzy Diskette	$25.00
computer novelty pillow	

All programs work with 40/80 column (VIC) and 80 column (64) adapters—compatible with tape or disk systems —shipped on cassette tape— available on disk $4.00 extra.

Quality You Can Afford

Available at your local dealer or by phone order

TOTL software inc.

1555 Third Ave., Walnut Creek, CA 94596

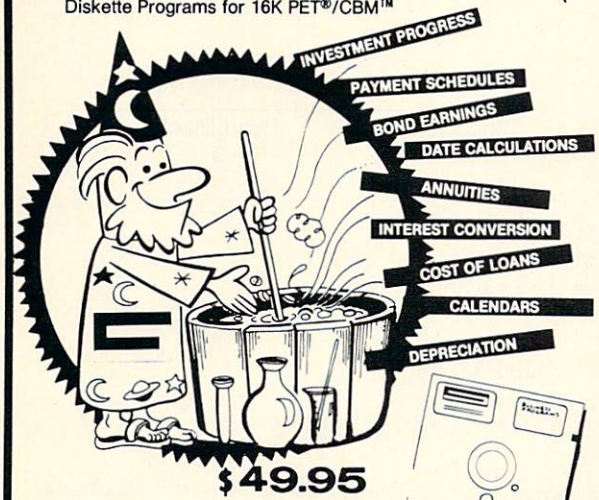
MasterCard VISA Call (415) 943-7877

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

Introducing The

FINANCIAL WIZARD

Diskette Programs for 16K PET®/CBM™



$49.95

Financial Programs

The Wizard helps you quickly calculate 12 major types of Financial Transactions with ease and accuracy.

Created for businesses and individuals, this DISK PACKAGE has 9 programs plus instructions. The Wizard delivers answers on the screen or printer.

CASCADE COMPUTERWARE • (206) 355-6121

Box 2354 • Everett, WA 98203 Dealer inquiries invited

MICROSPEC

Quit Playing Games . . .

Disk Based Software to Make Your Computer Get Down to Business

Disk Data Manager—Create and manage your own data base. Allows you to create, add, change, delete, search, sort, print, etc. Up to 1200 records on a single disk.
VIC 20. . . 59.95 CBM 64 . . . 89.95

Payroll System—Full featured, complete payroll system. Even prints checks.
VIC 20. . . 89.95 CBM 64 . . . 99.95

Mailing List—Up to 1200 records on a single disk. Presorts by Zip Code. Prints on stock up to four labels wide.
VIC 20. . . 44.95 CBM 64 . . . 54.95

Inventory Package—Maintains quantity on hand, cost, sales price, reorder point, etc. Generates suggested reorder, sales report, and sales analysis.
VIC 20. . . 89.95 CBM 64 . . . 99.95

General Ledger—Up to 75 accounts! Generates Balance Sheet, Income Statement, Update Report, etc.
VIC 20. . . 89.95 CBM 64 . . . 99.95

Checkbook Manager—Up to 25 expense categories. Tracks all outstanding checks until they are paid.
VIC 20. . . 49.95 CBM 64 . . . 49.95

CONTACT US FOR ALL YOUR DISK BASED SOFTWARE NEEDS

Call for specifics on Hardware Configurations. Send Self-Addressed Stamped Envelope for Catalogue of Games and other Applications

DEALER INQUIRIES WELCOME

VISA

P.O. Box 863085
Plano, Texas 75086
(214) 867-1333

MasterCard

VISA and MASTERCARD Accepted

Cassette Fix For Original ROM PETs

by Ronald E. Randolph

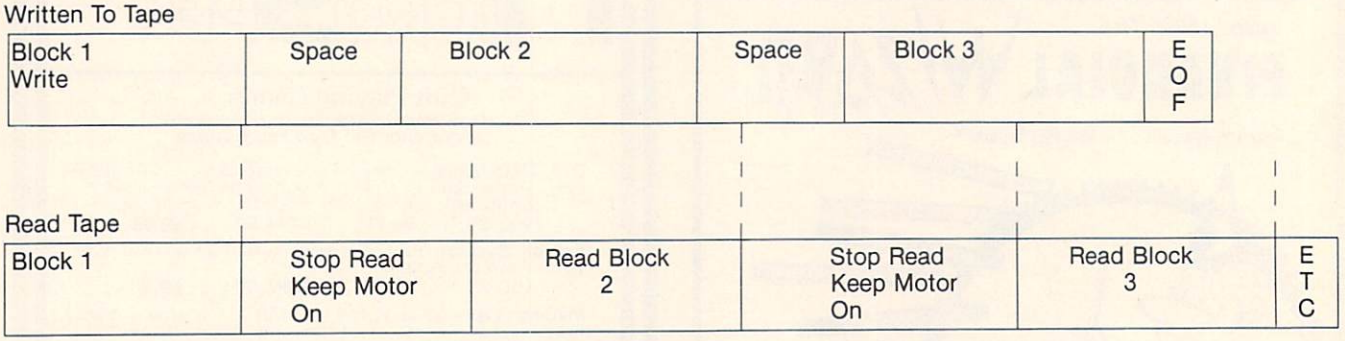
Owners of original ROM PETs suffer a bug in the way the system WRITES and READS data to cassette tape.

During a WRITE to tape, data is written as blocks of 190 characters in length. A space is then written, then the next data block is written. When all data has been written, an 'END OF FILE' (EOF) signal is then written to the tape.

During a READ from tape, the system reads the first 190-character block from the tape, keeps the cassette motor running but tells itself to cease its READ operation for a small amount of time. Then it begins to READ the next block of 190 characters. The problem occurs as the PET begins to READ data blocks subsequent to the first block. Apparently the insufficient time PET takes during its 'CEASE READING' periods will cause it to overlap the beginning of the next data block. Figure 1 gives a graphic illustration of this phenomenon. When this happens, data characters are lost and sometimes system crashes can occur.

Original ROM PET owners are probably familiar with this software routine, which increases the space between tape blocks during a WRITE-to-tape operation.

Figure 1



Shows tape writing and subsequent reading. Note the difference in spacing between that put on the tape at WRITE time vs. that at READ time. Note, too, that EOF marker could possibly occur at a time when the PET is in a stop-read phase. In this case the cassette would keep running and not stop.

```
1000 REM A$ CONTAINS LINE TO WRITE
1001 T6=TI:IFLEN(A$)<190THEN1006
1002 PRINT#1,LEFT$(A$,189);
1003 IFTI-T6>120THENGOSUB2000
1004 T6=TI:PRINT#1,RIGHT$
     (A$,LEN(A$)-189
1005 GOTO1005
1006 PRINT#1,A$
1007 IFTI-T6>120THENGOSUB2000
1008 RETURN
2000 REM TURN ON CASSETTE
     FOR .1 SEC EXTRA
2005 POKE59411,53:T6=TI
2010 IFTI-T6<6THEN2010
2015 POKE 59411,61:RETURN
```

This increase in space is sufficient to move the data blocks far enough apart so during the PET "cease reading while motor is still running" period, subsequent reads will catch the beginning of each data block.

I have always had a penchant for seeking hardware solutions to such problems rather than software routines. By corresponding with Mark Zimmerman, who had corresponded with Richard Beck of Ontario, I was able to make a little

modification to the resident PET cassette unit that insured me of reliably writing and reading data files to tape. The circuit effectively adds capacitance across the motor when the cassette is in the record mode only. This extra capacitance prevents the motor from shutting off as it normally would during a record or WRITE operation. It continues to run a bit more and therefore increases the space between tape blocks during a WRITE to tape. Figure 2 will give the circuit. Figure 3 will show the exact wiring to make the change to the cassette unit itself. You must remove the cassette from the PET and take off the back cover, exposing the unit's printed circuit board.

Figure 2

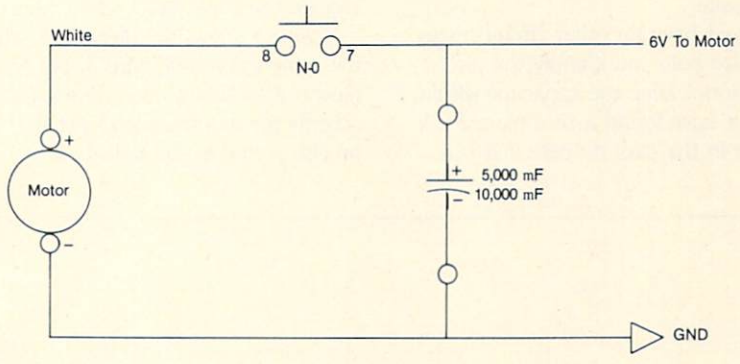
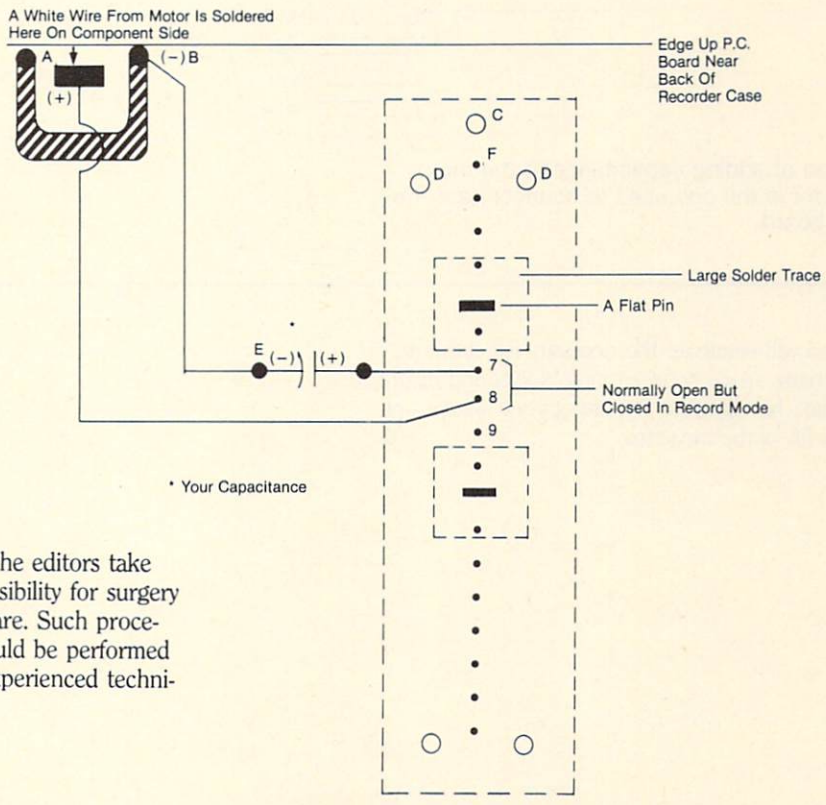


