$2.95
U.S. FUNDS

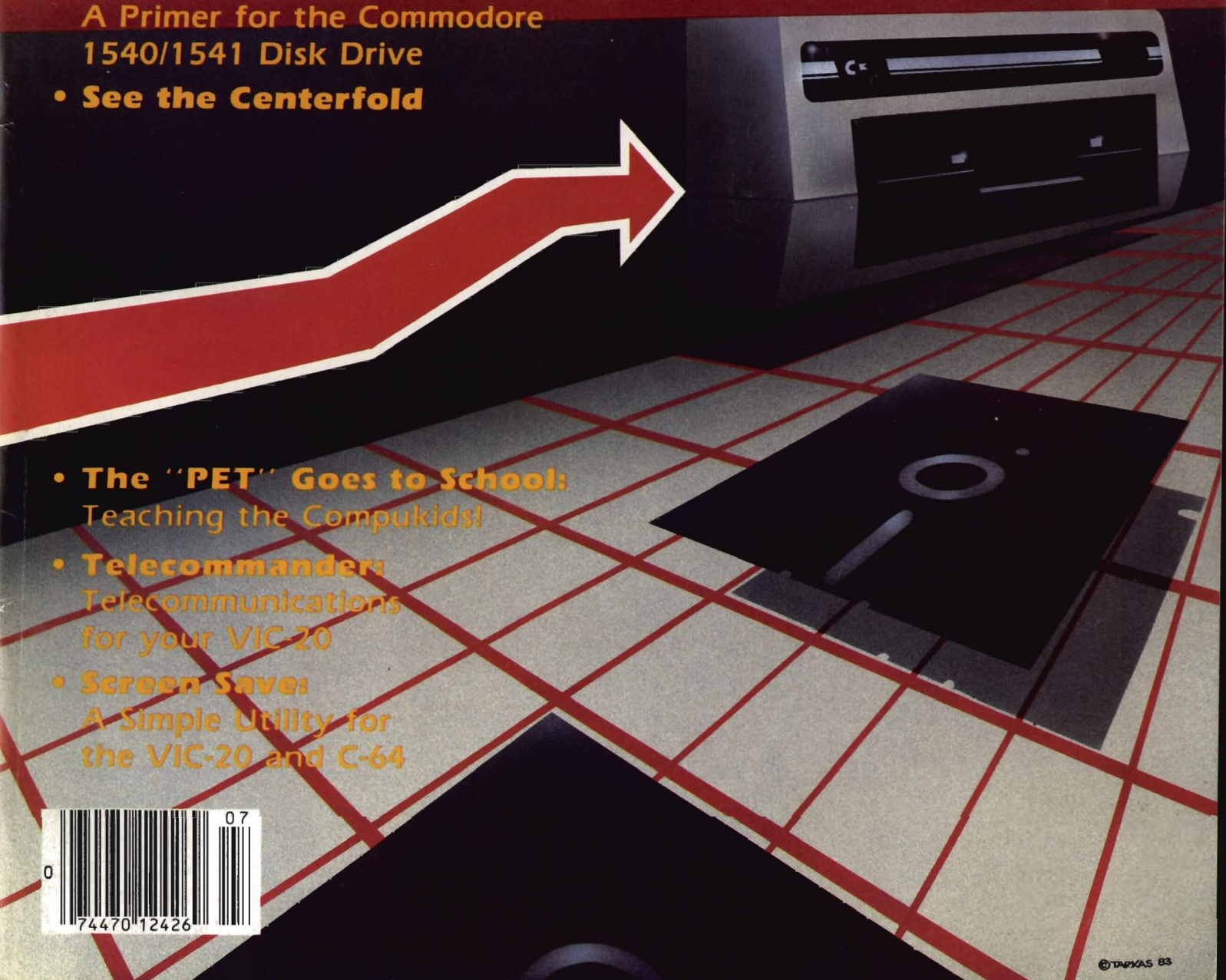
The Monthly Journal for Commodore Computer Users

COMMANDER

JULY 1983 VOL. 1 ISSUE 8

INSIDE:

- **VIC-20 Assembling Language Programming:** Simple Math
- **Revive that Old "Commodore PET" Computer**
- **So You Bought a Disk Drive:** A Primer for the Commodore 1540/1541 Disk Drive
- **See the Centerfold**



- **The "PET" Goes to School:** Teaching the Compukids!
- **Telecommander:** Telecommunications for your VIC-20
- **Screen Save:** A Simple Utility for the VIC-20 and C-64

0 74470 12426 07

Tired of chasing your tail?

If you're looking for Commodore 64® or VIC-20® software, look no further than Synapse!

THE PHARAOH'S CURSE
A fortune — yours for the taking. But can you avoid the ghost of Rama and the evil mummy? Are you nimble enough to leap the chasms and avoid the booby traps standing between you and freedom?

SURVIVOR
The first multi-player cooperative space adventure. Alone or with your hand-picked crew (1 to 4 players) you must out-maneuver the Xenogryph fleet and destroy their space fortresses. Can you train yourself and/or your crew to survive?

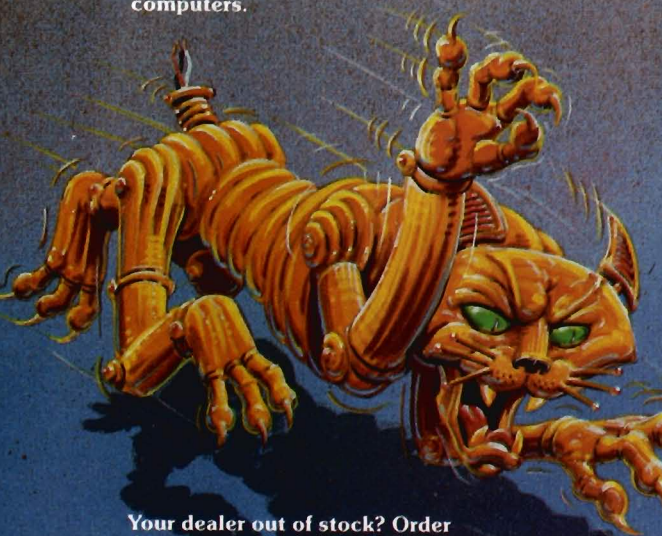
SHAMUS
The odor tells you the Shadow's there — in one of four levels of 32 rooms, each bristling with danger. You know it won't be a high school prom, but there's no turning back. Shamus — the sleuth adventure classic.

FT. APOCALYPSE
Your mission — fly your helicopter to capture fuel and weapons from the Kraalthan lords of inner Earth, free the enslaved masses and destroy the fortress itself. Will you triumph or be crushed by its fiendish defenses? Encounter Fort Apocalypse!

PROTECTOR II
You are the last hope! The Fraxullan Slimehordes are attacking your cities and carrying off their citizens. Can you get your people to safety as volcanoes erupt and enemy forces conspire to thwart your every effort?

For Commodore 64® in disk or cassette: $34.95. Also available — Slam Ball and Drelbs for the Commodore 64®, and Astro Patrol, Squeeze and Harrier for the VIC-20®. Cartridge versions available soon.

Other titles available for the Atari 400/800® computers.



Your dealer out of stock? Order direct — send your check or money order including $2 for shipping and handling. Or join the Synapse Elite! Write for free membership!

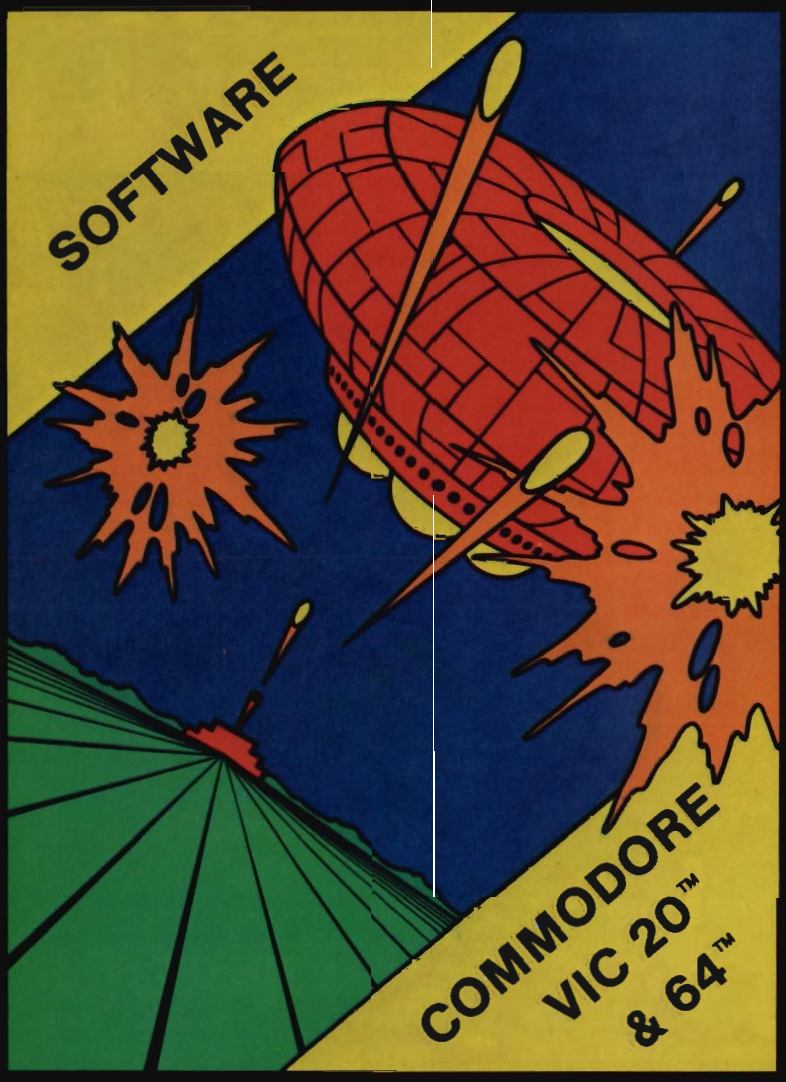
synapse

5221 Central Avenue, Richmond, CA 94804 · (415) 527-7751

COMM * DATA
COMPUTER HOUSE, INC.

ARCADE STYLE GAMES

- Pakacuda*
- Centropods†
- Ape Craze*
- Escape MCP*
- Logger†
- Supercuda†
- Firing Line†
- Pegasus Odyssey†



OTHER GAMES

- Caves of Annod v
- Dragons & Treasure v
- Street Maze*
- The Market*

And Many More

UTILITIES

- Basic Tools†
- Music Mentor†

EDUCATIONAL

- Sketch & Paint*
- Gotcha Math*
- Alphabet Tutor*
- English Invaders*
- Math Invaders Series v
- Tutor Math Series*

And Many More

On Shelves Everywhere - Ask For Comm*Data

Dealer Inquiries Welcome.

† Commodore 64
v VIC 20
* Both

COMM * DATA
COMPUTER HOUSE, INC.

320 Summit Avenue
Milford, Michigan 48042
(313) 685-0113

Arcade Style Games are High Res Full Machine Code.
Commodore 64 and VIC 20 are Registered Trademarks of Commodore Business Machines, Inc.

Circle No. 25

STAFF

Publisher
THOMAS L. ROSENBAUM

Editor-in-Chief
LINDA L. LINDEN

Technical Editor
COLIN THOMPSON

Editorial Assistant
TERILYN M. AICHLMAYR

Dealer/Distribution
DEBBIE A. GABBARD, Manager
PAULA M. ANDERSON, Assistant

Sales/Subscription
ELIZABETH K. STEAN, Manager
CATHY A. SALZER, Assistant

Advertising
LINDA L. LINDEN, Manager
PATRICIA A. ANDERSON, Assistant

Typesetting
GRANGE PRINTING

Consultant
EDWIN SUND
GEORGE R. GAUKEL
JOHN GABBARD

Production Artists
MIKE BRIGHENTI
BRIAN D. THOMPSON
JON WONG

Printed By
GRANGE PRINTING

COMMANDER is published monthly by:
MICRO SYSTEMS SPECIALTIES, PO Box 98827, Tacoma, Washington 98498

Subscription Rates (U.S. Funds)	*Per Year*
U.S.	$22.00
Canadian, Mexican	$26.00
Surface Rates, Foreign	$37.00
Air Mail, Foreign	$54.00

For back issues, subscriptions, change of address or other information, write to:
COMMANDER
PO Box 98827
Tacoma, Washington 98498
(206) 584-6757
BACK ISSUES—
2 months old—$4.50

Commander—The Monthly Journal for Commodore Computer Users is published monthly by Micro Systems Specialties, 3418 South 90, Tacoma, WA 98409. Domestic Subscriptions, 12 issues, $22.00. Second Class Postage pending at Tacoma, WA 98143 and additional mailing offices. Postmaster: Send address changes to **Commander—The Monthly Journal for Commodore Users; PO Box 98827, Tacoma, WA 98498.** Entire Contents copyright © 1983 by Micro Systems Specialties. All Rights Reserved.

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.

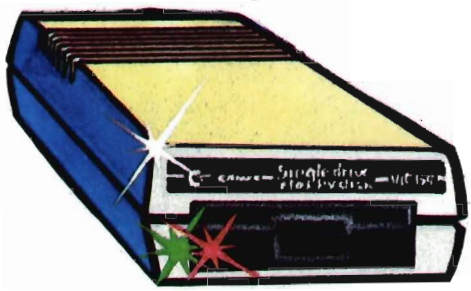
InfoDesigns

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

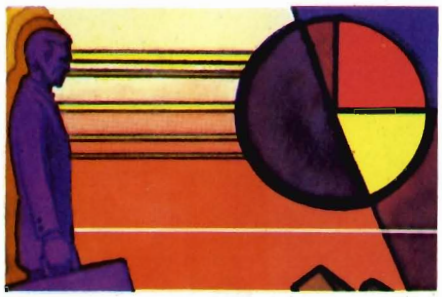
Circle No. 52

Table of Contents

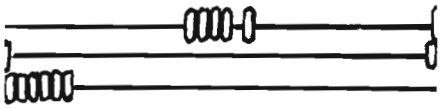
New Products, Page 6



PractiCalc, Page 55



A Disk Drive, Page 40



Programming on the VIC-20, Page 58

COVER BY:
Randy "Tarkas" Hoar

Due to complications, the Basics of Basic, Part 2. will run in the August Issue.

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

OFFICIALLY APPROVED BY COMMODORE

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

AVAILABLE IN CANADA THROUGH
CANADIAN MICRO DISTRIBUTORS, LTD.
500 Steeles Avenue, Milton, Ontairo, Canada L9T 3P7
Telephone 416-878-7277*Telex 06-961243

Circle No. 62

New Products

Rat Hotel Available from Creative Software

VIC-20—

Creative Software announced on June 1, 1983, the release of Rat Hotel, the new game cartridge for the VIC-20.