Figure 3



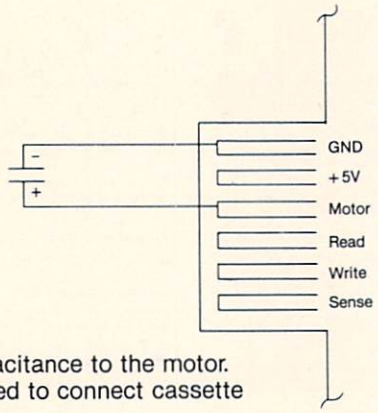
(NOTE: The editors take no responsibility for surgery on hardware. Such procedures should be performed only by experienced technicians.)

Notice the area on the P.C. board that I have designated as "A" in the diagram. This is the positive voltage going to the motor. "B" is the negative or return. Notice the row of pins in the center of the P.C. board. These are the underside of a switch. Ignore the posts labeled "D" and count down to the seventh pin starting at point "F". Include in your count the flattened pin. Pins 7 and 8 are normally open (verify this with a VOM in the resistance measurement mode). These pins are closed when the unit is in the record position (verify this also). Pin 8 to 9 function opposite.

These three pins (7,8,9) should have no other circuit traces going to them. This fact will also help you identify the pins. Add wiring as shown. If you cannot keep the capacitor within the cassette unit (I was not, but have found a nice place for it within the PET), punch a hole in the cassette case and run wires through the hole to the capacitor. Add from 5000 uF to 10,000 uF of capacitance to the circuit. The greater the capacitance, the more space is added. I am using 5200 uF wired as a 2200 uF and a 3000 uF parallel. Remember that capacitance is added when wired in parallel, not in series. Points "E" in the diagram indicate the intersection of the capacitor wires from my PET chassis-mounted capacitors, to the wires running out of the tape unit. I have used "disconnect" terminal lugs at these points, which enable me to change the capacitance easily at any time.

Figure 4 shows an alternative scheme which accomplishes the same thing as Figure 3, but will add capacitance to the motor at all times, even during fast forward. I used this scheme for months until I got the nerve to figure out the precise wiring of the unit itself.

Figure 4



Alternative method of adding capacitance to the motor. The edge connector is the one used to connect cassette to the main P.C. board.

This modification will eliminate the necessity for software subroutines to increase space between tape blocks and insure you trouble free data handling for the life of your system—or should I say for the life of the capacitor. **C**

ViewBAM and 1541 Disk Directory Programs

by Christopher Phillips

The following two programs are revised versions of programs found in the VIC-1541 User's Manual and on the TEST/DEMO disk.

Both programs are modified to allow selection of a unit number from 8 to 12. This provides selective access to any one of up to 5 disk drives daisy-chained together (the capacity of the VIC 20 and the Commodore 64).

The 64 View BAM program utilizes the 64's 40 column screen to display all 35 tracks of the disk on one screen.

The Disk Directory program has an added option of loading a program from the disk drive without having to exit the directory program.

Disk Directory

```
1 PRINT"▯▯▯████████████████8":PRINT"▯";:INPUT"UNIT NUMBER";UNIT
2 IF UNIT<8 OR UNIT>12 THEN GOTO 1
3 CLOSE2:CLOSE15
4 OPEN2,UNIT,15
5 PRINT"▯":GOTO 10000
10 OPEN1,UNIT,0,"$0"
20 GET#1,A$,B$
30 GET#1,A$,B$
40 GET#1,A$,B$
50 C=0
60 IF A$<>"" THEN C=ASC(A$)
```

```
70 IF B$<>"" THEN C=C+ASC(B$)*256
80 PRINT" "MID$(STR$(C),2);TAB(3);" ";
90 GET#1,B$:IF ST<>0 THEN 1000
100 IF B$<>CHR$(34) THEN 90
110 GET#1,B$:IF B$<>CHR$(34)THEN PRINTB$;:GOTO110
120 GET#1,B$:IF B$=CHR$(32) THEN 120
130 PRINT TAB(22);:C$=""
140 C$=C$+B$:GET#1,B$:IF B$<>""  THEN 140
150 PRINT LEFT$(C$,3)
160 GET T$:IF T$<>"" THEN GOSUB 2000
170 IF ST=0 THEN 30
1000 PRINT" BLOCKS FREE"
1010 CLOSE1:GOTO 10000
2000 IF T$="Q" THEN CLOSE1:END
2010 GET T$:IF T$="" THEN 2000
2020 RETURN
4000 REM DISK COMMAND
4010 C$="":PRINT">";
4011 GETB$:IFB$="" THEN4011
4012 PRINTB$;:IF B$=CHR$(13) THEN 4020
4013 C$=C$+B$:GOTO 4011
4020 PRINT#2,C$
5000 PRINT"■";
5010 GET#2,A$:PRINTA$;:IF A$<>CHR$(13)GOTO5010
5020 PRINT"■"
10000 PRINT "D-DIRECTORY"
10010 PRINT ">-DISK COMMAND"
10020 PRINT "Q-QUIT PROGRAM"
10030 PRINT "S-DISK STATUS "
10040 PRINT "/-LOAD PROGRAM"
10100 GETA$:IFA$=""THEN10100
10200 IF A$="D" THEN 10
10300 IF A$="." OR A$=">" OR A$=">" THEN 4000
10310 IF A$="Q" THEN CLOSE 2:CLOSE15:END
10320 IF A$="S" THEN 5000
10340 IF A$="/" THEN GOSUB 11000:GOTO10000
10999 GOTO 10100
11000 INPUT"LOAD";B$:IF B$=""ORB$="E"ORB$="Q"ORB$="BYE"THEN B$="":RETURN
11010 CLOSE 2:CLOSE 15
11020 PRINT"■■LOAD";CHR$(34);"0:";B$;CHR$(34);",";UNIT:PRINT"■■■■RUN:IIIIIII";
```

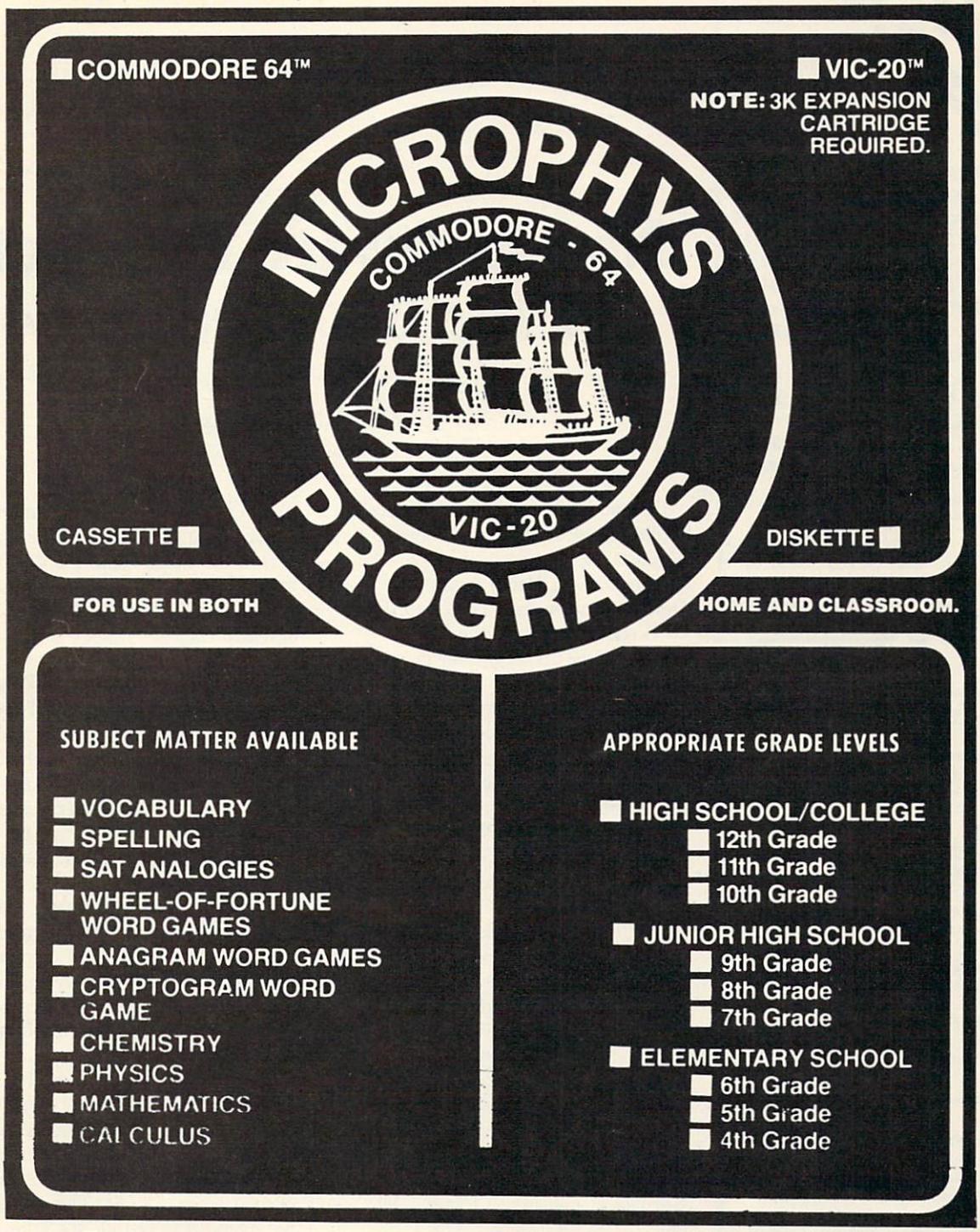
ViewBAM

```
40 REM
50 REM ---VIEW BAM FOR COMMODORE 64---
70 REM
80 PRINT"I■■■■■■■■■■■■■■■■■■3":PRINT"J";:INPUT"WHICH DRIVE";UNIT
```

```
85 IF UNIT<8 OR UNIT>12 THEN 80
100 REM ********************************
101 REM *     VIEW BAM FOR 64 DISK     *
102 REM ********************************
105 OPEN 15,UNIT,15
110 PRINT#15,"I0":NU$="N/A N/A N/A N/A N/A":Z4=1
120 OPEN 2,UNIT,2,"#"
130 Y$="
140 X$="
150 DEF FNS(Z) = 2↑(S-INT(S/8)*8) AND (SB(INT(S/8)))
160 PRINT#15,"U1:";2;0;18;0
170 PRINT#15,"B-P";2;1
180 PRINT"
190 Y=24:X=1:GOSUB430
200 FORI=0TO20:PRINT:PRINT"I"RIGHT$(STR$(I)+" ",3);:NEXT
210 GET#2,A$
220 GET#2,A$
230 GET#2,A$
240 TS=0
250 FORT=1TO35:GOSUB450
260 Y=24:X=T+4:GOSUB430:GOSUB540:NEXT
270 REM GETZ$:IFZ$=""THEN270:PRINT"
280 Y=24:X=1:GOSUB430
290 FORI=0TO20:PRINT:PRINT"I"RIGHT$(STR$(I)+" ",3);:NEXT
300 REM FORT=18TO35
310 REM GOSUB450
320 REM Y=24:X=T-13:GOSUB430:GOSUB540:NEXT
330 REM FORI=1TO1000:NEXT
340 PRINT"
350 PRINT#15,"B-P";2;144
360 N$="":FORI=1TO20:GET#2,A$:N$=N$+A$:NEXT
370 PRINT" "N$" "TS-17;"BLOCKS FREE"
380 PRINT"        <HIT ANY KEY TO EXIT>"
382 FOR I=1TO1000:NEXT:PRINT"      BLOCK ALLOCATION MAP (BAM) "
385 GETZ$:IFZ$=""THEN385:PRINT"
390 PRINT"
400 INPUT"ANOTHER DISKETTE  N";A$
410 IFA$="Y"THENRUN
420 IFA$<>"Y"THENPRINT#15,"I0":GOTO 1000
430 PRINTLEFT$(Y$,Y)LEFT$(X$,X)"";
440 RETURN
450 GET#2,SC$:SC=ASC(RIGHT$(CHR$(0)+SC$,1))
460 TS=TS+SC
470 GET#2,A$:IFA$=""THENA$=CHR$(0)
480 SB(0)=ASC(A$)
490 GET#2,A$:IFA$=""THENA$=CHR$(0)
```

```
500 SB(1)=ASC(A$)
510 GET#2,A$:IFA$=""THENA$=CHR$(0)
520 SB(2)=ASC(A$)
530 RETURN
540 PRINT"■■"RIGHT$(STR$(T),1);"■■";
550 REM  PRINTT"   "SC"  "SB(0)" "SB(1)" "SB(2)=CHR$(0)
560 IFT>24ANDS=18THEN:PRINTMID$(NU$,Z4,1);:GOTO660
570 FORS=0TO20
580 IF T<18THEN620
590 IFT>30ANDS=17THEN:PRINTMID$(NU$,Z4,1);:GOTO660
600 IFT>24ANDS=18THEN:PRINTMID$(NU$,Z4,1);:GOTO660
610 IFT>24ANDS=19THENPRINTMID$(NU$,Z4,1);:GOTO660
620 IFT>17ANDS=20THENPRINTMID$(NU$,Z4,1);:Z4=Z4+1:GOTO660
630 PRINT"■";
640 IF FNS(S)=0 THEN PRINT"+";:GOTO660
650 PRINT"■+";:REMRIGHT$(STR$(S),1);Z4,1);:GOTO72
660 PRINT"■■";
670 NEXT
680 RETURN
890 REM --- DISK COMMAND ---
900 OPEN15,9,15
910 INPUT"DISK COMMAND";C$:PRINT#15,C$
920 INPUT#15,E,ER$,E1,E2:IF E<20 THEN STOP
930 PRINT E;ER$;E1;E2:CLOSE15:STOP
1000 REM CLOSE15:LOAD"0:DIREC*",UNIT
```

MICROPHYS PROGRAMS have been used successfully in school and colleges throughout the country. Now, this educational and recreational software is available for use in your own home.



The Microphys catalog is available from leading stores and computer dealers or you may phone or write to:

MICROPHYS PROGRAMS
MARKETING DIVISION
1737 West 2nd Street
Brooklyn, New York 11223
(212) 375-5151

Foreign and Domestic Dealer and Distributor Inquires Welcomed.

New Software Products for the Commodore 64

by April M. Koppenhaver

Commodore is proud to announce the recent release of three new software products for the Commodore 64: The Commodore 64 Macro Assembler Development System, The Commodore 64 Screen Editor, and the PET EMULATOR. To continually encourage and technically support third party software developers, Commodore specifically designed The Commodore 64 Macro Assembler Development System and The Commodore 64 Screen Editor for programmers. To take full advantage of existing software, the PET EMULATOR package modifies the Commodore 64 so that it will operate identically to the 2.0 BASIC PET 2001.

The process of translating a mnemonic or symbolic form of a computer program to actual machine code is called assembly. A program which performs this translation is an assembler. The symbolic form of the program is referred to as source code and the actual machine form is referred to as object code. The symbols used and the rules of association for those symbols are the assembly language. In general, one assembly language statement will translate into one machine instruction. This distinguishes an assembler from a compiler which may produce many machine instructions from a single statement. An assembler which executes on a computer other than the one for which code is generated, is called a cross-assembler. Use of cross-assemblers for program development on microprocessors is common because often a microcomputer system has fewer resources than are needed for an assembler. However, in the case of the Commodore 64, this is not true. With a floppy disk and printer, the system is well suited for software development.

The Commodore 64 Macro Assembler Development System

lets you program in the native 6500 series assembler language on the Commodore 64. The software package provides you with a very powerful macro assembler, editor, loaders, and two machine language monitors, along with other support routines. This is everything you need to create, assemble, load, and execute 6500 series assembly language code. These development tools operate in a similar manner and provide the same level of direct machine interface as the assemblers on much larger computers.

The software contained on the diskette, as well as the user's manual, is directed towards the experienced computerist who is already familiar with 6500 series assembly and the operations of the Commodore 64. This product is not intended to provide the knowledge of "how to" in assembly language. (In the preface of the manual, several references are listed which can be consulted for a detailed description of 6502 assembly language and the Commodore 64.)

The Commodore 64 Macro Assembler Development System manual is divided into five parts:

Part One, "Introduction"—To provide a brief description of how an assembler works, along with an explanation of general terminology.
Part Two, "Capabilities and Conventions"—To describe capabilities and conventions used by this assembler.
Part Three, "Creating and Editing Assembly Source Files"—To describe how to create and edit an assembly language source file. This includes instructions for loading a support program or "wedge" which gives the user additional commands for maintaining the disk and loading and running programs. Also, operating instructions for loading and running the Editor64 program are given. This program lets the user create and edit assembly source files.
Part Four, "Assembling and Testing a Program"—To contain information on the programs that let the user assemble, test, and debug object programs.
Part Five, "Appendices"—Charts and tables as a summary.

The Commodore 64 Screen Editor software package lets you create BASIC programs with the aid of a screen field editor. The Screen Editor program provides the user with a powerful software tool that lets you define and edit fields that are input through the CRT screen. Screen design and the input and editing of data have always been one of the more difficult tasks facing the BASIC programmer. With the Commodore 64 Screen Editor, screen design and the input and editing of data is not only easier, but accomplished with the accuracy and speed of assembly language programming.