Rat Hotel is an arcade-style, hide-and-seek game in which the player takes the role of Ermine the Rat, a cheese-seeking inhabitant of the Hotel Paradisio who is pursued by Waldo the Maintenance Man.

Using a joystick, the player maneuvers the Rat from the attic, down six floors and into the basement where he can eat Le Grand Cheeseball. The Rat must reach the basement within a three-minute time period. Reaching the cheese at the very bottom enables the player to get to the next of a total of five difficulty levels in the game.

Tension is created when the Rat encounters the various obstacles in his path. The elevators that allow the Rat to move down each floor will only stop for him if he has eaten all the cheese on the floor he wishes to leave. The cheese, depending on color, will either give the Rat energizing points or leave him paralyzed for a few seconds. In addition, traps are strategically placed in the Rat's favorite dining spots and hiding places. If the player is not careful, the Rat will lose lives by running into as many as three different obstacles: a player may run into Waldo, a trap, or simply run out of time. The Rat is allotted only three lives per game.

Rat Hotel, which is compatible with the VIC-20, has a suggested retail price of $39.95. The new game will be shown at Creative Software's booth, #6580, during Summer CES in Chicago.

Creative Software is the largest independent publisher of VIC-20 software in the United States. The five-year old company is dedicated to publishing a full-line of consumer software programs. Offices are located at 230 East Caribbean Drive, Sunnyvale, CA. 94086, (408) 745-1655.□

Circle No. 72

Creative Software Introduces PIPES

VIC-20/64—

Creative Software introduces Pipes, a new concept home education title, for the VIC-20. Designed and written by John Doering who has written and adapted many successful programs for Creative Software, Pipes is the first in a series of programs intended to join the worlds of education and games. This new educational program plays like a game while teaching the concepts of spatial relationships and economics.

The object of Pipes is to connect all the houses in town to the main water supply. A joystick is used to direct Arlo the Plumber from the factory where he carefully selects the right pipe, to the work-site where he attaches it. Arlo can select elbow-joints, T-joints and valves,

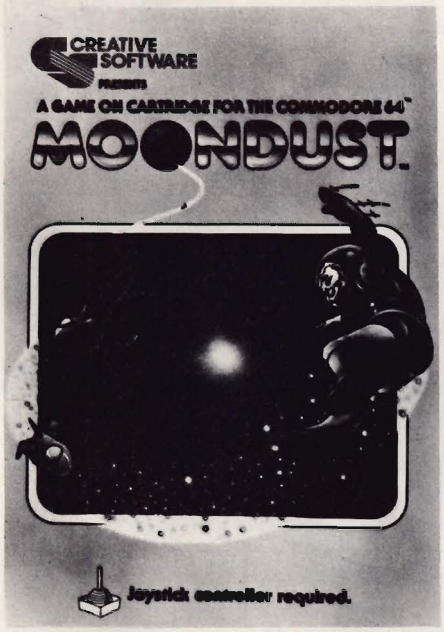
each with differing dollar values and inventory limitations, to create a cost-effective and efficient water network. If the pipes are not connected and sealed properly, leaks will be sprung and the game will end. The challenge of "Pipes" is to connect all the houses without running out of the right kind of pipe or using too much money.

"Although Pipes is geared for teaching children between the ages of 6 and 15, adults can also enjoy the strategic challenge" commented Elliott Dahan, Director of Marketing. "The arcade-style graphics, a trademark of Mr. Doering, combined with the learning aspect of Pipes will provide entertainment in a self-teaching atmosphere."

Pipes is currently available for the VIC-20 and will be available for the

Commodore 64 in August. Suggested retail price for both versions is $39.95. Pipes can be seen at Creative Software's booth, #6580, at Summer CES in Chicago.

Headquarters are located at 230 East Caribbean Drive, Sunnyvale, CA 94086, (408) 745-1655.☐ Circle No. 73



Moondust is Released by Creative Software

64—

Creative Software has introduced Moondust by Jaron Lanier. The announcement was made by Elliott Dahan, Director of Marketing for Creative Software. This new science-fantasy game is available for the Commodore 64.

Moondust is a whimsical game that challenges the player to draw colorful trails of moondust crystals through the heart of glowing concentric circles. It is a true computer game utilizing a full-range of sound and color capability. Music, composed according to joystick movement, accompanies every action.

Moondust features the friendly Spacewalker who is the master painter for the Moondust Fleet. The game begins when he drops moondust to be spread across the screen. Depending upon how far away from the center the moondust is dropped, a certain number of possible points will appear on the screen. The Spacewalker directs his colorful fleet to drag the moondust toward the center. As they whirl around the screen they leave trails of aqua, emerald green, cobalt, coral and purple color-emmissions. The player has to watch for the fleet's flying patterns in order to protect the Spacewalker from being bumped three times and ending the game.

The screen action produces mesmerizing, computer-generated music which reacts to the game play in musical patterns. Jaron Lanier, creator of Moondust, has a classical music background which he applied in the overall game design.

"I wanted to create magnificent color accompanied by exceptional sound," said Lanier. Moondust is a pure reflex game but has the aesthetic elements of music and vibrant color incorporated into a challenging theme."

Moondust was shipped June 1, 1983. It is available for $39.95.

Moondust can be seen at Creative Software's booth, #6580, at Summer CES in Chicago.

Headquarters are located at 230 East Caribbean Drive, Sunnyvale, CA 94086. (408) 745-1655.☐

Circle No. 73

RTTY II Modifies VIC-20 into Terminal

Turn your VIC-20 into a RTTY terminal. Features include split screen operation (compose your reply in a special text buffer while receiving), four 255 character user defined messages and four preset messages, including CQ, RY, time, and CW ID. Select 60,66,75, and 100 wpm BAUDOT speeds, morse code ID, RTTY ID (his call and yours), auto unshift on space—16 different functions and controls in all!

Manual includes instructions on how to modify software for your call and special 'permanent' messages. Hardware manual included with various interface designs (RS-232, TTL, current loop, etc.) as well as info on homebrew and commercial RTTY modulator/demodulators.

VIC RTTY II requires VIC-20 computer with 8K memory expansion, recorder, and VIC-to-Radio interface (RTTY terminal unit and interface) such as the HRA Electronics TU-][for VIC, available from RAK Electronics. (The TU-][is a complete, ready to use terminal unit for VIC MORSE and RTTY. It is available in kit or assembled and tested form. Write RAK Electronics for prices and availability.)

Package includes software on cassette, software and hardware manuals, and I/O edge connector, $19.95 + 2.00 shipping and handling.

Computer catalog of products is available. Specify type of computer.

RAK Electronics, PO Box 1585, Orange Park, FL 32067-1585.☐

Circle No. 75

CW Morse Turns Amateur Radio Station into a Morse Terminal

VIC-20/64/PET—

CW Morse allows your computer to become a morse terminal for your amateur radio station. It is capable of sending and receiving morse code at speeds of 5 to 25 wpm or more. Includes multiple 255 character message buffers, numerous special function keys, type-ahead keyboard buffering, and automatic speed control on receive.

Available for PET 2000/4000 series with 8K or more memory, VIC-20 with 5K memory (increased abilities with optional 3K memory expansion), Commodore-64, Morse requires construction of two transistor, one IC interface. Connection made through the I/O User Port on the VIC-20, C-64, and PET/CBM, Package includes software on cassette, complete documentation, interface schematic, and required connector—$17.95 + $2.00 shipping and handling.

RAK Electronics, P.O. Box 1585, Orange Park, FL 32067-1585.☐

Circle No. 74

V-36 and C-64 Expander Boards

VIC-20/64—

The V-36 is a six-slot expander board for the VIC-20. It features individually switched connectors, a reset button and a fuse.

The C-64 is a four-slot expander board for the CBM-64. It also features individually switched connectors, a reset button and a fuse.

All PTI expanders are based on professional quality fiberglass circuit boards with gold contacts and epoxy soldermask coating on both sides. Sturdy metal feet support the extended portions.

A 90-day warranty from the date of retail sale covers factory repair or replacement of any defective unit. Availability is by direct order or from many Commodore dealers.☐ For more information see ad on page 98.

Circle No. 76

PET Joystick Interface

J Systems Corp. announces the immediate availability of its new PET Joystick Interface. This versatile interface card adds joystick/paddle capabilities to all PET/CBM computers. Device enables the PET to accept inputs directly from 2 Apple joysticks/4 Apple game paddles or 2 Atari joysticks. Interface is complete and ready to plug into the user port. All modes of operation are software-selectable. The device features short access time (less than 10 milliseconds/joystick) and high resolution digitization (greater than 8 bits). This makes the interface ideal, not only for joysticks/paddles, but also, for connecting any four resistive sensors to the PET/CBM. Fast machine language input routines, callable from a BASIC program, are included.

The price of the PET Joystick Interface is $69.95. This price includes the card, power supply, documentation and sample software. VISA and MASTERCARD are welcome. The device can be ordered directly from:

J Systems Corp., 1 Edmund Place, Ann Arbor, MI 48103, (313) 662-4714.☐

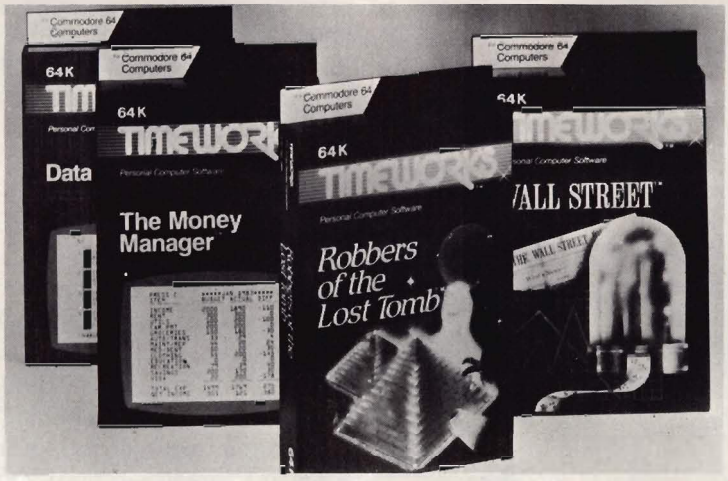
Circle No. 77

Sales/Expense Program

VIC-20/64—

Sales/Expense programs now available for Commodore 64 and VIC-20 computers from RAK Electronics. Sales/Expense is a home or small business program that maintains a full calendar year's sales and expense record for each month by three sales categories and ten expense categories. Totals are calculated for each month for sales and expenses and a total year and average month calculations are provided. Profit is calculated and provided by subtracting the total expense from the total sales for each month and the total year. Data is saved on tape for later use and updating. A print routine is provided and will work with the VIC-1515 and VIC-1525 Printers. Sales/Expense is available on cassette tape and sells for $7.95 plus $2.00 shipping and handling from RAK Electronics, PO Box 1585, Orange Park, FL 32067-1585. Sales/Expense requires an 8k memory expander (minimum) when used on the VIC-20 computer.

RAK ELECTRONICS, PO Box 1585, Orange Park, FL 32073.☐ Circle No. 78



Eleven New Commodore 64 Programs from Timeworks

Eleven new programs for the Commodore 64 are being introduced nationally by Timeworks, Inc., independent publisher of personal computer software. The first four of these programs which "sell on sight," and are now available include Wall Street, a competitive game of financial speculation; Robbers of the Lost Tomb, great adventure search for the Sacred Tablets from the lost 100 room Egyptian tomb; The Money Manager, home and business budget and cash flow system; and the Data Manager, a general information storage and retrieval system with features usually found in much more expensive programs.

Timeworks' new Commodore 64 programs come with complete and comprehensive, yet easy-to-understand manuals; are simple to operate, and are complete with sound effects and color.

Available in both cassette and 5¼" disc, the new Commodore 64 packaging is designed to self-sell at retail. Each carton includes superior dynamic graphics, intriguing descriptions, and program specifications to aid in identification of program parameters.

Suggested retail prices range from $21.95 to $29.95. The balance of the COMMODORE software programs are to be introduced at the rate of four per month.

Timeworks publishes personal computer software in these categories: entertainment, education, programming and home/small business utility.

Get the facts from the Timeworks sales representatives in your area, or communicate with Timeworks, Inc., 405 Lake Cook Road, Building A, Deerfield, IL 60015. For really fast action on these sellers, call (312) 291-9200.☐

Wordprocessing and Interface System for Commodore 64

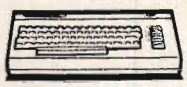
Computer Marketing has announced distribution of two exciting new products from Richvale Telecommunications Corporation for the Commodore 64.

SCRIPT 64 is the first Commodore 64 full-function wordprocessing system with a built-in dictionary, which can be expanded to 20,000 words. In addition to all normal text editing functions, SCRIPT 64 has enhancements such as the ability to hold 40 screen pages (1000 characters each) in the computer at one time, which can be linked for a document capacity of 999 screen pages; storage of deleted text in a buffer for recall in the same or a different place; automatic page numbering, book fashion; headers and trailers; redefinable keyboard; output to video with horizontal (left/right) scrolling; and help screens. SCRIPT 64 supports the Commodore 1541 Disk Drive and Commodore 1525 Printer. Other serial or ASCII printers can be supported with an optional RS-232 interface. In addition, SCRIPT 64 supports all other CBM disk drives and IEEE printers via an optional C-64 LINK! Suggested Retail for SCRIPT 64—$139.95.

The C-64 LINK is much more than just another IEEE Interface. It is smaller than competing units, yet adds more capability. In addition to providing compatibility with all Commodore IEE devices including letter quality printers and large capacity disk drives such as the 8050, 8250 and hard disks, the C-64 LINK adds 15 Basic 4.0 commands, and the ability to chain together eight or more Commodore 64's to time share a disk drive or common IEEE-interfaced printer. Suggested Retail for C-64 LINK—$169.95.

Computer Marketing Services, Inc. is located at 300 W. Marlton Pike, Cherry Hill, NJ 08002. (609) 795-9480.□

Circle No. 80



Parallel Interfaces for Commodore

VIC-20/64—

Micro-Systems Development, (MSD), has announced two new products for its **Interbus Series** of VIC-20™ and Commodore 64™ interfaces.

MSD's **CPI**, a parallel interface for the Commodore 64 and VIC-20, solves several problems for the user such as converting serial-to-parallel, providing ASCII conversion, automatic line feed and program legibility.

The use of the **CPI** interface will allow the user to choose from a variety of available printers and not be limited to only serial printers.

The **CPI** provides two listing modes to the user for program legibility. Since many printers do not support the codes that the VIC-20 and Commodore 64 produce, the **CPI** will generate "tags." For example, [YEL] is generated for "change to yellow." For graphic characters, which are not standard ASCII, the decimal value of the symbol is printed.

The **CPI** is compatible with virtually every program written for use with Commodore's 1515 and 1525 printers. If desired, the **CPI** can automatically send a line feed with each carriage return printed. This satisfies the requirements of some printers that do not have automatic line feed. The **CPI** will, if desired, automatically convert "Commodore ASCII" into standard ASCII. This feature is very desirable. These modes can be selected with either software or hardware control.

The **CPI** connects directly to the VIC-20 or Commodore 64. It works directly through the serial port using the serial port driver software (already built into the computer), and is completely interchangeable between the VIC-20 and Commodore 64.

MSD's **VPI** is a parallel interface for the VIC-20 only. It is lower priced and has fewer features than the **CPI.**

MSD manufactures a variety of expansion products for VIC-20 and the Commodore 64.

For further information, call 1-800-527-5285.

Commodore 64 and VIC-20 are trademarks of Commodore Business Machines, Inc. Interbus is a copyright of Micro-Systems Development, Inc.

Micro Systems Development, Inc., 11105 Shady Trail, Suite 104, Dallas, Texas 75229, 1-800-527-5285.□

Circle No. 81

Parallel Printer Interfaces for VIC20 and CBM64

Tymac Controls Corp. of Franklin, NJ is proud to announce two new Parallel Interfaces for the VIC-20 and CBM64. Our first simple interface is reasonably priced and will provide the essential link between your computer and virtually any Parallel Printer with a standard type Centronics Interface and connector. It simply plugs into the USER Port of the computer and comes with an easy to implement printer driver software listing. It is fully buffered to insure complete protection of the computer. Simple commands make printing easy as pie. This is the least expensive way to add a printer to your computer. Only $19.95.

If you're a VIC-20 owner and would like to take full advantage of graphics and the other special features that the Commodore printer provides, then the Printer Driver Cartridge should be purchased. This cartridge will provide the user with a way to do graphic and standard printing. Available for the Seikosha Printer, C. Itoh Prowriter, Okidata Microline Series and others in the near future. Specify printer when ordering. Only $29.95.

Our second interface is a fully intelligent device that plugs into the disk

(serial) socket just like the standard Commodore Printer. It can easily be assigned any device number and it is daisy-chained to other devices that may be plugged into your VIC20 or CBM64. Using state of the art technology, this parallel interface will display graphically or convert the special symbols to their equivalent representations in clear text. This interface also offers a buffer (almost 2K), a printer reset switch, and a complete self-test mode, Two LED indicators display status and buffer condition. Although this interface is priced higher than our other interface, it will allow virtually total emulation of the Commodore Printer. Nothing has been sacrificed to bring you a superior intelligent parallel board. There are even additional features such as skip over perf, margin set, and set length of line. This interface is printer specific to take full advantage of graphics, special print, and other items that apply individually to your printer. This interface is designed to work with virtually all Centronics Compatible Printers with a standard Centronics connector. Available for the VIC20 and CBM64. Specify your printer when ordering. (Seikosha, C. Itoh, OKI, and others). Only $119.00.

Micro-Ware Distributing, Inc., PO Box 113, Prompton Plains, NJ 07444, (201) 838-9027.☐ Circle No. 82

Computer/Videocassette Machine Connector Interface

64—

The VideoBook Corporation of Seattle, WA, has announced their new product. It is the **Promethius I**, a $49.95 interface module which will connect your Commodore 64 to your Panasonic VCR. The **Promethius I** will fit any of the following models; any of the Panasonic 5000 series (5000, 5200, 5500) or the new 6500 portable models. This new interface module plugs into the back of the Commodore 64 and into the front of the VCR. (Video Cassette Machine)

These new computer Interactive interfaces will allow you to not only have computer educational products but video tape material as well, so for the first time the hardware can be had for under $1,000. Prior to this time the hardware alone has cost $5,000. All this is made possible because of the new interface designed and created by VideoBook's President Jerry Pierson. VideoBook is a courseware (Publishing/Producing) House which makes Interactive Courseware (books, videocassettes and floppy discs) for the Commodore 64. Rumors also are flying about a new $49 (300, 600, 1200, 2400 baud rate) phone modem for the Commodore 64 called the **Hermes I** and a fantastic **new color graphics tablet** for the Commodore 64 called **The Wizard**. The cost to be about $399 but to have unbelievable features, including a discrimination rate of .005. This means **no jawgels** in your graphics—only smooth lines and 128 different colors from the original 16 colors of the Commodore.

Contact: Jerry Pierson, VideoBook Corporation, PO Box 19597, Seattle, WA 98109.☐ Circle No. 84

New for VIC-20— The W20-35

Now VIC-20 owners can have 35K of RAM and IEEE 488 interface on one standard size cartridge without the need for costly expansion chassis. The compactness of this design is achieved by using 8Kx8 CMOS RAMs. These devices are also interchangable with the popular 2764 EPROMs allowing users to mix RAM and EPROM. The IEEE 488 interface allows VIC-20 owners to expand their selection of accessories such as high speed printers, hard disks, floppy disks, etc.

All boards are fully socketed for future expansion and are available in any configuration from bare board to fully populated.

For further information, contact Wave Computers, Inc., PO Box 3883, Federal Way, WA 98003, (206) 839-WAVE or 839-9283.☐ Circle No. 83

The following list of VIC-20 product suppliers was graciously provided by Colin F. Thompson of BASF. It will be published bimonthly, and weekly updates are available. Please send a self-addressed, stamped envelope to the following:

Mr. Colin F. Thompson
BASF Systems Corp.
1307 Colorado Avenue
Santa Monica, CA 90404

Circle No. 85

KEY: SORT	TELEPHONE	NAME	STREET ADDRESS	CITY / STATE / ZIP
1 KEY LISTING----------		h: = HARDWARE MANUFACTURERS	p: = PERIODICALS - NEWSLETTERS	o: = OTHER
2 KEY LISTING----------		s: = SOFTWARE WRITERS	d: = WHOLESALE DISTRIBUTORS	m: = MAIL ORDER HOUSES
3 SOFTWARE TYPES (after ~~)		~~G Games.....~~B Business....	~~E Education...~~U Utility...	~~H Home.....~~R Ham Radio
				vo 06-05
d: . WHOLESALE DISTRIBUTORS				
d: alp	800 438 0676	Alpha Et Cetera, Ltd.	Box 231	Shelby, NC 28150
d: avs	800 638 1688	AVS	7566 Main Street	Sykesville, MD 21784
d: csi	914 425 2800	CSI Distributors	33 Murry Hill Drive	Spring Valley, NY 10977
d: ics	801 373 2901	ICS Micro Wholesale	Box 1243	Provo, UT 84603
d: lip	800 245 6350	B. Lipsitz Co.	450 Melwood Ave.	Pittsburgh, PA 15213
d: par	800 251 5959	B.A. Pargh	1280 Murfreesboro Rd.	Nashville, TN 37217
d: sma	415 964 8201	Small Systems Engineering	1056 Elwell Court	Palo Alto, CA 94303
d: soe	213 412 1700	Softsel	8295 South La Cienega	Inglewood, CA 90301
d: sof	800 828 7250	Software Distribution Services	1280 Main Street	Buffalo, NY 14209
d: sos	415 887 6699	Softsmith	2935 Whipple Road	Union City, CA 94587
h: . HARDWARE				
h: adv	512 441 3202	Advanced Processor Systems	Box 43006	Austin, TX 78745-0001
h: apr~~H	805 484 3604	Apropos Technology	350 N. Lantana Suite 821	Camarillo, CA 93010
h: arf	318 988 2478	Arfon Microelectronics, U.S.	111 Rena Drive	Lafayette, LA 70503
h: baz	206 874 3029	BAZ Electronics	Box 4895	Federal Way, WA 98003
h: bus	413 567 8584	Business Comp. Sys. of New Eng	Box 2285	Springfield, MA 01101
h: byt	206 236 BYTE	Bytesize Micro Technology	Box 21123	Seattle, WA 98111
h: cab	800 343 4311	CAB-TEX, Inc.	Riverside St.	Nashua, NH 03062
h: cai	517 687 7343	CAI Instruments	152 E. Saginaw Road	Sanford, MI 48643
h: car	316 267 6525	Cardco	313 Mathewson	Wichita, KS 67214
h: cen		Century Micro	7881 La Riviera Dr. Suite 131	Sacramento, CA 95826
h: cir		Cir-Kit Engineering	10136 E. 96th St.	Indianapolis, IN 46256
h: com	609 795 9480	Computer Marketing Services	300 W. Marlton Pike Suite 26	Cherry Hill, NJ 08002
h: cos~~EBUG	617 961 5700	Computer Software Associates	50 Teed Drive	Randolph, MA 02368
h: cow	602 249 0611	Computer Works	2028 West Camelback	Phoenix, AZ 85015
h: dat~~B	714 770 2366	Data 20 Corp.	20311 Moulton Parkway Ste. B10	Laguna Hills, CA 92652
h: des~~GBH	213 923 9361	Data Equipment Supplies	8315 Firestone Blvd.	Downey, CA 90241
h: dig	503 295 5898	Digital Interface Systems	Box 8715	Portland, OR 97207
h: dyn	214 542 6012	Dynamic Technologies	Box 351	Allen, TX 75002
h: dyt	813 384 1539	Dytek	Box 241	Pinellas Park, FL 33565
h: ecx	415 944 9277	ECX Computer Co.	2678 North Main Street	Walnut Creek, CA 94526
h: elc~~HG	714 623 8314	Elcomp Publishing	53 Red Rock Lane	Pomona, CA 91766
h: exa	800 538 8559	Exatron	181 Commercial St.	Sunnyvale, CA 94086
h: fer	317 297 0842	Ferris Associates	Box 68421	Indianapolis, IN 46268
h: gen	215 861 0850	Genesis Computer Corp.	1444 Linden St.	Bethlemem, PA 18018
h: glo	617 283 7719	Glouster Computer Bus Co., Inc	6 Brooks Road	Glouster, MA 01930
h: gos	316 265 9992	Gosub Int'l.	501 E. Pawnee Suite 430	Wichita, KS 67211
h: hyp~~UG		Hypertech	1820 NE 142nd St. Penthouse 7	Miami, FL 33181
h: inc	801 226 3809	Interface Computerware	Box 862	Orem, UT 84057
h: ind	312 975 2040	Interdesign, Inc	2054 Irvine Park Road	Chicago, IL 60618
h: int	714 641 0181	Integrated Controls	1240-L Logan Avenue	Costa Mesa, CA 92626
h: kan~~R	913 842 7745	Kantronics	1202 East 23rd	Lawrence, KA 66044
h: lym		Frank Lyman	Box 3091	Nashua, NH 03061
h: met~~U	503 232 1712	Metaresearch, Inc.	1100 SE Woodward	Portland, OR 97202
h: mic		Micro-Star	5633 Emigration Canyon	Salt Lake City, UT 84108
h: mid~~UH	303 934 1973	Micro World Electronix	3333 S. Wadsworth Blvd. #C-105	Lakewood, CO 80227
h: mik	619 569 0900	Microtek, Inc.	9514 Chesapeake Dr.	San Diego, CA 92123
h: mis	303 475 0083	Missing Link Products	Box 6460	Colorado Springs, CO 80934
h: msd~~GBH	214 484 7836	Micro-Systems Development	11105 Shady Trail Suite 103	Dallas, TX 75229
h: mws	301 632 0620	MWS Electronics	Box 418	Pocomoke, MD 21851
h: oem~~UG	305 465 9363	O.E.M. Inc.	2729 So. US # 1 Suite 12	Fort Pierce, FL 33450
h: opa	804 973 5482	Optimal Technology Inc.	Blue Wood 127	Earlysville, VA 22936
h: opt		Optimized Data Systems	Box 595	Placentia, CA 92670
h: oxf		Oxford Computer Systems	Hensington Rd. Woodstock	Oxford OX7 1JR ENGLAND
h: par	415 651 3160	Parsec Research	Drawer 1766-P	Fremont, CA 94538
h: per	312 961 2347	Personal Peripheral Products	Box 3423	Aurora, IL 60505
h: pre	801 487 6266	Precision Technology	2970 South Richard St.	Salt Lake City, UT 84115
h: pro~~UG	303 778 1312	Progressive Peripherals & Sftw	6340 W. Mississippi Ave.	Lakewood, CO 80226
h: ram		RAM/RBC Systems	Box 351	Malden, MA 02148
h: ric	416 884 4165	Richvale Telecommunications	10610 Bayview	Richmond Hill ONT CAN L4C 3N8
h: rvr~~u	315 446 2763	RVR Systems	Box 265	Dewitt, NY 13214
h: sl (send SASE)	313 846 6666	Slagh System Services	Box 53	Dearborn, MI 48121
h: sof		Soft-Aware	Box 725	Glendora, CA 91740
h: sun		Sunshine Peripherals	1229 East 28th Street	Brooklyn, NY 11210
h: sym	313 272 2950	Symtec	15933 W. 8 Mile Road	Detroit, MI 48235
h: ver	714 850 1108	Vertex Int'l.	3941 B So. Bristol #108	Santa Ana, CA 92704
h: vip	714 527 8264	VIP Enterprise	919 N. Cambria St.	Anaheim, CA 92801
h: voi	714 481 7390	Voice World	13055 Via Esperia	Del Mar, CA 92014
o: .OTHER SUPPLIERS				
o: anv	213 575 8614	Anvil Cases	4128 Temple City Blvd.	Rosemead, CA 91770
o: car		Carry Comp	24687 Aric Way	Elkhart, IN 46517
o: cie	714 757 4849	Computer Info Exchange	Box 158	San Luis Rey, CA 92068
o: com	800 848 7548	Computer Case Company	5650 Indian Mound Court	Columbus, OH 43213
o: cop	800 258 7862	Computer Power Int'l.	1779 East Florida	Hemit, CA 92343
o: cre		Creative Computing Catalog	39 E. Hanover Ave,Dept HA1X	Morris Plains, NJ 07950