This package contains everything that the user will need to create, edit, and manipulate data fields from within a BASIC program. The software and manual are both directed towards the experienced computer user that already has some familiarity with the BASIC language and the operations of the Commodore 64.

The Commodore PET EMULATOR software package lets you execute on your Commodore 64 programs that were originally designed for the PET. The PET Emulator actually modifies the Commodore 64 so it will operate almost identically to the 2.0 BASIC PET 2001. This modification consists of two parts: memory re-configuration and system interaction interpretation. The exact technical specifications of just how the PET EMULATOR operates are outside the scope of the manual. However, the more important conceptual information of its operation are presented clearly and concisely.

There are many other quality software products headed towards completion. For your information and enjoyment, we intend to include a review of the new software products as they are released. **C**

More Columns for the VIC Typewriter

by Ron Cohen

When using a printer with the VIC Typewriter, the printer prints 66 characters on one line. In order to expand this to anything past 66 characters the following should be done:

1. Load the program
2. Run the program
3. Hit Run/Stop and restore
4. POKE V+456,N
5. POKE Z+518,N

Where "N" is the number of characters desired on one line, i.e., N=80. After the second POKE, when READY comes up on the screen, the user should type RUN, and the program will print out as desired. **C**

User Groups Listing

ALABAMA

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

ALASKA

COMPOOH-T
c/o Box 118
Old Harbor, AK 99643
907-286-2213

ARIZONA

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

Commodore User Group
Metro Computer Store
4500 E. Speedway,
Suite 13
Tucson, AZ 85712
602-323-3116

Central Arizona PET People
842 W. Calle del Norte
Chandler, AZ 85224
602-899-3622
Roy Schahrer

ACUG
c/o Home Computer Service
2028 W. Camelback Rd.
Phoenix, AZ 85015
602-249-1186
Dan Deacon
First Wed. of month

West Mesa VIC
2351 S. Standage
Mesa, AZ 85202
Kenneth S. Epstein

ARKANSAS

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

Booneville 64 Club
c/o A.R. Hederich
Elementary School
401 W. 5th St.
Booneville, AR 72927
Mary Taff

CALIFORNIA

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
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

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
7615 Leviston Ave.
El Cerrito, CA 94530
(415) 527-9286
Bill MacCracken

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

The Commodore Connection
2301 Mission St.
Santa Cruz, CA 95060
(408) 425-8054
Bud Massey

San Fernando Valley
Commodore Users Group
21208 Nashville
Chatsworth, CA 91311
(213) 709-4736
Tom Lynch
2nd Wed. 7:30

VACUUM
277 E. 10th Ave.
Chico, CA 95926
(916) 891-8085
Mike Casella
2nd Monday of month

VIC 20 Users Group
2791 McBride Ln. #121
Santa Rosa, CA
(707) 575-9836
Tyson Verse

South Bay Commodore Users Group
1402 W. 218th St.
Torrance, CA 90501
Contact: Earl Evans

Slo VIC 20/64 Computer Club
1766 9th St.
Los Osos, CA

The Diamond Bar R.O.P. Users Club
c/o Rincom School
2800 Hollingworth
West Covina, CA 91792
(213) 965-1696
Don McIntosh

Commodore Interest Association
c/o Computer Data
14660 La Paz Dr.
Victorville, CA 92392
Mark Finley

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

PETs and Friends
129 NE 44 St.
Miami, FL 33137
Richard Plumer

Sun Coast VICs
P.O. Box 1042
Indian Rocks Beach, FL 33535
Mark Weddell

Bay Commodore Users Group
c/o Gulf Coast Computer Exchange
241 N. Tyndall Pkwy.
P.O. Box 6215
Panama City, FL 32401
(904) 785-6441
Richard Scofield

Gainesville Commodore Users Club
3604-20A SW 31st Dr.
Gainesville, FL 32608
Louis Wallace

64 Users Group
P.O. Box 561689
Miami, FL 33156
(305) 274-3501
Eydie Sloane

Brandon Users Group
108 Anglewood Dr.
Brandon, FL 33511
(813) 685-5138
Paul Daugherty

Commodore 64/VIC 20 User Group
Martin Marietta Aerospace
P.O. Box 5837, MP 142
Orlando, FL 32855
(305) 352-3252/2266
Mr. Earl Preston

GEORGIA

VIC Educators Users Group
Cherokee County Schools
110 Academy St.
Canton, GA 30114
Dr. Al Evans

IDAHO

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

S.R.H.S. Computer Club
c/o Salmon River H.S.
Riggins, ID 83549
Barney Foster

Commodore Users
548 E. Center
Pocatello, ID 83201
(208) 233-0670
Leroy Jones

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

user groups

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 Users Group
1608 Benton Street
Rockford, IL 61107

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

VIC Chicago Club
3822 N. Bell Ave.
Chicago, IL 60618
John L. Rosengarten

Chicago Commodore 64 Users & Exchange Group
P.O. Box 14233
Chicago, IL 60614
Jim Robinson

Fox Valley PET Users Group
833 Willow St.
Lake in the Hills, IL 60102
(312) 658-7321
Art DeKneef

The Commodore 64 Users Group
4200 Commerce Ct., Suite 100
Lisle, IL 60532
(312) 369-6525
Gus Pagnotta

Oak Lawn Commodore Users Group
The Computer Store
11004 S. Cicero Ave.
Oak Lawn, IL 60453
(312) 499-1300
Bob Hughes

INDIANA

PET/64 Users
10136 E. 96th St.
Indianapolis, IN 46256
(317) 842-6353
Jerry Brinson

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

VIC Indy Club
P.O. Box 11543
Indianapolis, IN 46201
(317) 898-8023
Ken Ralston

Northern Indiana Commodore Enthusiasts
927 S. 26th St.
South Bend, IN 46615
Eric R. Bean

Commodore Users Group
1020 Michigan Ave.
Logansport, IN 46947
(219) 722-5205
Mark Bender

IOWA

Commodore User Group
114 8th St.
Ames, IA 50010

Quad City Commodore Club
1721 Grant St.
Bettendorf, IA 52722
(319) 355-2641
John Yigas

Commodore Users Group
965 2nd St.
Marion, IA 52302
(319) 377-5506
Vern Rotert
3rd Sun. of month

Siouxland Commodore Club
2700 Sheridan St.
Sioux City, IA 51104
(712) 258-7903
Gary Johnson
1st & 3rd Monday of month

KANSAS

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

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

Commodore Users Group
6050 S. 183 St. West
Viola, KS 67149
Walter Lounsbery

KENTUCKY

VIC Connection
1010 S. Elm
Henderson, KY 42420
Jim Kemp

LOUISIANA

Franklin Parish Computer Club
#3 Fair Ave.
Winnisboro, LA 71295
James D. Mays, Sr.

NOVA
917 Gordon St.
New Orleans, LA 70117
(504) 948-7643
Kenneth McGruder, Sr.

VIC 20 Users Group
5064 Bowdon St.
Marrero, LA 70072
(504) 341-5305
Wayne D. Lowery, R.N.

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

Long Lines Computer Club
323 N. Charles St., Rm. 201
Baltimore, MD 21201
Gene Moff

VIC & 64 Users Group
The Boyds Connection
21000 Clarksburg Rd.
Boyds, MD 20841
(301) 428-3174
Tom DeReggi

VIC 20 Users Group
23 Coventry Lane
Hagerstown, MD 21740
Joseph Rutkowski

Hagerstown Users Group
1201-B Marshall St.
Hagerstown, MD 21740
(301) 790-0968
Greg Stewart
1st & 3rd Friday of month 6:30 p.m.

Rockville VIC/64 Users Group
13013 Evanstown St.
Rockville, MD 20853
(301) 946-1564
Meryle or Tom Pounds

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

The Boston Computer Society
Three Center Plaza
Boston, MA 02108
(617) 367-8080
Mary E. McCann

VIC Interface Club
c/o Procter & Gamble Inst. Shop
780 Washington St.
Quincy, MA 02169
C. Gary Hall

Masspet Commodore Users Group
P.O. Box 307
East Taunton, MA 02718
David Rogers

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

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

VIC Commodore Club
2765 Bristol Rd.
Dowling, MI 49050
Bill Kelley

W. Michigan VIC 20-64 Users
1311 Portland NE
Grand Rapids, MI 49505
(616) 459-7578
Jim D'Haem

VIC for Business
6027 Orchard Ct.
Lansing, MI 48910
Mike Marotta

South Computer Club
South Jr. High School
45201 Owen
Belleville, MI 48111
Ronald Ruppert

Commodore Users Group
c/o Eaton Rapids Medical Clinic
101 Spicerville Hwy.
Eaton Rapids, MI 48827
Albert Meinke III, M.D.