KEY: SORT	TELEPHONE	NAME	STREET ADDRESS	CITY / STATE / ZIP
o: des		Desert Charts	Box 810	Glendale, AZ 85311
o: edu		Educational Software, Inc.	4565 Cherryvale Ave.	Soquel, CA 95073
o: ele	916 677 8120	Electrosmith	Box 1430	Shingle Springs, CA 95682
o: hym	503 636 6888	Hymac Systems	Box 446	West Linn, OR 97068
o: lor (nat'l VIC user group)		Lords of Basic	Box 459	Ladson, SC 29456
o: mac	619 452 5151	Macro Dynamics	8950 Villa La Jolla Dr. #1200	La Jolla, CA 92037
o: mad	608 255 5552	Madison Computer	1825 Monroe	Madison, WI 53711
o: mic	408 374 4364	Micro Mittens	Box 10246	San Jose, CA 95157
o: nva		National Vic Association	9 Crabapple Lane	Nanuet, NY 10954
o: pmp		PM Products	4455 Torrance Blvd. #177	Torrance, CA 90503
o: sof	615 457 5068	Software To Go	Rt. 3 Box 309 A 52	Clinton, TN 37716
o: sys		Systems Management Associates	3700 Computer Drive	Raleigh, NC 27619
o: tor~~GBEUH	TORPET	Toronto PET Users Group	1912A Avenue Road, Suite 1	Toronto, ONT CAN M5M 4B1
p: . PERIODICALS-MAGAZINES and		NEWSLETTERS ------ key $YEARLY	SUBSCRIPTION- # of ISSUES	
p: coa $18-12	800 426 1830	Commander Magazine	P.O. Box 98827	Tacoma, WA 98498
p: cog $20-12	800 334 0868	The Compute! Gazette	Box 5406	Greensboro, NC 27403
p: com $20-12	800 334 0868	Compute!	Box 5406	Greensboro, NC 27403
p: coo $15-6	215 687 9750	Commodore-Microcomp. Magazine	487 Devon Park Drive	Wayne, PA 19087
p: fox $53-12 cass	713 473 6723	Foxfire Systems, Inc.	3811 Newton	Pasadena, TX 77503
p: jou $12-24	214 482 6679	Journal/20	Box 1149	Van Alstyne, TX 75095
p: loa $50-12 cass	913 762 4730	Load 20 Magazine	550 Grant Ave.	Junction City, KS 66441
p: mic -12	617 256 5515	Micro	Box 6502	Chelmsford, MA 01824
p: mid $20-6	217 864 5320	Midnite/Paper	635 Maple	Mt. Zion, IL 62549
p: nat $10-12	NUGGET$	National VIC-20 Users Group	Box 34575	Omaha, NE 68134
p: pow $10-4	215 345 8112	Power Play	Box 651	Holmes, PA 19043
p: pro $50-12 cass	919 489 2198	Programmer's Institute	Box 3191	Chapel Hill, NC 27514
p: str $15-6		Strictly Commodore	47 Coachwood Place N.W.	Calgary, ALTA CAN T3H 1E1
ps: by~~G $6-12 VIC-NIC NEWS		The Byte House	Box 981	Salem, NH 03079
m: . MAIL ORDER HOUSES				vm 06-05
m: aa8 (cat $1)	313 669 3110	Aardvark-80	2352 S. Commerce	Walled Lake, MI 48088
m: abc	215 822 7727	AB Computers	252 Bethlehem Pike	Colmar, PA 18915
m: all		Allegiance Enterprises	868 96th Ave. N.E.	Blaine, MN 55434
m: ame		American Peripherals	122 Bangor Street	Lindenhurst, NY 11757
m: cib	800 323 4228	Cibcoa	6252 W. Oakton	Morton Grove, IL 60053
m: cma	404 981 5939	C Mart	Box 77286	Atlanta, GA 30357
m: coe	313 528 1554	Computer Express	Box 569	Troy, MI 48099
m: com		Computer Mail Order	EAST (800) 233-8950	WEST (800) 648-3311
m: cos		Computer Specialties	1253 Broadway	El Cajon, CA 92021
m: cot	800 558 8803	Comstar	Box 1730	Goleta, CA 93116
m: cou	316 684 4660	Compu Sense	812 S. Lightner	Wichita, KS 67218
m: cow	516 621 1362	Compuway, Inc.	24 Lumber Road	Roslyn, NY 11576
m: cpm (cat. $1.25)		CPM	Box 19137	Charlotte, NC 28219
m: dis	414 231 1696	Discount Software House	Box 93	Winnebago, WI 54985
m: eas	919 924 2889	Eastern House	3239 Linda Drive	Winston Salem, NC 27106
m: eav		EAV Software	17 Marble Avenue	Pleasantville, NY 10570
m: ekt	415 489 1532	Ektype Office Systems	1655 Whipple Road	Hayward, CA 94544
m: emb	212 961 9806	Embassy Computer Products	Box 88	Little Neck, NY 11363
m: hai	800 645 9187	Harrison Computer Center	2263 Broadhollow Road	E. Farmingdale, NY 11735
m: har	800 221 8927	Harmony Video & Electronics	2357 Coney Island Ave.	Brooklyn, NY 11223
m: jmc		JMC	1025 Industrial Drive	Bensonville, IL 60106-1297
m: lyc	800 233 8760	Lyco Computer	Box 10	Cogan Station, PA 17728
m: mai	800 752 1341	Mail Comp	9434 Chesapeake Drive	San Diego, CA 92123
m: mar	800 331 9131	Marco Polo Co.	4681 S. 83rd E. Ave.	Tulsa, OK 74145
m: mim	800 841 0860	Micro Management Systems, Inc.	2803 Thomasville Road	Cairo, GA 31728
m: mir	800 982 6352	Micro Sense	Box 6273	San Bernardino, CA 92412
m: mis		Microsignal Publications	Box 22	Millwood, NY 10546
m: miv		Micro-VIC-Computers	Box 587	Pittsfield, MA 01202
m: miw	201 838 9027	Micro-Ware Distributors	Box 113	Pompton Plains, NJ 07444
m: moo		Mooseware, Inc.	Box 17868	Irvine, CA 92713
m: mtg	800 343 0854	MTG Technical Sales	281 Needham Street	Newton, MA 02164
m: nat (cat.$1)	800 854 6654	National Computer Products	8338 Center Drive	La Mesa, CA 92041-3791
m: oly	800 421 8045	Olympic Sales	216 South Oxford Avenue	Los Angeles, CA 90004
m: opt	916 621 1090	Optoman Consumer Products	Box 1038	Placerville, CA 95667
m: per	315 478.6800	Personal Computer Systems	Box 1073	Syracuse, NY 13201
m: pr (send SASE)	602 886 1505	Prickly-Pear Software	9822 East Stella Road	Tucson, AZ 85730
m: prg	301 488 7719	Programs International	Moravia Center Industrial Park	Baltimore, MD 21206
m: pri	800 343 1078	P.R.I.C.E.	67 Teed Drive	Randolph, MA 02368
m: pro	312 382 5244	Protecto Enterprizes	Box 550	Barrington, IL 60010
m: prs	800 424 2738	The Program Store	4200 Wisconson Ave. NW	Washington DC 20016
m: pyr	609 386 9363	Pyramid Computerware	278 Warren Street	Edgewater Park, NJ 08010
m: que	800 232 2224	Queue Cat. #11	5 Chapel Hill Drive	Fairfield, CT 06403
m: rde		RDE Services, Games Dept.	3580 Warringham	Waterford, MI 48095
m: roc	303 371 2430	Rocky Mountain Micro, Inc.	10890 E. 47th	Denver, CO 80239
m: sav	800 241 2682	SAVE	1782 Marrietta Blvd. N.W.	Atlanta, GA 30318
m: sjb		SJB Distributors	10520 Plano Road Suite 206	Dallas, TX 75238
m: soc	317 353 4855	Software Clearing House	Box 68756	Indianapolis, IN 46268
m: sof	800 828 2838	The Software Connection	5133 Vista Del Oro	Fair Oaks, CA 95628
m: sou	214 484 7836	Southwest Micro Systems	2554 Southwell	Dallas, TX 75229
m: spa~~BH		Space Shuttle Software	Box 252	Cape Canaveral, FL 32920

KEY: SORT	TELEPHONE	NAME	STREET ADDRESS	CITY / STATE / ZIP
m: sun		Sunrise Electronics	7057 Lompoc Court	Citrus Heights, CA 95610
m: swc	503 257 9464	SW Computers	1125 N.E. 82nd	Portland, OR 97220
m: tel	800 255 2000	Tele Soft, Inc.	Box 3456	Troy, MI 48084
m: tis		Total Information Services	Box 921	Los Alamos, NM 87544
m: ust	619 224 8016	U.S. Technologies	Box 7735	San Diego, CA 92107
m: wds	608 271 6889	Wisconson Discount Sales	2417 W. Badger Rd,	Madison, WI 53713
m: zep	215 387 5266	Zepher Micros	323 S. 43rd St. Suite C	Philadelphia, PA 19104
mh: wor (cat. $1)		World Electronics	177 27th Street	Brooklyn, NY 11232
s: . SOFTWARE				vs 06-05
s: 3gc~~H	503 357 9889	3G Company	Rt. 3, Box 28A	Gaston, OR 97119
s: aba~~U	616 241 5510	Abacus Software	Box 7211	Grand Rapids, MI 94510
s: aca~~EG		Academy Software	Box 9403	San Rafael, CA 94912
s: art~~G	800 828 6573	Artworx Software Company	150 N. Main St.	Fairport, NY 14450
s: ath~~E		Athena Software	727 Swarthmore	Newark, DE 19711
s: ava~~G	301 254 5300	Avalon Hill Games	4517 Hartford Road	Baltimore, MD 21214
s: bas~~B	800 835 2246 x237	Basic Byte, Inc.	Box 924	Southfield, MI 48037-0824
s: bec~~E (cat.$2)	617 536 5116	Boston Educational Computing	78 Dartmouth St.	Boston, MA 02116
s: beh~~E	408 438 5649	Behavioral Engineering	230 Mt. Hermon Rd. #207	Scotts Valley, CA 95066
s: bri~~H	415 455 9139	Briley Software	Box 2913	Livermore, CA 94550-0291
s: bro~~G	415 456 6424	Broderbund Software	1938 Fourth Street	San Rafael, CA 94901
s: cac		Cactus Computing	Box 261	Bouse, AZ 85325
s: cbs~~G		CBS Software (unit of CBS Inc)	41 Madison Ave.	New York, NY 10010
s: cen~~U	215 536 2135	Center Line Mfg.	Box 205	Milford Square, PA 18935
s: coa	306 525 3386	Commercial Data Systems, Ltd.	730 Eastview Ave.	Regina, SAS CAN S4N 0A2
s: cob~~G	408 757 0788	Computer Barn	319 Main St. #2	Salinas, CA 93901
s: cod~~GEU	805 683 1585	The Code Works	Box 550	Goleta, CA 93116
s: com~~E	313 685 0113	Comm¥Data Computer House	P.O. Box 325	Milford, MI 48042
s: cop~~GU	602 855 3357	Computermat	Box 1664	Lake Havasu City, AZ 86403
s: cre~~EGH	415 948 9595	Creative Software	201 San Antonio Circle #270	Mountain View, CA 94040
s: dtc		DTC Software	Box 916	Janesville, WI 53547
s: eag~~UE		David Eagle	Box 982	King of Prussia, PA 19406
s: edu~~E		Edufun/Milliken	1100 Research Blvd.	St. Louis, MO 63132
s: epy~~G		EPYX / Automated Simulations	1043 Kiel Court	Sunnyvale, CA 94086
s: fab~~BH		Fabtronics	51 Quarry St.	Brockport, NY 14420
s: far~~EBUGH	309 382 3191	Farthest Fringe s.a.	101 Highway Blvd.	North Pekin, IL 61554
s: fol~~HUGE		Folklife Terminal Club	Box 2222-M	Mt. Vernon, NY 10551
s: fox~~G	408 988 6666	Fox Video Games, Inc.	4701 Patrick Henry Dr. Bldg 9	Santa Clara, CA 95050
s: fre~~U		French & Silk Smoothware	Box 207	Cannon Falls, MN 55009
s: fsh~~G		Frederick Scheper	8347 Dock Road	Pasadena, MD 21122
s: gal~~BG	408 247 4434	Galactic Software	Box 10516	San Jose, CA 95157
s: gen~~B		General Systems Consulting	2312 Rolling Rock Drive	Conley, GA 30027
s: gla~~H		Martin Glasser	121B Birch Circle	Eglin AFB, FL 32542
s: gpm~~UG		GP Microsystems	72-31 67th Place	Glendale, NY 11385
s: gru~~RU		Jim Grubbs K9EI	Box 3042	Springfield, IL 62708
s: hal		J. Halliday	302 Dogwood Circle	LaFayette, GA 30728
s: har~~G		Harli Software	1740 Garden Briar Court RR#2	Thundar Bay ONT CAN P7C4V1
s: hdm~~B	413 549 3744	H.D. Manufacturing	91 Long Hill Road	Leverett, MA 01054
s: hes~~BEUG	415 468 4110	Human Engineered Software	71 Park Lane	Brisbane, CA 94005
s: ids~~E		Ideal Development Software	652 W 700 N	Clearfield, UT 84015
s: inl	415 459 2905	Intelligent Software	Box 3745	San Rafael, CA 94912
s: int~~G	213 328 9422	Interesting Software	21101 So. Harvard Blvd.	Torrance, CA 90501
s: isa		ISA Software	14114 Dallas Parkway Suite 530	Dallas, TX 75240
s: k8s~~E		K8 Software	Box 248 C	Canton, CT 06019
s: ker~~G		Kerr Software	1390 S. Newton	Denver, CO 80219
s: leo~~U	614 846 1823	Harry F. Leonard	5556 Cherrywood Road	Columbus, OH 43229
s: lig~~E	415 327 3280	Lightning Software	Box 11725	Palo Alto, CA 94306
s: lit~~GU	414 273 5460	Little Wizard Distributing	1211 Lambeth Rd. Suite 4	Waukesha, WI 53186
s: log	805 687 0205	Logos Software	3192 Laurel Canyon Road	Santa Barbara, CA 93015
s: lun~~BG	408 378 7793	Luna Software	Box 26922	San Jose, CA 95159-6922
s: mac~~G		Magic Carpet	Box 35115	Phoenix, AZ 85069
s: mag~~HU	612 559 1108	(M)agreeable software, inc.	5925 Magnolia Lane	Plymouth, MN 55442
s: man~~U	507 345 7048	Mantronics Software Design	360 Pierce Plaza	No. Mankato, MN 56001
s: mar		Mariah Computing	Box 513	Columbia, MO 65207
s: mel		Melbourne House Software	333 East 46th St.	New York, NY 10017
s: mer~~G	213 316 0945	Merlin Enterprises	Box 2876	Torrance, CA 90509
s: mic~~G		Micro-Digital	752 John Glenn Blvd.	Webster, NY 14580
s: mid~~BU		Midwest Micro Associates	Box 6148	Kansas City, MO 64110
s: mie~~E	612 926 2292	Micro-Ed, Inc.	Box 24156	Minneapolis, MN 55424
s: mig~~E	815 965 2464	Microgram	Box 2146	Loves Park, IL 61130
s: min~~E		MicRo Information Systems	Box 73	Wayne, NJ 07470
s: mma		Micro-Mania	Box 4110	Elkhart, IN 46514
s: mph~~E	212 646 0140	Microphys Programs	2048 Ford Street	Brooklyn, NY 11229
s: mpl~~GH		Micro Plus	Box 473	Kenville, NJ 07847
s: mri~~U		Microsignal	900 Embarcadero Del Mar Unit A	Goleta, CA 93117
s: mrp~~GBE	214 867 1333	MicroSpec Ltd.	2905 Ports O'Call Court	Plano, TX 75075
s: mis~~HEG	408 338 9546	MIS	250 Fern Rock Way	Boulder Creek, CA 95006
s: mor		L. Morris	2401 Wengert #26	Las Vegas, NV 89104
s: mos~~E	205 837 3356	Moses Engineering	Box 11038	Huntsville, AL 35805
s: mws~~B		MW Software	Box 126	Urbana, IL 61801