South East Michigan Pet Users Group
Box 214
Farmington, MI 48024
Norm Eisenberg

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
633 Bent Oak Drive
Lake St. Louis, MO 63367
(314) 625-2701 or 625-4576
Tony Ott

VIC INFONET
P.O. Box 1069
Branson, MO 65616
(417) 334-6099
Jory Sherman

Worth County PET Users Group
Grant City, MO
(816) 564-3551
David Hardy

Mid-Missouri Commodore Club
1804 Vandiver Dr.
Columbia, MO 65201

(314) 474-4511
Phil Bishop

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
18 Alpine Drive
Wayne, NJ 07470

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

Educators Advisory
P.O. Box 186
Medford, NJ 08055
(609) 953-1200
John Handfield

VIC-TIMES
46 Wayne Street
Edison, NJ 08817
Thomas R. Molnar

VIC 20 User Group
67 Distler Ave.
W. Caldwell, NJ 07006
(201) 284-2281
G. M. Amin

VIC Software Development
Club
77 Fomalhaut Ave.
Sewell, NJ 08080
H. P. Rosenberg

ACGNJ PET/VIC/CBM
User Group
30 Riverview Terr.
Belle Mead, NJ 08502
(201) 359-3862
J. M. Pylka

NEW HAMPSHIRE

Northern New England
Computer Society
P.O. Box 69
Berlin, NH 03570

TBH VIC-NICs
P.O. Box 981
Salem, NH 03079

NEW MEXICO

Commodore Users Group
6212 Karlson, NE
Albuquerque, NM 87113
(505) 821-5812
Danny Byrne

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
P.O. 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

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

West Chester County VIC
Users Group
P.O. Box 146
Pelham, NY 10552
Joe Brown

SPUG
4782 Boston Post Rd.
Pelham, NY 10803
Paul Skipski

VIC 20 User Club
151-28 22nd Ave.
Whitestone, NY 11357
Jean F. Coppola

VIC 20 User Club
339 Park Ave.
Babylon, NY 11702
(516) 669-9126
Gary Overman

VIC User Group
1250 Ocean Ave.
Brooklyn, NY 11230
(212) 859-3030
Dr. Levitt

L&M Computer Club
VIC 20 & 64
4 Clinton St.
Tully, NY 13159
(315) 696-8904
Dick Mickelson

Commodore Users Group
1 Corwin Pl.
Lake Katrine, NY 12449
J. Richard Wright

8*8 Enthusiasts
P.O. Box 28 Rhodes Rd.
Apalachin, NY 13732
Keith Merrill

VIC 20/Commodore 64
Users Group
31 Maple Dr.
Lindenhurst, NY 11757
(516) 957-1512
Pete Lobol

VIC Information Exchange
Club
336 W. 23 St.
Deer Park, NY 11729
Tom Schlegel
SASE & phone please

New York Commodore
Users Group
380 Riverside Dr., 7Q
New York, NY 10025
(212) 566-6250
Ben Tunkelang

Parsippany Computer Group
51 Ferncliff Rd.
Morris Plains, NJ 07950
(201) 267-5231
Bob Searing

Capitol Dist. Commodore 64/VIC Users
Group
363 Hamilton St.
Albany, NY 12210
(518) 436-1190
Bill Pizer

Hudson Valley Commodore Club
1 Manor Dr.
Woodstock, NY 12498
F.S. Goh
1st Wednesday of month

LIVICS (Long Island VIC Society)
20 Spyglass Lane
East Setauketm, NY 11733
(516) 751-7844
Lawrence Stefani

VIC Users Group
c/o Stoney Brook Learning Center
1424 Stoney Brook Rd.
Stoney Brook, NY 11790
(516) 751-1719
Robert Wurtzel

NORTH CAROLINA

Amateur Radio PET Users
Group
P.O. Box 30694
Raleigh, NC 27622
Contact: Hank Roth

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

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

VIC Users Club
Rt. 11, Box 686
Hickory, NC 28601
Tim Gromlovits

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, OH 43204
(614) 274-6451
Contact: Philip H. Lynch

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

Chillicothe Commodore
Users Group
P.O. Box 211
Chillicothe, OH 45601
William A. Chaney

Licking County 64 Users Group
323 Schuler St.
Newark, OH 43055
(614) 345-1327

OKLAHOMA

Southwest Oklahoma
Computer Club
P.O. Box 6646
Lawton, OK 73504
Garry Lee Crowell
1:30 1st Sunday at
Lawton City Library

Tulsa Area Commodore Users Group
Tulsa Computer Society
P.O. Box 15238
Tulsa, OK 74112
Annette Hinshaw

OREGON

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

PENNSYLVANIA

PET User Group
Gene Beals
P.O. 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

Westmoreland Commodore
Users Club
c/o DJ & Son Electronics
Colonial Plaza
Latrobe, PA 15650
Jim Mathers

COMPSTARS
440 Manatawny St.
Pottstown, PA 19464
Larry Shupinski, Jr.
Meet at Audio Video
Junction

Commodore Users Club
3021 Ben Venue Dr.
Greensburg, PA 15601
(412) 836-2224
Jim Mathers

VIC 20 Programmers, Inc.
c/o Watson Woods
115 Old Spring Rd.
Coatesville, PA 19320
Robert Gougher

G.R.C. User Club
300 Whitten Hollow Rd.
New Kensington, PA 15068

user groups

Bill Bolt

NADC Commodore Users Club
248 Oakdale Ave.
Horsham, PA 19044
Norman McCrary

PUERTO RICO

CUG of Puerto Rico
RFD #1, Box 13
San Juan, PR 00914
Ken Burch

VIC 20 User Group
655 Hernandez St.
Miramar, PR 00907
Robert Morales, Jr.

RHODE ISLAND

Irving B. Silverman, CPA
160 Taunton Ave.
E. Providence, RI 02914
Contact: Michelle Chavanne

Newport VIC/64 Users
10 Maitland Ct.
Newport, RI 02840
(401) 849-2684
Dr. Matt McConeghy

SOUTH CAROLINA

Beauford Technical College
100 S. Ribaut Rd.
Beauford, SC 29902
Dean of Instruction

SOUTH DAKOTA

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

VIC/64 Users Club
203 E. Sioux Ave.
Pierre, SD 57501
(605) 224-4863
Larry Lundeen

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

Commodore User Club
Metro Computer Center
1800 Dayton Blvd.
Chattanooga, TN 37405
Mondays 7:30 pm

TEXAS

SCOPE
1020 Summit Circle
Carrolton, TX 75006

PET Users
2001 Bryan Tower
Suite 3800
Dallas, TX 75201

Larry Williams
P.O. 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

Corpus Christi Commodores
3650 Topeka St.
Corpus Christi, TX 78411
(512) 852-7665
Bob McKelvy

Commodore Users Group
5326 Cameron Rd.
Austin, TX 78723
(512) 459-1220
Dr. Jerry D. Frazee

VIC Users Group
3817 64th Dr.
Lubbock, TX 79413

Southeast Houston VIC
Users Group
11423 Kirk Valley Dr.
Houston, TX 77089
(713)481-6653

64 Users Group
2421 Midnight Circle
Plano, TX 75075
S.G. Grodin

UTAH

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

The Commodore Users
Club
742 Taylor Avenue
Ogden, UT 84404
Contact: Todd Woods Kap,
President
David J. Shreeve,
Vice President

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

VIC 20 Users
324 N. 300 W.
Smithfield, UT 84335
Dave DeCorso

Northern Utah VIC & 64
Users Group
P.O. Box 533
Garland, UT 84312
David Sanders

The Utah Commodore Users Group
652 West 700 North
Clearfield, UT 84015
(801) 776-3950
Rodney Keller or Richard Brenchly

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
P.O. Box 2004
Dale City, VA 22193
(703) 680-2270
James Hogler

Tidewater Commodore
Users Group
4917 Westgrove Rd.
Virginia Beach, VA 23455
Fred Monson

Fredericksburg Area
Computer Enthusiasts
P.O. Box 324
Locust Grove, VA 22508
(703) 972-7195
Michael Parker

Commonwealth 20/64
Users Group
1773 Wainwright Dr.
Reston, VA 22090
(703) 471-6325
Tal Carawan, Jr.

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

VIC 20 Computer Club
947 N. Burroughs Ave.
Oak Harbor, WA 98277
Michael D. Clark

Central Washington
Commodore Users Group
1222 S. 1st St.
Yakima, WA 98902
Tim McElroy

WEST VIRGINIA

Personal Computer Club
P.O. Box 1301
Charleston, WV 25325
Cam Cravens

WISCONSIN

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

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

Commodore User Group
1130 Elm Grove St.
Elm Grove, WI 53122
Tony Hunter

Commodore 64 Software
Exchange Group
P.O. Box 224
Oregon, WI 53575
E. J. Rosenberg

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 Mr. Brown
Valley Heights Secondary School
Box 159
Langton, Ont. N0E 1G0

Vancouver PET Users Group
P.O. 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

W.P.U.G.
9-300 Enniskillen Ave.
Winnipeg, Manitoba R2V0H9
Larry Neufeld

VIC-TIMS
2-830 Helena St.
Trail, British Columbia,
V1R 3X2
(604) 368-9970
Greg Goss

Arva Hackers
Medway High School
Arva, Ontario N0M 1C0
D. Lerch

Nova Scotia Commodore
Computer Users Group
66 Landrace Cres.
Dartmouth, N.S. B2W 2P9
Andrew Cornwall

Bonnyville VIC Cursors
Box 2100
Bonnyville, Alberta T0A 0L0
(403) 826-3992
Ed Wittchen

FINLAND

VIC-Club in Helsinki
c/o Matti Aarnio
Linnustajankj 2B7
SF-02940 ESP00 94
Finland

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.