KEY: SORT	TELEPHONE	NAME	STREET ADDRESS	CITY / STATE / ZIP
s: nel~~BG	612 738 1080	Nelson Software	2232 Odgen Court	St. Paul, MN 55119
s: new~~BGEH	604 929 7347	New Horizons Group	#3-119 Charles St.	N. Vancouver, BC CAN V7H 1S1
s: nib~~G		Nibbles + Bits, Inc.	Box 2044	Orcutt, CA 93455
s: nor~~B	218 834 3600	Northland Accounting, Inc.	606 Second Ave.	Two Harbors, MN 55616
s: nuf~~GU	503 878 2113	Nufekop	Box 156	Shady Cove, OR 97539
s: pal~~E		Palos Verdes Learning Center	716 Yarmouth Road #203L	Palos Verdes Estates, CA 90274
s: pam (cat.$1)		M. Pascall Software	Box 1143	Santee, CA 92071
s: par~~G	219 885 0611	Parr Programming	2664 Tyler Street	Gary, IN 46407
s: pas~~G		James Paslay	803 Lucerne Dr.	Spartansburg, SC 29302
s: per~~U	312 961 2347	Personal Peripheral Products	Box 3423	Aurora, IL 60505
s: pms~~B	612 633 0891	PM Software	4400 Arden View Court	St. Paul, MN 55112
s: pow		Powerline Software	110 Woodview Dr.	Horseheads, NY 14845
s: pra~~H		Practical Applications of Cal.	P.O. Box 255768	Sacramento, CA 95825
s: pri(send sase)	602 886 1505	Prickly-Pear Software	9822 East Stella Road	Tucson, AZ 85730
s: pro	301 366 0010	Professional Micro Service	100 West 22nd Street	Baltimore, MD 21218
s: prs~~G		P.R. Software	Box 169	S. San Fransisco, CA 94080
s: pub~~HUGE	513 698 5638	Public Domain	5025 So. Rangeline Road	West Milton, OH 45383
s: qbf~~B	212 925 8290	Quick Brown Fox	548 Broadway Suite 4F	New York, NY 10012
s: qum~~UEG	716 338 2145	Qumax / GRW Laboratories	Box 17010	Rochester, NY 14617
s: rak~~HUGER		RAK Electronics	Box 1585	Orange Park, FL 32073
s: ran~~HGU	904 837 7201	Random Access Computers	Box 1453	Benning, FL 32541
s: rar~~GU		RAR-TECH	Box 761	Rochester, MI 48063
s: ray~~BE	408 338 9848	RAYMAC Software Group	495 Band Road	Boulder Creek, CA 95006
s: rgs~~G(send stamp for cat.)		R.G. Software	417 Susquehanna Avenue	Wyoming, PA 18644
s: rob~~B SASE for cat.		William Robbins	Box 3745	San Rafael, CA 94912
s: sau~~B	907 272 1373	Saura	7510 Foxridge Way	Anchorage, AK 99502
s: sce~~EH		Scientific & Educational Sftw.	Box 54	Dayton, OH 45420
s: sch~~E	212 505 3000	Scholastic, Inc.	730 Broadway	New York, NY 10003
s: sci~~U		Scientivic Software	525 Lohnes Drive	Fairborn, OH 45324
s: scp~~G		Fred Scheper	8347 Dock Rd.	Pasadena, MD 21122
s: sie~~G	209 683 6858	Sierra On-Line, Inc.	Sierra On-Line Building	Coarsegold, CA 93614
s: sim		Simplexsoft, Ltd.	617 N. Property Lane	Marion, IA 52302
s: sir~~G	916 366 1195	Sirius Software	10364 Rockingham Dr.	Sacramento, CA 95827
s: skl~~GU		Skylight Software	22 Miller Street	Belfast, ME 04915
s: sky~~U	415 965 1735	Skyles Electric Works	231E South Whisman Road	Mountain View, CA 94041
s: smi~~G	408 738 1751	D. Smith & Co.	1164 Andover Dr.	Sunnyvale, CA 94087
s: so4~~U		Soft 4 You	Box 3259	Reston, VA 22090
s: soc~~BEUGH		The Software Co-Op	Box 275	Elizabeth, NJ 07207
s: sos		Soft-Sell	Box 1226	LaFayette, GA 30728
s: sou~~E		Southern Solutions	Box P	McKinney, TX 75069
s: sta~~G		Startech, Inc.	13450 Maxella G 185 Suite 200	Marina Del Rey, CA 90291
s: str~~G	813 939 1310	Straley System Software	1868 Grace Ave.	Ft. Meyers, FL 33901
s: sub~~U		Suburban Electronics	6224 Transit Road	DePew, NY 14043
s: sus~~GHB	312 394 5165	Susie Software	Box 380	Prospect Heights, IL 60070
s: sya~~G	415 527 7751	Synapse	5221 Central Ave. #200	Richmond, CA 94804
s: syn~~G	416 221 8008	Syntax Software, Inc.	33 Elmhurst Ave. Suite 502	Willowdale ONT CAN M2N 6G8
s: t&f		T&F Software Co.	10902 Riverside Dr.	North Hollywood, CA 91602
s: tam		Tamerack Software	Water Street	Darby, MT 59829
s: tay~~EUG	402 464 9051	Taylormade Software	8053 East Avon Avenue	Lincoln, NE 68505
s: tea~~E		Teachware	3277-B Roswell Rd. Suite 450	Atlanta, GA 30305
s: tel~EHBG(cat50)	416 263 8064	Telegames Software	Hampton, Box 152	Ontario, Canada L0B 1J0
s: tho~~G	800 526 7843 x291	THORN EMI Video	1370 Avenue of the Americas	New York, NY 10019
s: tot~~B	415 943 7877	TOTL Software	1555 Third Avenue	Walnut Creek, CA 94596
s: trc		Tronic Software	#125-1450 Jonston Rd.	White Rock, BC CAN V4B 5E9
s: tri	714 771 4038	Tri-Micro	58 Plaza Square Suite J	Orange, CA 92666
s: tro~~G	213 671 8440	Tronix Publishing	701 West Manchester Blvd.	Inglewood, CA 90301
s: typ~B		Type Thrift	Shelburne, Ontario	Canada L0N 1S0
s: uhl~~H		UHL Research Associates	7926 Berner St.	Long Beach, CA 90808
s: umi~~GBU	714 594 1351	United Microware Industries	3503-C Temple Ave.	Pomona, CA 91768
s: val		Valley Sof'touch	4974 N. Fresno Suite 132	Fresno, CA 93776
s: vic~~GE	215 576 5625	Victory Software	2027-A SJ Russell Circle	Elkins Park, PA 19117
s: vid	408 263 9858	Video Wizards, Inc.	292 Charcot Avenue	San Jose, CA 95131
s: vix~~U		Victrix & Co.	Box 12232	Boulder, CO 80303
s: voy~~G	415 343.0955	Voyager Software	Box 1126	Burlingame, CA 94010
s: wes~~U		Western New England Software	Box 31	Wilbraham, MA 01095
s: wun~~GE	503 899 7549	Wunderware	Box 1287	Jacksonville, OR 97530
s: zim~~G	213 217 0077	ZiMag	14600 S. Broadway	Gardena, CA 90248

THIS LIST IS PUBLISHED SOLELY TO PROMOTE THE VIC 20 INDUSTRY. IT IS UPDATED TWICE MONTHLY. TO RECEIVE AN UPDATED LIST, SEND A STAMPED SELF-ADDRESSED ENVELOPE TO:

Colin F. Thompson
BASF SYSTEMS CORPORATION
1307 Colorado Avenue
Santa Monica, CA 90404

SAVE

SOUTHERN AUDIO VIDEO ELECTRONICS, INC.
1782 Marietta Blvd., N.W., Atlanta, Georgia 30318

COMMODORE VIC20 $119.00*

PRINTER

DISK DRIVE

DATASSETTE

COMMODORE 64 ALSO AVAILABLE

Commodore VIC20/64 1541 Single Disk Drive	325.00
Commodore VIC20/64 1530 Datassette	62.00
Commodore VIC20/64 1525 Printer	325.00
Commodore VIC20/64 1701 Color Monitor	245.00
Commodore VIC20/64 1600 Vicmodem (Telephone Interface)	91.00

*When purchased with 1530 Datassette and VL110 Reference Programmers Guide.

VIC1110 VIC 8K Memory Expander Cartridge	41.00
VIC1111 VIC 16K Memory Expander Cartridge	65.00
VIC1923 Gorf	28.00
VIC1924 Adventure Land Adventure	28.00
VIC1917 The Count	28.00
VIC1929 Personal Finance	28.00
VIC1211A VIC 20 Super Expander	53.00
VIC1212 Programmers Aid Cartridge	44.00
VIC1213 Vicmon Machine Language Monitor	44.00
VL102 Introduction to Basic Programming-Part I	18.00
VL103 Basic Programming-Part II	18.00
VT107A Home Calculation Program Six Pack	42.00
VT164 Programmable Character Set/ Gamegraphics Editor	11.00
VM100 Personal Computing On The VIC20	4.50
VL110 VIC20 Programmers Reference Guide	13.00

SAVE

Get the best prices on hardware and software. For a complete listing of all SAVE's products, send $5.00 for our catalogue (refundable with your first order).

Enjoy the convenience of in-home shopping. Call our toll free number today for orders only.

Use your American Express, VISA, Mastercard, check or money order. Minimum order of $50. Shipping and handling charges are extra. All prices are subject to change without notice. Allow 2-4 weeks for delivery. Prices good through July 15, 1983.

Order Toll Free 1-800-241-2682
In Georgia (404)-351-8459

News Releases

Thanks for Inquiries on Commodore 64 Software

We are pleased to announce that beginning June 1983, we should have all of our regular line of Software adapted to the Commodore 64 Computer.

The pricing will be the same as for the 8032 and 4032 machines, and will all be disk based. (Designed for the 1541 Disk Drive but can be used with the Commodore 4040 with proper IEEE interface.)

New programs that do not appear on the printed brochure are: **Amortization Program**—$30; and the **Inventory Control Program**—$55.

The description of the programs on the brochure called **An Overview**, also describe the programs for the Commodore 64.

We support and stand behind all Software produced or distributed by Input Systems, Inc.

Contact: Input Systems, Inc., 25101 SW 194th Avenue, Homestead, FL 33031, (305) 245-3141.☐ Circle No. 86

Typro Data Manager and Wordprocessor

PET—

The TYPRO Data Manager and Word Processor Package is comprised of two independent programs on disk that can be used separately, but work excellently together when there is a need to integrate Data Base information with Text.

This set of programs is not intended to compete directly with the WORDPRO's or the QZZ's or the Big Boys, however, this set is offered as a cost effective alternative for the serious computerist who can use this type program for his business applications. With TYPRO he can do so without the big price tag. TYPRO Data Base Manager and Wordprocessor contain many of the features of the Big Boys, yet at only a fraction of the price.

Input Systems suggests that the purchaser **Backup** the TYPRO Master Disk the very first thing. Set the master aside so if a disk is corrupted, you may make another copy.

The Data Manager

The user formats his own fields within the records. Once this has been done it remains permanent in the **Key** file. TYPRO Data Manager will handle records to the maximum limit determined by the capacity of the disk. All Data files are formatted into Relative files, and it is suggested that these files be stored on a separate disk and drive than the Master program file. This allows maximum space on disk for file data.

The sample Data file named **db test** is included on the TYPRO Master Disk for the new user to get the feel of TYPRO Data Base.

Direct screen editing is used, thereby eliminating the necessity of individual **Input** statements for each field of data. If a field within a record is changed, the field is reversed, emphasizing the change. **Two shift returns** are necessary to permanentize the record before going to another record. This allows changes to be made, but not permanentized, unless deliberately done.

The Sort Search feature is used in making labels or mailing lists, and in putting addresses to form letter text. Each field in each record may be searched, and if desired, put into Alpha Numeric Order. A separate file called a Search Data File is created which handles this chore.

Pattern matching is another feature which will enable a user to search a field and only retrieve those which actually match the prescribed pattern. Wild Card Pattern Matching can be used for the matching of only part of a field, searching through the entire Data Base.

You may get hard copy printing of selected fields within a Record when in screen display mode by simply pressing the ampersand (@) key when the cursor is on the line you want printed.

Hard Copy printing of all fields or selected fields can be done to fan-fold cards (such as Rolodex or postcards) through print Menu Selections five and six. These allow you to format to any size fan-fold paper or card. You may select to have each Field Number printed at the beginning of each line, or you may have the Field Name printed in sequence after the field number. This feature can come in handy for auxiliary files such as inventory records. All Fields of each record may be printed, or you may print only selected fields of each record.

Address insertion can be done with Form letter text prepared with TYPRO Word Processor, and again uses the **Search Data** file which has been previously prepared.

TYPRO Word Processor

The Word Processor has two modes for writing text—the **Write Mode** and **Edit Mode**. The 8032 and 4032 will accommodate about 250 average lines of text at 55 characters per line. The append feature extends this to the outer limits. You may append hundreds of files together in the **Print Modes**, thereby enabling the user to assemble a gigantic manuscript, one file at a time.

The Edit Mode features Screen Line Editing, Insert Lines, Delete Lines, Global Edit, (moving blocks of text from one place in the text to another), Forward and Backward scroll.

Text may be right justified.

All text may be **Saved** to disk. (If the same name file is used as previously on disk, then the new **Save** will replace the one on disk.)

Retrieval of text from **Disk** is simplified by the program which reads the Disk Directory and displays the Directory to screen. You may then move the cursor to the desired file, press return and it will be retrieved automatically. (The same feature can be used with **Save** text file.)

You may underscore a line or part of a line by enclosing the part to be underscored with **less than** and **greater than** signs.

Page numbering (at top or bottom) and line titling (top) are available by selection.

All major portions of TYPRO are compiled with Petspeed compiler, and are fast enough for the best typist.

The system prompts you for every command. Anticipated errors are trapped and you will be returned to the prompt or to the Menu. An unbelievable package for the price!

For more information, see ad on page 42.☐ Circle No. 33

Software Protection Devices for Commodores

VIC-20/64—

Softlock Technology has introduced a complete line of software protection devices for Commodore computers. These devices attach to either cassette port on all Commodore computers except the Commodore 64 and VIC 20 on which they connect to the second games port. Each device is custom manufactured to return a code which is unique to the protected program. Programs can periodically check for these codes and be told to malfunction if they are not found. The routine provided to check for these codes is conveniently integrated with Basic, assembly, and Petspeed programs. There are stackable versions of some of the devices so that more than one uniquely protected program can be run on a computer without concern about switching devices between programs. The advantage that device protection offers over "copy-protected" disks is that it allows end-users to make back-up copies of their program disks. The devices are roughly matchbox size and smaller and are simple to attach. There is also a mini device (non-stackable) for the Commodore 64 and VIC 20 for which there are only 20 unique codes. Screen printing of logos, etc. is available for the non-stackable devices except the mini device. The device prices vary from $5.98 to $18.60 per unit depending upon the computer, type, and quantity desired.

Call or write for further details and complete pricing information.

Softlock Technology, 13031 San Antonio Drive, Norwalk, CA 90650, (213) 868-7820.☐ Circle No. 87

Hypnotist II— Latest from Psycom

64—

Patrick Williams, president of Psycom Software International Corporation announced today that their latest product for the Commodore 64, The Hypnotist II, would be available for shipment May 23. The suggested retail price will be $59.90 including the $19.95 Biofeedback device.

The Hypnotist II is the latest in the series of programs by Psycom Software International aimed at the mature market place. The Hypnotist II contains five major program modules:

1. Introduction and Vital Data Gathering.
2. Relaxation and Stress Reduction through Biofeedback.
3. Trance Induction.
4. Neural Reprogramming and Suggestion.
5. Return to Consciousness.

The second of these, Relaxation and Stress Reduction through Biofeedback, will require the use of the PSI Biofeedback device; and will be available for purchase separately for anyone who is not interested in being hypnotized.

The Biofeedback device will be used in still another program soon to be available from PSI, called The Lie Detector. The Lie Detector will operate with the Heartbeat Trance and Digital, Pulse Read-Out displays that are used in the Hynotist II. Hardcopy output will utilize the new pen plotters when they are available.

Still another program in this series is called Super-Learning. Super Learning should be popular with anyone trying a last minute cram for a test or important presentation.

Psycom Software International is a software development and manufacturing corporation located in Cincinnati, Ohio. Psycom specializes in interactive software dealing with psychological self-discovery, awareness, decision making, biofeedback and parapsychological subjects, all aimed at the adult marketplace. Inquiries from distributors, dealers, individuals and organizations interested in our work are warmly welcomed.

For more information, see ad on page 118.☐ Circle No. 35

by Donald L. Stoner
Mercer Island, WA

This month I'd like to present a "they said it couldn't be done" program. How would you like to be able to download programs to your cassette recorder with an unexpanded VIC (3583 bytes)? Well, you can, and the accompanying programs accomplish this miraculous task.

What is Downloading? Shortly after you acquire a modem, you will discover bulletin boards, The Source and CompuServe. Many of these data bases hold free programs that will run on your VIC-20. They also contain text information, such as electronic mail, mailing lists, or instructions and documents that you may wish to retrieve. The trick is to transfer the information from the memory of the host computer to the memory of your computer. This is called downloading.

The downloading of programs and text involves opening a buffer in your computer. The incoming characters are then placed in this storage area. When the operator wants to terminate the download, the buffer storage is turned off and the material saved on a cassette or disk drive.

Unfortunately, the unexpanded VIC-20 has barely sufficient memory for a decent terminal program. How can you store a 2-3K program in your memory when it is already occupied by the terminal program? The answer is simple. You cannot do it.

However, a genius VIC programmer named Terry Imler figured out a crafty way to have your cake and eat it too. His program allows you to download with nothing more than the memory available on an unexpanded VIC. He placed the program on CompuServe's public access program storage for others to use and enjoy. This is where I discovered Terry's program. I downloaded it to my printer, then entered the program on my VIC-20.

Essentially what Terry does is to open a small buffer that just fits within the available VIC memory. As soon as the incoming data fills the buffer, the program sends a character (Control S) to the host to pause the transmission of the material being downloaded. Then the program turns on the cassette recorder and dumps the buffer contents. As soon as the buffer is empty, another signal (Control Q) is sent to the host and the transmission resumes, once again filling the buffer. As soon as the downloading is complete, the program closes the cassette file. Using this technique, programs of virtually any length can be downloaded.

In most cases, the material to be downloaded is stored in the ASCII code. This presents no problem if you are downloading text since you can read the ASCII characters on your screen.

Token Conversion

BASIC programs, however, are stored in your computer as **tokens** to minimize the amount of memory required. For example, the command PRINT does not require five bytes of memory. Rather, it is tokenized and stored as a single byte (a decimal 153). When the VIC operating system brings the byte in from the program in memory, the computer knows the 153 token really means PRINT. This is what's shown on the screen when the program is listed, not a 153. If the program is running, the token prints the information following the 153 rather than appending a 153 to it.

What all this means is that the downloading program stored on the cassette recorder is not in the correct form to run properly on the VIC-20. The purpose of the second program. (Program 2) is to convert the ASCII characters stored on the cassette into the token characters required by the VIC operating system. Enter this program as shown, then save and verify several copies:

SAVE"TOKEN":SAVE"TOKEN"

The program also allows the user the discard unwanted lines. For example, when downloading, extraneous line feeds often accompany the wanted material. Also, it is difficult to open and close the buffer at exactly the right instant. As a result, you may set a **please close your buffer**, or similar message, tacked onto the end of the wanted program. Obviously, this is not going to load into the computer without a line number and would cause a syntax error even if it did. The tokenizing program in Program 2 allows you to edit out unwanted information. More about this later.

How It Works

First, let's look at the downloader program (Program 1) and see how it functions. Line 100 opens the modem file and sets the parameters to 300 baud, a 7-bit word, even parity and

Technological Innovations for the VIC 20™ and Commodore 64™

Designed by RAK-Ware

EXPAND-O-RAM VIC20

EXPAND-O-RAM is truly a multifunction memory expansion board that provides 2 switch selectable slots to double the capacity of your computer. In addition to these important features, a RESET button has been provided to allow you to regain control of a "hung" or unstoppable program. A bank of mini-switches control memory addressing on the board so that you can use EXPAND-O-RAM as a useful tool for cartridge development and debugging. A write protect function is even provided so that you can simulate a ROM environment or investigate a previously programmed ROM Cartridge Pack. A quick summary of the features are as follows:

- **A) 16K of additional Memory.**
- **B) Compatability with additional Memory & Games Cartridges.**
- **C) Two Additional Switch Selectable Slots.**
- **D) Reset Switch for warm starts.**
- **E) Slide Switch for easy selection of Cartridges or memory packs.**
- **F) Switch selectable Memory Map selections of the 16K memory.**
- **G) Use as a Program and Cartridge development tool.**
- **H) Write protection of memory area for cartridge simulation and investigation.**

ALL THIS FOR ONLY.....$119.00

UNIVERSAL TAPE INTERFACE

No need to purchase an expensive data recorder for your VIC 20™ or Commodore 64™. The Tymac Universal Tape Interface and Duplicator will allow you to load, save and even duplicate* your programs and data cassettes with virtually any portable home tape recorder. To insure positive LOADS and SAVES, a special audio enhancement circuit is provided in the interface. Three LED indicators monitor the status at all times and provide a visual indication of loads and saves. A parity switch will also insure that all types of data tapes can be successfully loaded. Operator controlled action of the recorder is another desirable feature. Finally, you can easily make back-up tapes without the need of loading your software back into the computer. A unique switch function will allow you to make direct recordings to another tape recorder. A great duplication device for the user who may want to start a software business at home. The TYMAC UNIVERSAL TAPE INTERFACE AND DUPLICATOR is way ahead of the competitors in features and quality.

ALL THIS FOR ONLY.....$49.00

*NOTE: Duplication requires the use of Two Standard tape recorders.

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

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.

Circle No. 56

one stop bit. The next line dimensions the input and output buffers along with a small buffer for pointer storage.

Line 120 defines strings for the cursor and assigns numbers to several variables. The lines between 200 and 290 set up the ASCII to Commodore conversion tables. Line 250 allows the F1-F4 keys to transmit Control C, P, Q and S respectively.

The end of line 290 opens a cassette file for the storage of the material to be downloaded. For lack of a better name, it is called **COMP**. You must place a cassette into your recorder before running the downloader program. This is so the downloader can write the file open information before proceeding to the terminal section starting at line 300.

Line 300 clears the screen while line 310 looks to see if the keyboard buffer is empty. Line 320 examines the modem for a character. If a character is present, the program jumps to line 330 and prints it on the screen. Assuming no character is present, the program passes to line 325. This line is active if the buffer is open and controls the on/off codes sent to the host computer.

Line 360 examines the keyboard to see if a key is pressed. If it is, the character is sent out the modem by line 370. Line 376 looks to see if the buffer open (F7) or buffer close (F8) key is pressed. The value of M determines if the buffer is open or closed. Line 380 is part of the cursor blink routine. Each time the program loops through this line it increments CT by one. If the total equals eight in line 390, the cursor is reversed and appears to blink. Line 400 loops the modem I/O routine back to the beginning.

Lines 420-690 are part of the cassette write routine. Line 420 writes the headers. Lines 600 and 610 set and reset the pointers while the DATA statements in line 690 provide the memory locations for these pointers.

Using the Program

After you have entered the program as shown in Program 1, be sure to save and verify it. Then replace the cassette with a fresh one and run the program. Log onto a system or BBS that recognizes the Control Q and S, X-on/X-off handshaking protocol. CompuServe is a good system on which to test the program. Their routines are constantly checking for the handshaking protocol.

As soon as you find something you wish to download, press the F6 key (shift F2). The buffer will open to accept characters. As soon as it fills, you will see the screen suddenly stop printing and the motor on the cassette will run. When you have captured the desired information, press the F8 key (shift F4). The cassette will again run and close the file, then the handshaking will tell the host to proceed.

After you log off the system, remove the data cassette and load the token conversion program (Program 2). Make certain you have saved a copy before running the tokenizer program. It erases itself from the VIC memory after it completes the conversion. By the way, I'd love to tell you how this program works. However, after line 60100, I'm lost!

After the program is loaded, replace the cassette with the original cassette of data downloaded from the host. Run the program, which will open the original COMP file you recorded. All material up to a carriage return will be displayed on the screen and you will be asked if you want to keep the line or skip (discard) it. If you see a line that seems to end in the wrong place, or one that does not have a line number, write down exactly what you see. This unusual display means a line feed crept into the middle of a line of BASIC. Every line of BASIC must start with a line number. If it does not, the program will not load and/or run. You should discard defective lines (or parts of lines), then reenter them to splice the line together. You'll have an opportunity to do this after the program is converted to tokens.

Once you have edited the material, the program will go through some incredible gyrations caused by lines 60100 to 60200. Finally, if all goes well, you should see the READY prompt appear. Now, if you type list, the program should float up the screen just like normal. If it was necessary to discard a defective line, it should be reentered at this time. It is a good idea to save the program on yet another cassette before you run it. It is possible that a lock up could occur for unexplained reasons.

Now, when you run the program, it should do whatever it was designed to do. If it does not, list the program section by section until the problem is uncovered.

The preceeding instructions are related to downloading BASIC programs. Downloadng and reviewing text files is considerably less complicated. For ASCII files, open and close the buffer exactly as before. After logging off, load the program shown in Program 3. This will bring in the file (either text of BASIC) and display it on the screen. The display will continue to scroll until you press a key. This will pause the display so you can read it.

Next month, I'll present another of Terry Imler's programs. This one will allow you to upload programs to a host computer or a friend with another VIC.

In the meantime, if you have any telecommunications programs, tips or information which you would like to share with other readers, be sure to send it to Donald L. Stoner, c/o Commander Magazine, PO Box 98827, Tacoma, WA 98498. If there are any radio amateurs out there in readerland, let me know if you are interested in ham radio applications for the VIC. My call is W6TNS/7 and I use the VIC-20 extensively for RTTY communications. □

Program 1

```
100    OPEN5,2,3,CHR$(38)+CHR$(224)
110    DIMF%(255),T%(255),X%(6):R=198
120    K$="  "+CHR$(157):J$=K$+CHR$(146):
M=0:T=166:L=190:Y=212:E=34:D=0
200    FORJ=32TO64:T%(J)=J:NEXT
210    T%(13)=13:T%(20)=8:RV=18:CT=0
220    FORJ=65TO90:K=J+32:T%(J)=K:NEXT
```

```
230   FORJ=91TO95:TX(J)=J:NEXT
240   FORJ=193TO218:K=J-128:TX(J)=K:NEXT
250   TX(133)=3:TX(134)=16:TX(135)=17:TX(136)=19
260   FORJ=32TO95:FX(J)=J:NEXT:FORJ=97TO122:FX(J)=J-32:NEXT
270   FORJ=123TO126:FX(J)=J-64:NEXT
280   FX(96)=32:FX(92)=17:FX(127)=20:
FX(8)=20:FX(13)=13:FX(10)=17
290   FORJ=0TO127:FX(J+128)=FX(J):NEXT:OPEN1,1,2,"COMP"
300   PRINT" "CHR$(147)
310   IFPEEK(R)<>0THEN360
320   GET#5,A$:IFA$<>""THEN330
325   IFM=2THENM=1:GOSUB610:PRINT#5,CHR$(17);:GOTO310
327   GOTO360
330   A=ASC(A$):PRINTK$;CHR$(FX(A));:IFA=ETHENPOKEY,D
340   IFMTHENPRINT#1,CHR$(FX(A));:IFPEEK(T)=>LTHEN500
350   GOTO310
360   PRINTCHR$(RV);J$;:GETA$:IFA$=""THEN380
370   A=ASC(A$):IFA<137THENPRINT#5,CHR$(TX(A));:GOTO380
376   IFA=140THEN410
378   IFA=139THENM=1
380   CT=CT+1
390   IFCT=8THENCT=0:RV=164-RV
400   GOTO310
410   IFM=0THEN380
420   GOSUB550:PRINT#1,CHR$(60);CHR$(3);CHR$(252);CHR$(3);
430   I=PEEK(T):PRINT#1,CHR$(10);:IFPEEK(T)>ITHEN430
440   GOSUB610:PRINT#5,CHR$(17);:M=0:GOTO380
500   IFM<>2THENGOSUB550
510   PRINT#1,CHR$(10);:IFPEEK(T)=>LTHEN510
520   M=2:GOTO310
550   PRINT#5,CHR$(19);
560   J=PEEK(667):FORI=0TO500:NEXT:IFPEEK(667)<>JTHEN560
570   GOSUB600:RETURN
600   RESTORE:FORI=0TO6:READA:XX(I)=PEEK(A):NEXT:RETURN
610   RESTORE:FORI=0TO6:READA:POKEA,XX(I):NEXT:RETURN
690   DATA167,168,169,170,180,181,182
```