Club de Usarios Commodore
Sigma del Norte
Mol del Valle, Local 44
Garza Garcia, N.L. 66220

NEW ZEALAND

Commodore Users Group
Meet at VHF Clubrooms
Hazel Ave.
Mount Roskill
3rd Wed. of month, 7:30 pm
Roger Altena 278-5262

Nelson VIC Users Group
c/o P.O. Box 860
Nelson, New Zealand
Peter Archer

E.R. Kennedy
c/o New Zealand Synthetic
Fuels Corp. Ltd.
Private Bag
New Plymouth

NORWAY

VIC Club of Norway
Nedre Bankegt 10,
1750 Halden
Norway

UNITED KINGDOM

North London Hobby
Computer Club
Dept. of Electronics &
Communications
Engineering
The Polytechnic of North
London
Holloway Rd.
London N7 8DB

Croydon Microcomputer Club
111 Selhurst R.
Selhurst, London SE25 6LH
01-653-3207
Vernon Gifford

User Bulletin Board

User Groups Forming:

IOWA

Contact Don Geisinger
1407 W. Milwaukee
Storm Lake 50588
(712) 732-4711

LOUISIANA

VIC 20 Group
Contact Rick Lane
4521 General Ewell Drive
Bossier City 71112

OHIO

Contact Bill Novak
2299 W. 11th
Cleveland 44113
(216) 696-5093

PENNSYLVANIA

South Philly Commodore Users Group
Contact Delta Video
1425 W. Passyunk Avenue
Philadelphia 19145
(215) 467-4900

TENNESSEE

VIC 20 and Commodore 64
Contact Edward W. Pritchard
7405 Oxmoor Road, Rt. #20
Knoxville 37921

VIRGINIA

Contact Jimmy Richardson
Rt. 1, Box 12-A
Elk Creek 24326
(703) 655-4144

ANNOUNCEMENTS

MASSPET, a Commodore user group based in East Taunton, Massachusetts, has an electronic bulletin board. Access it with your modem by dialing 617-824-4878 Monday through Friday 7 pm to 9 am and all day Saturday and Sunday.

CLASSIFIED

Starting a User Group

by Jeff Hand

The how's and why's of getting a group going.

You've brought your computer home, read the manual and played a few games, and now you're ready for something more challenging. You thought paying for the computer was the worst hurdle you had to overcome. But belatedly you realize there is a lot to learn before you can accomplish all those things you envisioned: the synthetic symphony, the video game that will make you a fortune, the computer brain to control your house, or the program that was going to do most of your homework.

What do you do now? Embark on the perilous task of overcoming the evil computer on your own? Hope a computer guru moves into the neighborhood? Call customer support on a daily basis? (You would quickly run out of dimes, and besides customer support is not set up to walk you through a programming problem.) Get your Ph.D in computer science from Harvard?

Well, I don't think any of these methods will work too well for most people. Solution: *learn from a local user group.* This is usually the easiest and most enjoyable way to get more information about your computer. (When I want to find out what's going on inside Commodore I go to a user group meeting: they usually know more than I do.) But what if there are no user groups in your area? Great! That gives you a chance to start your own!

Getting Started

One person can do all the work and pull off a user group organization, but it's an awful lot of work. It would be easier and more enjoyable to get several people to help in the pre-planning stages. Talk to a friend from work, or make acquaintances through a dealer who might be interested in starting a user group.

Once you have lined up several people interested in organizing a user group, call a planning meeting.

To make your planning meeting a success there are several items of business that have to be taken care of.

1. Draft a statement of purpose for the user group. For example, "This organization shall promote computer education for the Commodore computer."
2. Select temporary officers to administer the organization until permanent officers can be elected by the general membership. The president and the secretary are really all you need at this time.
3. Decide on a working name for the organization. Keep in mind that existing trademarks and logos cannot be used. Computer names incorporated into your organization name can be used only in their full, trademarked form. In other words, you can call yourselves the "Upstate VIC 20 User Group", but NOT the "Upstate VICPICS". The safest choice is to create an original name that identifies the function and intent of your group in a straightforward way.
4. Draw up tentative bylaws for the group to be presented to the general membership for a vote. Bylaws are guiding rules that define responsibilities, duties and privileges of the members of the group, so the organization can carry out the purpose for which it was formed. The following topics should be covered in the bylaws: name, purpose of the group, membership, guidelines for officers and chairpersons, election procedure, duties of officers, funds, discipline, provisions for amending the bylaws, and finally procedures for dissolution of the group.
5. Decide on a date, and location of the first meeting. Someone's home is probably the easiest arrangement to make for expediency. But this usually puts an unnecessary amount of responsibility on one individual, so as soon as arrangements can be made move the meeting to a business location or rotate the meeting place to each constituent's home. Make sure you have adequate facilities to meet the turn out.
6. Make arrangements to notify prospective members of the meeting. This can be done a number of ways. An advertisement in the local newspaper or computer magazine can work, for instance. Commodore will list your user group in *Power/Play* and *Commodore: The Microcomputer Magazine* for free; just let us know. Another way to notify prospective members is by putting your message on telecommunication services such as the Commodore Information Network (on CompuServe), The Source, or a local electronic bulletin board in your area. Posting the formation of a user group at the local school, church or Commodore dealer in your area will almost guarantee you a response.
7. Make a list of projects the group wants to undertake. Try to provide services to make the user group valuable to the members. The first

very obvious project, often taken for granted, is the format of the meetings. Meetings ideally should be open, informal, present up-to-date information and allow time for interaction among users. All of us have experienced the pain of stifled yawns at one meeting or another. To avoid this, keep the formalities to a minimum and have the activities of the meeting planned in advance.

Another project the group might want to undertake is establishing a library of related manuals, magazines, and public domain software for all the members to have access to. A newsletter is a standard project for most user groups. If your group has a lot of energy and talent some other good projects would be to establish a telecommunications bulletin board, organize a computer show, and arrange product discounts with local businesses in your area.

Commodore Information Network Can Help

Before we go on, I'd like to plug the Commodore Information Network in a little more detail. The Commodore Information Network is an ideal medium for user groups, because it provides a place to trade ideas, information, and experiences with other users, both through a bulletin board and in a real-time conference mode. Not only that, but we've set aside unlimited computer space for public domain software. Your group can contribute public domain software to the archives or use any of the software for your own library for only the cost of connect time! The Commodore Information Network will select the best public domain program submitted during the month and offer it for sale on the system. Royalties will be paid to the user group or author who submitted the program.

A database has also been set aside specifically for user groups across the country to post announcements and newsletters on the system for national distribution. Meetings can be carried out over the Network with the conference mode. This is particularly useful if your organization is spread across a large geographic area. What better way can you think of to carry out a computer user group meeting than using the powerful capabilities of the computer without leaving your home?

The Network can be accessed by anyone in North America. All that's needed is a phone, a modem, computer and terminal software, and you're ready to take advantage of all those services for your user group.

A Typical Meeting

Now that you've decided what you want to do with the user group and completed all the arrangements for the membership, you're ready for the first meeting. Here is an outline of the activities of the first meeting.

Allow the people who attend to get to know each other. Introduce yourself to people you're not familiar with and let them know the purpose of the meeting. Fifteen to twenty minutes should be adequate time to let people mix and allow for latecomers. At this time the president should call the meeting to order. The first order of business is for those present to vote on the acceptance of the temporary presiding officers to continue their duties until an official election is held.

Next the presiding officer explains why the meeting has been called and the purpose of the organization. The objectives and activities of the group should be outlined clearly. The topic of objectives and activities is turned over to the general membership for an informal discussion. At the end of this general discussion a vote is taken on whether to continue the existence of the group. Assuming that the decision was affirmative, tentative bylaws are submitted to everyone for approval.

Arrangements are made for the time and place of the next meeting. The presiding officer announces that dues, as agreed upon in the bylaws, will be collected at the next meeting. Permanent officers will be elected at the next meeting. Then the meeting can be adjourned. Whew! How does it feel to make it through your first meeting?

At the second meeting the opening is the same. The first order of business is to nominate and elect the officers of the club. The elected president immediately takes control of the meeting. The treasurer announces that dues are payable. The treasurer should write receipts and keep copies, with the name, address, and phone number of each member.

The objectives of the group are reiterated by the president, who then opens the floor for discussion on topics of interest to the membership. For example, the discussion could be on how to educate the membership on programming skills. After a reasonable discourse the president closes the discussion and calls for a motion regarding the discussion. The motion is to get a member of the group to have a short course on assembler programming for the VIC 20. Or if your group is in no hurry, study groups can be created to investigate educational classes, the budget, fund raising, finances, public relations, newsletter, library, special events, or whatever is appropriate for the group.

Some No-No's

Several user groups have run into problems by making some bad decisions. Here are a few don'ts to help steer you away from unnecessary problems. Don't put any copyrighted information into your newsletters without written permission. Don't duplicate and distribute any copyrighted software, regardless of whether you're doing it for free or charging a fee. There is a tremendous amount of public domain software around that

will probably suit your purposes. The last piece of advice: don't print negative articles or make negative comments about any company unless you have hard evidence and are willing to go through the legal entanglements and the financial setbacks that this includes.

To Inc. or Not to Inc.

The user group will probably grow in leaps and bounds, so within a few months you may have to make a decision on whether to remain a home operation or to move on to something more serious to accommodate the growth and protect yourself. Usually a committee is assigned to the task of weighing the advantages and disadvantages of incorporation as a non-profit organization. Here is a quick outline of the options that might make the decision a little easier when the time comes.