Program 2

```
60000   OPEN1,1,0,"COMP"
60010   B$=CHR$(60):C$=CHR$(3):POKE152,1:B=0:GOSUB60190
60020   GET#1,A$:IFA$=""THEN60020
60030   IFA$=C$ANDD$=B$THENPRINT"END OF FILE":PRINT"HIT RETURN
TO FINISH":GOTO60120
60040   IFA$=CHR$(13)ANDB=0THENPRINTCHR$(19);:GOTO60020
60050   PRINTA$;:B=B+1:D$=A$:IFA$=CHR$(34)THENPOKE212,0
60060   IFA$<>CHR$(13)THEN60020
60070
PRINTCHR$(5);"GOTO60010";CHR$(31):PRINT:PRINT:E$="":POKE198,0
60080   PRINT"RETURN=KEEP LINE      S=SKIP LINE":B=0
60090   GETE$:IFE$=""THEN60090
60100   IFE$="S"THEN60010
60110   GOTO60200
60120   IFPEEK(197)<>15THEN60120
60130   A=60000:A$="60140":GOTO60170
```

PROGRAM 2—continued from page 21

```
60140    A=60060:A$="60150":GOTO60170
60150    A=60100:A$="60160":GOTO60170
60160    A=60160:GOTO60170
60170
GOSUB60190:FORI=ATOA+50STEP10:PRINTCHR$(157);I;CHR$(141):NEXT
60180    PRINT"GOTO";A$:GOTO60200
60190
PRINTCHR$(147);CHR$(17);CHR$(17);CHR$(17);CHR$(17);:RETURN
60200
FORI=631TO640:POKEI,13:NEXT:POKE198,10:PRINTCHR$(19);:END
```

Program 3

```
64    PRINT"SEARCHING FOR FILE":OPEN1
65    K=197:M=212:N=0:P$=CHR$(3):Q$=CHR$(10):J=64:R$=CHR$(34)
70    IFPEEK(K)=JTHEN70
80    IFPEEK(K)<>JTHEN80
90    GET#1,A$:PRINTA$;:IFA$=R$THENPOKEM,N
100    IFA$=P$THEN200
110    IFA$=Q$THENPRINT:PRINT"END OF THIS SECTION.  HIT ANY KEY
TO CONT.":GOTO70
120    GOTO80
200    PRINT:PRINT"END OF FILE:CLOSE1:STOP
```

Am I Prepared for Processing?

by Barbara Naness
Staten Island, NY

It isn't easy saying goodby to an old friend. The time and place are never right. Parting words seem shallow and insignificant. Motive and rationale wither and wane as you wrestle with guilt and uncertainty.

Can I, with clear conscience and no qualms or regrets, trade in a perfectly good electric typewriter for a word processor?

Webster's defines typewriter as "a writing machine with a keyboard for reproducing letters, figures, etc. that resemble printed type."

How impersonal. How antiseptic, insensitive and insufficient. Had Mr. Webster ever had the pleasure of spending a week with me and my typewriter, he would certainly have described it as the living, breathing organism that it is.

Amid the cozy clutter of coffee cups, reference books and unfinished drafts, my faithful friend and constant companion sits patiently awaiting our daily sojourn. Like a stoic and stalwart soldier, it bolts to life and attention as I approach.

Unwavering, uncomplaining and unrelentless in strength and stamina, this mighty machine gives substance to my thoughts and sanity to my senseless and scatterbrained babbling. Its steady hum (of contentment, no doubt) soothes and inspires as my fingers dance across the smooth and familiar keys. Jumbled ideas and rambling and elusive concepts become pearls of wisdom and wit. It is an extension of my mind, an analyst, editor, interpreter and translator—nothing short of a magician.

I feed my typewriter a couple of cassettes a month. It's oiled and dusted and carted off to the Type-Righter for semi-annual checkups. Preventative medicine, I have found, is the key to a healthy, happy and productive machine.

My typewriter has been good to me in return. It has read my mind, recorded ingenious ideas and half-baked brainstorms and journeyed with me through fantasylands of imagination. Together we've weathered severe and debilitating cases of writer's block, raced against deadlines and agonized over usage and spelling.

I haven't always been easy to live with, but my typewriter's never complained. I've pummeled it, cursed it and blamed it mercilessly and unjustly. I've driven it beyond exhaustion into the wee hours, then neglected it for days on end. I've cried on it, spilled on it and yes, even pulled the plug on it. But never has this pillar of strength and stability let me down.

Now like the infamous ice age, growing, spreading, blanketing everything in its path, the computer age is upon us. Digital watches, pocket calculators and video games have embedded themselves in our culture. The secrets of the universe are stored on chips. Our lives are laid bare and our future programmed for us through the mesmerizing display of electrons on cathode ray tubes.

Typewriters, we are told, are archaic, inadequate and obsolete. Word processors boast features that the typewriter, in its prehistoric ignorance and infancy, can never hope to compete with. These electronic geniuses have powers and abilities far beyond those of mortal machines. We can now work less, produce more and perform mind-boggling, literary miracles.

Programs and disks replace file cabinets and all manner of paperwork. RAMS and ROMS, the remarkable memory components, summon up and spew forth anything they've ever been fed. With the flick of a switch and the push of a button, artificial brainwaves take over.

Rough copies, those fresh first drafts of creativity, can be changed, rearranged and corrected right before your eyes—without the tedious, time-consuming use of dictionary, thesaurus or human intellect. Spelling is corrected, sentences deleted and paragraphs transposed while you sit idly by. With lightning speed and accuracy, a printer rearranges and records what's been typed onto the screen, eliminating all personal contact.

Global search, word wrapping and interrobanging become everyday activities, replacing proofreading, planning and polishing. It's a whole new world.

It all sounds . . . well . . . oddly enticing. Scary, in a seductive sort of way. Progressive, yet somehow cold and calculating.

The naturally inquisitive mind of a writer wonders at the sacrifices, problems and drawbacks to owning a machine that is smarter than I am.

What if, heaven forbid, it doesn't like what I've written? Will it, in its infinite wisdom, refuse to print my less than perfect prose? My typewriter would never pass judgment or take matters into its own keys. But if I've programmed this omnipotent creature to revise, retouch and reconstruct, who am I to intervene in its hallowed and divine decisions?

The disk drive can become

hungry—ravenously so, I am told—devouring and digesting software, floppy disks and valuable information (without reason, warning or use of ketchup). Although it's rare, this culinary calamity has been known to occur.

Suppose I pour out my heart and soul at the keyboard for months—brilliance and beauty unbridled, if I do say so myself. My Pulitzer Prize-winning novel stares back at me in all its dazzling iridescence, corrected to perfection, waiting to be immortalized. Dare I push the button? It couldn't possibly erase when it's supposed to print, could it? How silly of me to fear such an implausible error. But could it? Alas, I've been warned that it could. Naturally, without premeditated malice or a forethought (so they say).

Computers don't make mistakes very often. But when they do, be prepared for a whopper.

Am I ready to face the consequences of such untimely and unfortunate acts? Consequences that could rival those incurred by the stock market crash? At least when a manuscript is typed on paper, it can be stored safely without fear of instant eradication.

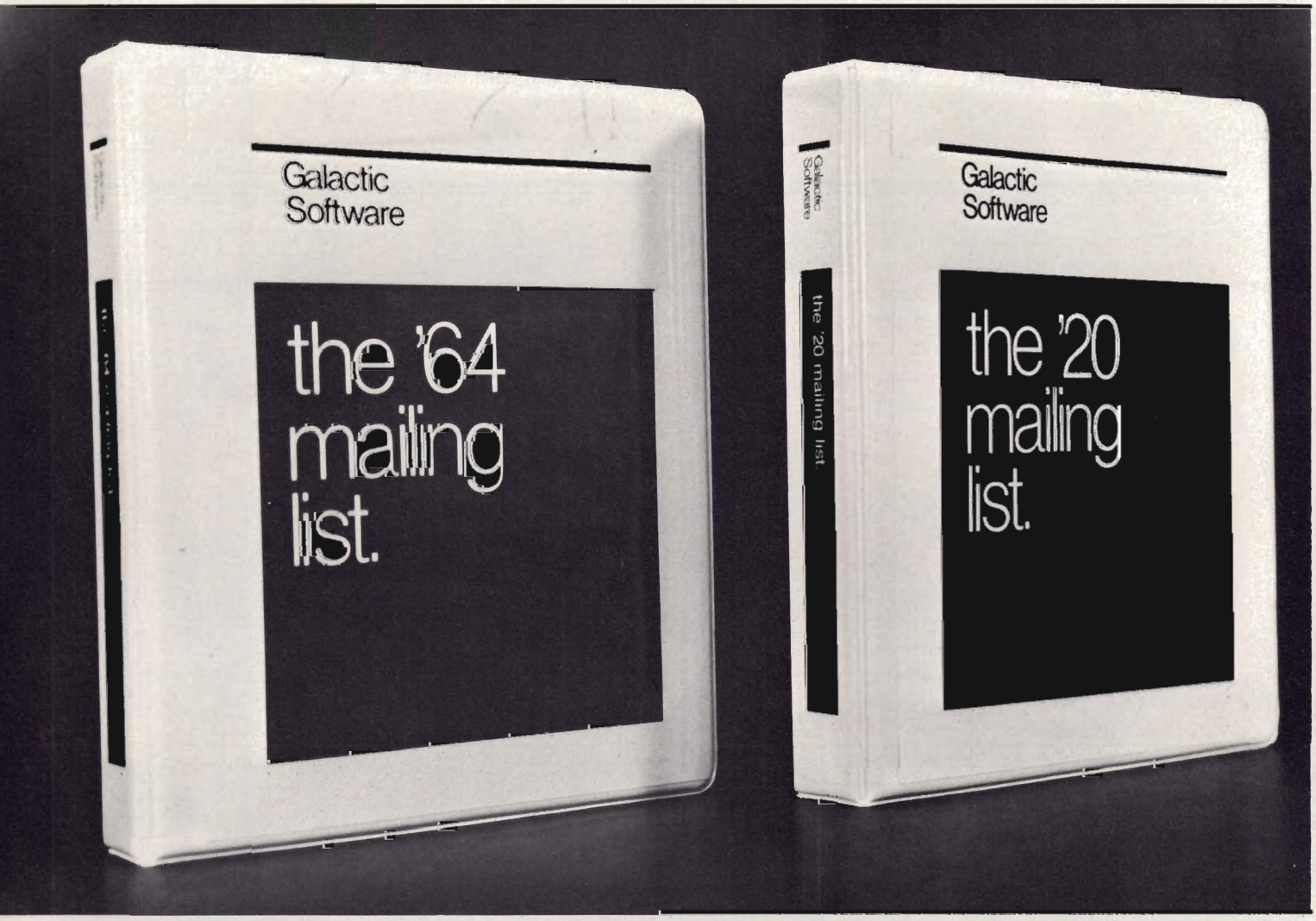
Physically, mentally and emotionally speaking, will staring at those glaring electrons all day cause eye strain, backaches and irritability in one who already does daily battle with ridicule, rejection and writer's block?

Finally, won't I miss the intimacy and interaction I've come to need and expect from my own writing?

Maybe I'm just the cautious type. The type who has a good thing going and knows it. The type who wouldn't divorce a perfectly good husband just because someone younger, handsomer, richer, smarter and sexier came along. Hmm. Maybe there is something to be said for modern technology. Maybe its byte isn't as bad as its bark.

Maybe my trusty, true-blue friend deserves a rest. After all, his predecessor, that cute little, rickety old portable could use some company in typewriter heaven.□

ENTER THE GALACTIC EXPERIENCE WITH



We wouldn't settle for second best, so why should you? In this day, too many people are compromising quality for price. We feel that we have the alternative. Announcing THE MAILING LISTS, the most powerful programs of their kind.

By being completely menu-driven, these programs are friendly and very easy to use. With each record containing name, address, city, state, zip code, telephone number, and several comments, you not only have a complete mailing list but also a small data base manager. With capabilities reaching into alphabetizing upon entry, sorting and searching on all fields, printing labels and printing complete records, you start to feel the Galactic Experience.

Currently there are four versions to cover anybodies needs.

These are:

The mailing list	VIC-20*	tape $25.95
The mailing list	VIC-20*	Disk $27.95
The mailing list	64 *	tape $27.95
The mailing list	64 *	Disk $29.95

To order, send check or money order to:

Galactic Software
P.O. Box 10516
San Jose, CA 95157

or phone (408) 247-4434 for COD

or

see your local dealer
dealer inquiries invited

Circle No. 18

*VIC-20 and 64 are trade marks of Commodore Business Machines.

Revised Sprite Editor for the Commodore 64

by Gary Kiziak
Burlington, Ontario, Canada

Since my Sprite Editor first appeared in the February issue of Commander magazine, I have had numerous requests for the changes necessary to make the program work with cassette as well as disk. This article will present a new version of the Sprite Editor that incorporates those changes as well as a few others in the form of the additional commands.

If you typed in the Sprite Editor from the February issue, then you will not have to retype the whole program. Most of the lines are exactly the same. Some new lines have been added to incorporate the cassette routines and the new commands. Some of the old lines have also been changed for various reasons.

Rather than list only the lines that have changed and the new lines (which could cause problems should I forget just one) I have chosen to include the entire listing for both the **Sprite Boot** and the **Sprite Editor** programs. When making the changes, make sure you check each line carefully to see if any changes have been made.

One of the first changes that I decided to make was to include the data for the machine language routines in the **Sprite Boot** program, thus eliminating the necessity of the **Scroll.Data** program. This change is necessary for cassette users. For disk users, it will facilitate the copying of these programs from one disk to another (just **Load** the programs and **Save** them).

The New Commands

Here are the new commands that I added together with a brief description of their use. For a complete description of all the other commands see the February issue of Commander.