The home operation is flexible, easy to control and more than adequate for most user groups. The disadvantage is that operating expenses are defrayed almost totally by the members and liability for the group is tied to each member.

Requirements for incorporation as a non-profit organization are usually governed by state law. The requirements vary from state to state, but I can guarantee it will be more work than the home operation. You will have to write rules, bylaws and a constitution, assign shareholders, directors, officers and designate the authority of each. These laws restrict the activities of the group. For example, your group cannot get involved in unrelated activities such as political activism or selling puppies to hungry tigers. Benefits to the membership are also curtailed; if there are any profits generated by the organization they cannot be distributed to the members. In the event the group is dissolved, all funds should usually go to a similar non-profit organization or a charity.

The biggest advantage of incorporating is to protect yourself by diverting liability from yourself to the corporation. A non-profit organization can be sued, but the managers and members of the group cannot. Tax-exempt status is a very attractive plus when considering incorporation. A non-profit organization can also apply for lower mail rate privileges. People are more likely to join a non-profit corporation because of the limited liability. The organization will last longer than anyone's individual participation. This means that the purpose for which the group was organized will continue indefinitely.

A Final Word

If you need a more substantial reason for starting a group than "Because it wasn't there!" a good reason is that you can meet people and make friends who have the same interests you do, and have fun while you're doing it. It's a rare person who is happy living the life of a computer hermit. A user group provides an easy way to meet people. The group also gives a feeling of comraderie and support in solving common problems.

Trading ideas and information on hardware, applications and programming are natural purposes for a user group. This information will open new horizons to help you make more effective use of your computer, giving you a feeling of accomplishment that you are one step closer to understanding your microcomputer. Instead of having one teacher you have a number of them!

Another advantage is that several people can buy equipment and services cheaper than an individual. The group can use its leverage to negotiate with vendors to get cheaper prices for the members.

While prestige, business or economic advantage, education, and social opportunities can be motives for joining a user group, the intangible rewards are the reasons people continue to pay dues. The respect, admiration, self-confidence and close relationships that develop cannot be measured.

This article is by no means the final word on user groups. If you need more information there are numerous books on the market that will tell you in much greater detail how to establish an organization, how to raise funds, how to administer effectively, how to incorporate without a lawyer and so on. If you want more information on becoming a tax-exempt non-profit corporation ask the Internal Revenue Service for Publication 557, "How to Apply for and Retain Exempt Status for Your Organization."

Have fun! **C**

that does not compute...

64 ROM Update

There have been three different Kernal ROM's used so far in the Commodore 64. The changes improved some internal features in the machines, but there is one side-effect. When POKEing to the screen with the first ROM, characters appeared because the color in color RAM was set to white. In the second ROM, the color RAM was set to the same color as the screen, so POKEs were invisible unless color RAM was also POKEd.

In the third and latest ROM, color memory is set to the same color as the cursor. The color RAM is changed only when the clear screen key (or CHR$ (147)) is used. This affects some programs on the market. Outside vendors have been alerted to the potential problem and are fixing their programs. Commodore itself is changing its 656 public domain programs to reflect the new ROMs, and they should be available again shortly.

To find out whether you have kernal 1 or kernal 2, PEEK at location 65408 (Hex $FF08). Kernal 1 will contain the value 170. Kernal 2 will contain 0.

"My Favorite VIC Cartridge"

March, 1983

In this article there is a mistake in Program 2 (page 87). Thanks to the readers whose ILLEGAL QUANTITY ERRORs in line 180 helped us find this.

Line 40 should be changed to:

```
40 XC=511:YC=511
```

The *PET Personal Computer Guide* and The *CBM Professional Computer Guide*

Authors: Adam Osborne, Jim & Ellen Strasma; Publisher: Osborne/McGraw Hill
Reviewed by John Stockman

Until recently there has never been one good PET users manual or one good CBM users manual. They've always been somehow mashed together into the same text. But that is no longer a problem now that the *PET/CBM Personal Computer Guide* has been split into two books, each with their own distinctive focus.

Jim and Ellen Strasma are probably two of the most knowledgeable CBM/PET users in the U.S. who have not actually worked for Commodore (and more than many who have). Their work in the *Midnight Software Gazette* and *The Paper* has always been interesting to me.

The two new books share a common structure in the logical flow of topics. Each has ten chapters and three appendices, arranged in the following order:

1. Introduction
2. Operation
3. Screen Editing
4. Programming
5. Making the Most of the Features
6. Peripheral Devices
7. Other Peripherals
8. Inside and Beyond BASIC
9. Upgrades and Fixes
10. Summary of Commands and Errors

Appendices
- A. Character Codes
- B. Memory Maps
- C. Resources

The *PET Personal Computer Guide*

This book is specifically written for use with the Commodore 4032 and 4016 computers, the Commodore 4040 and 2031 disk drives and the Commodore 4022 dot matrix printer. The selection of products to involve in this book is based on the assumption that personal computer users are not going to shell out the extra money for a more expensive disk drive like the 8050 or an even more expensive letter quality printer, which I feel are very valid assumptions. The book also includes information on the Commodore 8010 modem (including sample software) and the 4010 voice synthesizer, neither of which has much documentation floating around about it.

There are two chapters that really impressed me with their thoroughness: chapters 5 and 6, "Making the Most of PET Features" and "Peripheral Devices" (the longest chapters in the text). Chapter 5 includes tips and ideas on utilizing features like the keyboard buffer, numeric strings, prompts, programmed cursor movement, programming graphics, and random numbers. Chapter 6 includes one of the most complete disk programming guides I've seen for Commodore equipment.

The *CBM Professional Computer Guide*

The CBM book specifically describes the Commodore 8032 computer for use with the Commodore 8050 disk drive and Tally 8024 or Commodore 8023P dot matrix printers, as well as the Commodore 8300P letter quality printer. The Commodore 9000 SuperPET is also mentioned as well as the Expansion Memory Board and the forthcoming 8250 disk drive. The Commodore 8010 Modem and 4010 voice synthesizer are both described in this book, but I'm not sure the professional audience will get as much from this section.

Again, I find the "Features" and "Peripherals" sections the most noteworthy. Most of the features are described

here also, and the controls for the 80-column screen editor are finally documented clearly, including inserting and deleting lines, screen scrolling, and windowing. Some good suggestions for data entry routines are explained for those of you who intend to use some sort of data capture and retrieval software. As in the PET book, the disk programming section is quite above reproach.

The "Peripherals" section of the book contains a very good explanation of the features of the printers, including how to format outputs, select special print options and define special characters on some units.

All in all these books are quite complete and a good addition to the technical library of any PET or CBM owner. In my own case, the CBM book alone replaces nearly three binders of handwritten and photocopied notes I've collected over the past 2½ years. But I have to admit that one thing really amazes me about the sample programs in these books. Guess what, *they actually work!!* **C**

new products

The following information is taken from new product announcements sent to us by various manufacturers, and is provided to keep our readers abreast of developments. Commodore does not endorse any of the products mentioned, has not tested them, and cannot vouch for their availability. If you have any problems with any of the products listed here, please write to us.

Company:
Eastern House Software
3239 Linda Drive
Winston-Salem, NC 27106
919-924-2889

Product:
64-RABBIT—a high-speed cassette interface on ROM cartridge that adds twelve commands to BASIC. The new SAVE, LOAD and VERIFY commands operate five times faster than the normal BASIC commands. Other commands include load/run, test memory, decimal to hexidecimal conversion, hex to decimal conversion and more. RABBITS are also available for the VIC 20, and PET/CBM.

Price: $39.95

Product:
MAE (Macro Assembler and Text Editor) for the Commodore 64—a professional development tool, written in machine language, that includes these features: 38 error codes, 27 commands, 26 pseudo ops and 5 conditional assembly operators; built-in software UART (110-9600 baud); macro, conditional assembly and interactive assembly capability; optional creation of executable object code on disk; text editor; word processor.

Price: $99.95

Product:
64 STCP (Standard Terminal Communications Package)—turns a Commodore 64 into a sophisticated smart terminal. Features include: control from BASIC or machine language program; upload/download to or from disk drives (menu driven); outputs to Commodore ASCII printers; automatic modem receive buffer control via standard ASCII XON/XOFF protocol codes; status line display at top of screen; clock that constantly displays time; alarm clock signals on time out; ability to send commands to disk to rename, scratch and others. Includes an ACIA-based hardware interface board that works with your RS-232 modem. STCP is also available for PET computers.

Price: $129.95

Company:
Data Equipment Supply Corp.
8315 Firestone Boulevard
Downey, CA 90241
213-923-9361

Product:
ACC/SYS GA 1600 Accounting System for the Commodore 64—conforms with the Generally Accepted Accounting Principles (GAAP) established primarily by the American Institute of Certified Public Accountants and the Financial Accounting Standards Board. Provides journal, posting, general ledger and automatic financial statements. Works with two 1541 disk drives and 1525 printer or 4040 dual disk drive and CBM printer with IEEE interface.

Price: $395.00

Company:
BrainBank, Inc.
220 Fifth Avenue
New York, NY 10001
212-686-6565

Product:
The Skeletal System—educational software for the PET, age 11 and up. Contains five programs, each covering a different part of the system. This title is the

first follow-up to the popular *The Human Body: An Overview*. Includes extensive documentation with line drawings and a teacher guide.

Price: $70 on disk or cassette
$32 for back-up copies

Product:
Word Functions—educational software for the PET, age 9 and up. Nine separate programs in two parts. Part I includes "Homonyms," "Homonyms Matching Game," "Synonyms" and "Synonym Matching Game." Part II, on a second cassette or disk, includes "Antonyms," "More Antonyms," "Troublesome Words" and a review/test. Contains a game in maze format in which a player must correctly place a missing word in a sentence. Includes documentation and teacher guide.

Price: $99 on disks or cassettes
$37 for each back-up set

Product:
Classes of Nouns—for the PET, age 10 and up. Helps teach recognition of common nouns, proper nouns and special classes of nouns. Includes a review/test, extensive documentation and a teacher guide.

Price: $60 on disk or cassette
$22 for back-up copies

Company:
Continental Software
11223 S. Hindry Ave.

Los Angeles, CA 90045
213-410-3977

Product:

Home Accountant for the Commodore 64—financial management system that can perform a variety of accounting functions, from balancing several checkbooks to monitoring cash flow. Can also print statements.

Price: $74.95

Company:
Distribution Unltd.
P.O. Box 81702
San Diego, CA 92138
619-299-3718

Product:

SECURE encryption kit—manufactured in the United Kingdom by Computer Applied Technology, *SECURE* produces 256 encryptions of single programs at random. Before decrypting programs, it performs a series of checks to insure the system is normal and then continually monitors for improper hardware or software interruptions. Compatible with BASIC composite and machine codes. The unit consists of a leather-walleted key, a cassette or disk and comprehensive manual and registration card.

Price: $100.00

Company:
Brokers Listing System
190 Waverly Road
Scarsdale, NY 10583
914-633-8437

Product:

Commercial Property *Super-Search*—for CBM with dual disk drive. Commercial property real estate brokers and investors can quickly describe and store their income-producing property listings using any combination of features and get fast retrieval at will. Store and search up to 4,000 properties on one diskette. Quickly find features you need, such as ceiling height, floor load, GLA, price per square foot. Can be used by MLS and local real estate boards.

Price: $495.00

Company:
Scholastic, Inc.
730 Broadway
New York, NY 10003
212-506-3000

Product:

Bank Street Writer—for the Commodore 64. Word processing software for the young writer, now being offered to schools through Scholastic, Inc. Developed at the Bank Street College of Education in New York, the package contains three copies of the program disk, each with a step-by-step tutorial, a teacher's manual designed for teachers who may never have used a computer before, and a simple, reproducible student guide.

Price: Contact company

Company:
Computer Marketing Services, Inc.
300 W. Marlton Pike
Cherry Hill, NJ 08002
609-795-9480

Product:

New products for the VIC 20 and Commodore 64—VIC/64 SWITCH connects up to eight VIC 20s or Commodore 64s to share disks and printers. VIC-RELAY cartridge simplifies control of electrical equipment. Contains six relays and two optocouplers. VIC-GRAF cartridge is an aid for studying complicated equations and functions by their graphs. VIC-STAT cartridge is a programmable cartridge consisting of assembler codes to simplify work with statistics and graphics, adding 15 commands to BASIC. VIC-FORTH cartridge is a powerful operating system and programming language suitable for business and process control.

Price: Contact company

Product:

CALC RESULT—three-dimensional spreadsheet for the CBM 8032 and Commodore 64. Provides a minimum of 32 pages of 63 x 254 cells, graphics on screen and printer, window and split screen capabilties and help functions online. Can load VisiCalc files.

Price: Contact company

Company:
Embassy Computer Products
P.O. Box 88
Little Neck, NY 11363

Product:

SIGMA-STAT—statistics program for the VIC 20 with 8K expander. Does simple statistical analyses. Computes descriptive statistics, does linear correlations and regressions and t-tests. Creates data files, saves them, and retrieves them for further analysis or editing. Other features include missing value handling, data transformations and variable name.

Price: $19.95 on cassette only

Company:
Powerbyte Software
2 Chipley Run
West Berlin, NJ 08091

Product:

Business and Home Application Software—for the VIC 20 and Commodore 64. Over 60 applications, including accounting, inventory, business calendar, billing, bar charts, net worth statement, home budget, medical records, diary, recipe file, golf scorecard and word processor. Free catalog available.

Price: From $8.95 to $34.95

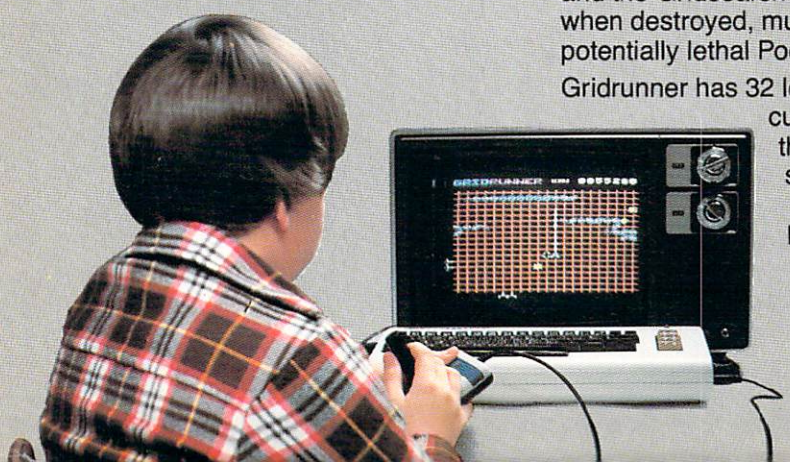
advertisers index

IS GRIDRUNNER™ UNBEATABLE?

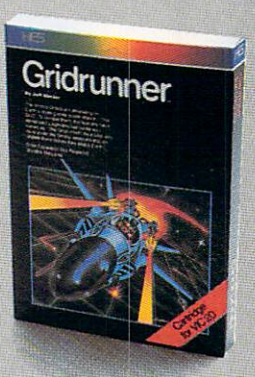
No one, not even the author, has ever achieved the last Gridrunner. It is an extremely fast-paced arcade-quality game designed to test your coolness under fire and challenge your reflexes.

As the pilot of the Gridrunner, a combat ship, you must annihilate the various enemies traveling along the "Grid." High scores are possible only through the mastery of the patterns of the X/Y Zappers and the Gridsearch Droids which, when destroyed, mutate into potentially lethal Pods.

Gridrunner has 32 levels of difficulty (20 levels in the VIC 20 version). To this date, the 13th level has been the highest achieved.



Gridrunner is available for VIC 20™ and Commodore 64™.

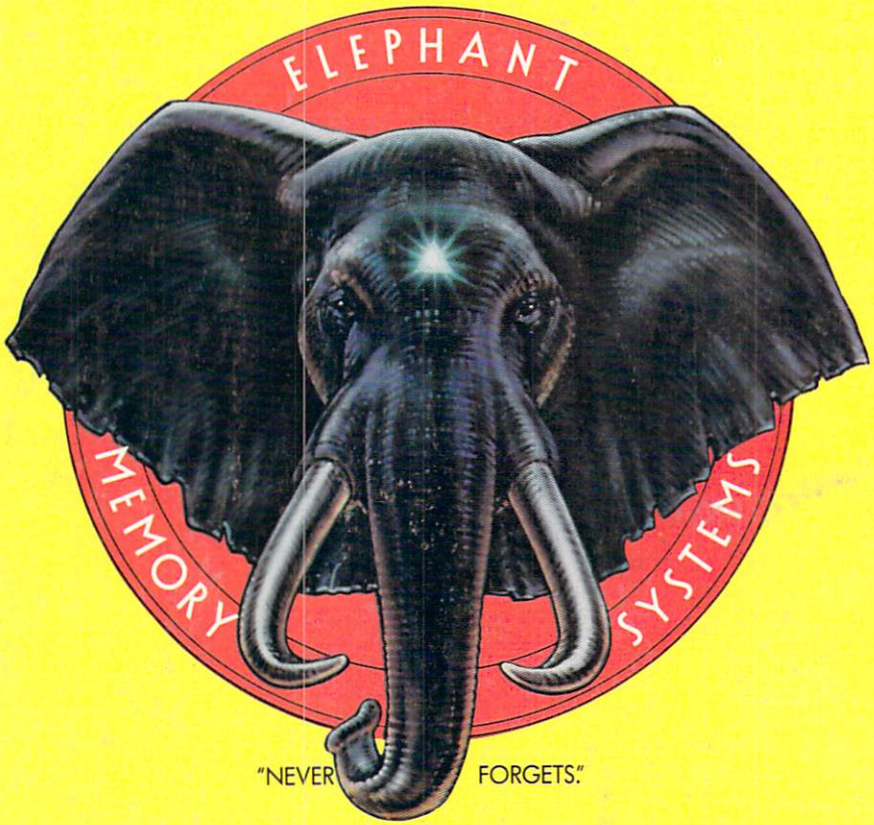


Can you beat Gridrunner? See your local computer or games dealer and find out.

Human Engineered Software
71 Park Lane
Brisbane, CA 94005

HES
a division of USI

REMEMBER:



MORE THAN JUST ANOTHER PRETTY FACE.

Says who? Says ANSI.

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

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

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

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

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

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

ELEPHANT.™ HEAVY DUTY DISKS.

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