Shift-C—Copies the current sprite to

LISTING 1

```
10 REM
15 REM  SPRITE EDITOR BOOT
20 REM
25 PRINT "[CLEAR][WHITE]":POKE 53280,14
:POKE 53281,6
30 PRINT"[HOME][DOWN][DOWN][DOWN][DOWN]
[DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][DO
WN][DOWN][RIGHT][RIGHT][RIGHT][RIGHT][R
IGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT]LOADING ..
.[DOWN][DOWN]":GOSUB 100
40 PRINT "[HOME][BLUE]LOAD"CHR$(34)"SPR
ITE EDITOR"CHR$(34)",8"
50 POKE 198,8
55 FOR I=32422 TO 32761:READ X:POKE I,X
:NEXT
60 FOR I=1 TO 8:READ X:POKE 630+I,X:NEX
T
70 POKE 44,64:POKE 16*1024,0:POKE 56,12
6:CLR:NEW
100 PRINT"       [REV]    [OFF] [REV]
 [OFF] [REV]    [OFF] [REV] [OFF] [REV]
     [OFF] [REV]    [OFF]
110 PRINT"       [REV] [OFF]    [REV] [O
FF]  [REV] [OFF] [REV] [OFF]  [REV] [OF
F] [REV] [OFF]   [REV] [OFF]   [REV] [O
FF]
120 PRINT"       [REV] [OFF] [REV] [O
FF]  [REV] [OFF]  [REV] [OFF]   [RE
V] [OFF]   [REV] [OFF]
130 PRINT"
140 PRINT"       [REV]    [OFF] [REV]
 [OFF] [REV] [OFF] [REV]     [OFF] [REV
]    [OFF] [REV]    [OFF]
150 PRINT"       [REV] [OFF]    [REV] [O
FF]  [REV] [OFF] [REV] [OFF]   [REV] [O
FF]   [REV] [OFF]  [REV] [OFF] [REV] [O
FF]  [REV] [OFF]
```

the next page.

The previous Sprite Editor did have a command (CTRL-C) for copying a range of sprites from one area to another, but I found it awkward to use when I simply wanted to copy a single sprite from one page to the next—the kind of thing that you want to do when creating a series of almost identical sprites for animation purposes. Now the creation of these sprites can be simplified by pressing **Shift-C** and then making the necessary changes to the sprite on the next page. Note: before pressing **Shift-C**, make certain that the next page doesn't contain a sprite that you want to keep.

Shift-S—This is the command that saves a range of sprite definitions to cassette (S by itself saves it to disk.) The sprites are saved as a **Data** file and can later be loaded into the Editor for further editing or additional sprite creation.

Shift-L—Loads a **Data** file of sprites from cassette (CTRL-L loads it from disk).

Remember, if any of the commands **CTRL-L, Shift-L, S, Shift-S, CTRL-C, CTRL-D,** or **CTRL-P** is pressed accidently, you can return to the editor without carrying out that command simply by pressing RETURN at the next prompt.

CTRL-F—Flips the sprite being edited upside down.

I required this command when creating a sprite (actually a car) that was to move up and down the screen. Moving up the screen required one sprite, while moving down required the exact same sprite but flipped upside down. So now I simply copy the sprite that I am working on to the next page **(Shift-C) and then flip it upside down (CTRL-F).**

Shift-F—Flips the sprite being edited sideways.

As above, this command will be useful for creating sprites that will move left and right on the screen. The sequence **Shift-C** followed by **Shift-F** creates the necessary sprites.

CTRL-I—Inserts a blank line at the current cursor position, moving everything on and below that line down one line.

I found this command useful when

```
160 PRINT"      [REV] [OFF]¯   [REV] [O
FF]  [REV] [OFF] [REV] [OFF]   [REV] [O
FF]   [REV] [OFF]  [REV] [OFF] [REV] [O
FF]¯¯┐
170 PRINT"      ──── ──── ─   ─   ────
¯  "[HOME]
180 RETURN
200 DATA 169,0,160,71,153,0,126,136,192
,62,208,248,177,251,153,0
210 DATA 126,136,16,248,96,160,71,185,0
,126,141,13,127,136,185,0
220 DATA 126,141,12,127,136,185,0,126,1
41,11,127,162,2,46,13,127
230 DATA 46,12,127,46,11,127,62,72,126,
202,62,72,126,202,62,72
240 DATA 126,232,232,232,232,232,224,74
,144,227,136,192,255,208,200,162
250 DATA 71,189,72,126,157,0,126,202,16
,247,160,62,185,0,126,145
260 DATA 251,136,16,248,96,0,0,0,169,21
,141,2,0,169,83,141
270 DATA 253,0,169,4,141,254,0,162,0,16
0,0,161,251,10,72,169
280 DATA 46,144,2,169,160,145,253,200,1
04,192,24,240,16,192,16,240
290 DATA 4,192,8,208,232,230,251,208,2,
230,252,208,222,230;251,208
300 DATA 2,230,252,24,173,253,0,105,40,
141,253,0,173,254,0,105
310 DATA 0,141,254,0,206,2,0,208,192,96
,169,20,133,251,162,3
320 DATA 134,252,160,3,177,253,136,136,
136,145,253,230,253,208,2,230
330 DATA 254,198,252,208,237,198,251,20
8,231,169,0,145,253,200,145,253
340 DATA 200,145,253,96,169,20,133,251,
162,3,134,252,160,0,177,253
350 DATA 200,200,200,145,253,165,253,20
8,2,198,254,198,253,198,252,208
360 DATA 235,198,251,208,229,169,0,160,
1,145,253,200,145,253,200,145
370 DATA 253,96,162,21,160,2,24,177,253
,42,145,253,136,16,248,24
380 DATA 169,3,101,253,133,253,169,0,10
1,254,133,254,202,208,229,96
390 DATA 169,21,133,251,160,0,162,3,24,
177,253,106,145,253,200,202
400 DATA 208,247,24,152,101,253,133,253
,169,0,101,254,133,254,206,251
410 DATA 0,208,225,96
420 DATA 19,13,32,32,82,85,78,13

READY.
```

continued on page 28

creating a Pac-Man like sprite. My first attempt produced a somewhat skinny Pac-Man. I found it easier to move the bottom half of the sprite down and fill in the empty line than to work around the outline of the sprite to make it fatter.

CTRL-K—Kills (deletes) the line at the current cursor position, moving everything below upwards to fill in.

Some Other Changes

A number of other changes have been made simply for cosmetic purposes. For example, when the Sprite Editor is first run, page 200 comes up clear—not with the garbage that it used to have. I have also changed the large dot, that signified that the corresponding pixel of the sprite was on, to a reversed space. I feel that it looks better. Actually, you can change it to whatever character that you want by changing the value of PT in line 148. PT = 160 gives the reversed space, PT = 81 gives the large dot. You can choose whatever you feel looks best.

Working With Sprites on Cassette Based Systems

If you are writing programs to work with cassette, it would be somewhat awkward for the user to have to load a Data file of sprites from tape. It would be better to save the sprite tables along with the BASIC program, as I demonstrated in the previous Sprite Editor article, or to include the sprite definitions within the program itself using Data statements.

To facilitate converting your sprite definitions to Data statements, the program Sprites to Data (listing 3), is also included. To use it, first load the sprites into memory using the Sprite Editor. Then load the Sprites to Data program, run it and follow the prompts. Your Data statements will be created automatically for you. Delete lines 0 to 11 of the resulting program and you can now type in your BASIC program, or, if it already exists, you can Append this to the end of your BASIC program using the method that I described in my article *A Character Editor for the Commodore 64* which appeared in the June '83 issue. ENJOY!!!!□

REVISED SPRITE EDITOR—continued from page 27

```
86 REM ************************************
*****
87 REM *
    *
88 REM *      -8-  A SPRITE EDITOR
    *
89 REM *
    *
90 REM *     -6-  FOR THE COMMODORE 64
    *
91 REM *
    *
92 REM *          -11-     BY
    *
93 REM *
    *
94 REM *      -8-  GARRY G. KIZIAK
    *
95 REM *
    *
96 REM *      -8-  COPYRIGHT  1982
    *
97 REM *
    *
98 REM ************************************
*****
99 :
100 PRINT "[CLEAR][WHITE]";:IF G=0 THEN
 DIM CO$(15)
110 DEF FNX(X)=X-INT(X/24)*24:DEF FNY(X
)=X-INT(X/21)*21
120 IF PEEK(8181)<>81 THEN POKE 8181,81
:FOR X=0 TO 63:POKE 12800+X,0:NEXT
130 V$="[HOME][DOWN][DOWN][DOWN][DOWN][
DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][DOW
N][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][
DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][DOW
N]"
140 DOT$=".....":BL$
="
 "
141 FOR I=0 TO 7:TWO(I)=2↑I:NEXT
142 CO$(0)="BLACK":CO$(1)="WHITE":CO$(2
)="RED  ":CO$(3)="CYAN ":CO$(4)="PURPL"
144 CO$(5)="GREEN":CO$(6)="BLUE ":CO$(7
)="YELOW":CO$(8)="ORNGE":CO$(9)="BROWN"
146 CO$(10)="LT RD":CO$(11)="DK GY":CO$
(12)="MD GY":CO$(13)="LT GN"
148 CO$(14)="LT BL":CO$(15)="LT GY":MR=
PEEK(53276):PT=160:POKE 32554,PT
149 M0=PEEK(53285)AND15:M1=PEEK(53286)A
ND15:B=PEEK(53281)AND15:E=PEEK(53280)AN
D15
```

```
150 G=13*4096:CR$=CHR$(13):DE$=CHR$(20)
:C=PEEK(G+39)AND15:SX=30:SY=150:X1=0:Y1
=0
160 PA=200:SP=0:SC=1024+80+3:AD=32608
170 GOSUB 960
180 GOSUB 870
190 GOSUB 930
200 GOSUB 850
210 PX=0:PY=0
220 P=SC+PY*40+PX:Q=PEEK(P):R=Q
230 R=(NOTRAND128)OR(NOT128ANDR)
240 POKE P,R
250 FOR I=1 TO 30:GET A$:IF A$="" THEN
NEXT:GOTO 230
260 POKE P,Q
270 IF A$="[RIGHT]" THEN PX=FNX(PX+1):G
OTO 220
280 IF A$="[LEFT]" THEN PX=FNX(PX-1):GO
TO 220
290 IF A$="[DOWN]" THEN PY=FNY(PY+1):GO
TO 220
300 IF A$="[UP]" THEN PY=FNY(PY-1):GOTO
 220
310 IF A$=DE$ THEN Z2=(PEEK(P)AND127):P
OKE P,-(Z2=46)*46-(Z2<>46)*PT:GOTO 680
320 IF A$=" " THEN GOSUB 660:PX=FNX(PX+
1):GOTO 220
330 IF A$=CR$ THEN PX=0:PY=FNY(PY+1):GO
TO 220
340 IF A$="." THEN 740
350 IF A$="[CLEAR]" THEN GOSUB 820:GOTO
 210
360 IF A$="[REV]" THEN GOSUB 830:GOTO 2
10
370 IF A$="+" THEN 780
380 IF A$="-" THEN 800
390 IF A$="Q" THEN POKE G+21,0:PRINT "[
CLEAR]";:END
400 IF A$=">" THEN C=(C+1)AND15:POKE G+
39,C:GOSUB 3020
410 IF A$="[@]" THEN B=(B+1)AND15:POKE 53
281,B:GOSUB 3010
420 IF A$="[WHITE]" THEN E=(E+1)AND15:P
OKE 53280,E:GOSUB 3000
430 IF A$="[↑]" THEN X1=1-X1:GOSUB 900:GO
SUB 870:GOTO 220
440 IF A$="[↓]" THEN Y1=1-Y1:GOSUB 900:GO
SUB 870:GOTO 220
450 IF A$="[■]" THEN 580
460 IF A$="[■]" THEN 600
470 IF A$="[■]" THEN 620
480 IF A$="[■]" THEN 640
490 IF A$="[■]" THEN 1510
500 IF A$="S" THEN 1270
505 IF A$="[♦]" THEN 3500
```

continued on page 30

```
510 IF A$="■" THEN 1030
520 IF A$="■" THEN 1070
530 IF A$="■" THEN 1430
535 IF A$="L" THEN 3800
540 IF A$="[HOME]" THEN 210
550 IF A$="■" THEN GOSUB 1860:GOTO 170
555 IF A$="£" THEN 690
556 IF A$="M" THEN MR=255-MR:POKE 53276
,MR:GOSUB 3030
557 IF A$="0" THEN M0=(M0+1)AND15:POKE
53285,M0:GOSUB 3030
558 IF A$="1" THEN M1=(M1+1)AND15:POKE
53286,M1:GOSUB 3030
559 IF A$="■" THEN 3100
560 IF A$="_" THEN 3200
561 IFA$="─"ANDPA<>15ANDPA<>63ANDPA<>25
5THENFORX=0TO63:POKE(PA+1)*64+X,PEEK(PA
*64+X):NEXT
562 IF A$="■" THEN FOR X=62 TO PY*3+3 S
TEP-1:Z2=PA*64+X:POKE Z2,PEEK(Z2-3):NEXT
563 IF A$="■" THEN FOR J=0 TO 2:POKE PA
*64+PY*3+J,0:NEXT:GOSUB 850
564 IF A$="■" THEN FOR X=PY*3 TO 59:Z2=
PA*64+X:POKE Z2,PEEK(Z2+3):NEXT
565 IF A$="■" THEN FOR J=0 TO 2:POKE PA
*64+60+J,0:NEXT:GOSUB 850
569 GOTO 220
570 POKE G+21,0:GOSUB 930:GOSUB 870:GOS
UB 850:GOTO 210
574 REM ******************************
575 REM *                            *
576 REM *  MOVE ENTIRE SPRITE UP,    *
577 REM *   DOWN, LEFT, OR RIGHT     *
578 REM *                            *
579 REM ******************************
580 J=PA*64:POKE 253,J-256*INT(J/256):P
OKE 254,J/256
590 SYS AD:GOSUB 850:GOTO220
600 J=PA*64+59:POKE 253,J-256*INT(J/256
):POKE 254,J/256
610 SYS AD+42:GOSUB 850:GOTO220
620 J=PA*64:POKE 253,J-256*INT(J/256):P
OKE 254,J/256
630 SYS AD+88:GOSUB 850:GOTO220
640 J=PA*64:POKE 253,J-256*INT(J/256):P
OKE 254,J/256
650 SYS AD+118:GOSUB 850:GOTO220
655 REM ******************************
656 REM *                            *
657 REM * ERASE OR DELETE A POINT    *
658 REM *                            *
659 REM ******************************
660 POKE P,46:PP=PA*64+PY*3+INT(PX/8)
670 POKE PP,PEEK(PP) AND 255-2↑(7-(PX-I
NT(PX/8)*8)):RETURN
```

```
680 PX=FNX(PX-1):P=SC+PY*40+PX:GOSUB 66
0:GOTO 220
685 REM ******************************
686 REM *                            *
687 REM * ROTATE SPRITE 90 DEGREES *
688 REM *                            *
689 REM ******************************
690 HI=INT(PA/4):LO=PA*64-256*HI:POKE 2
51,LO:POKE 252,HI:SYS 32422
700 SYS 32443:SYS 32526:POKE 251,LO:POK
E 252,HI
710 GET A$:IF A$<>"£" AND A$<>CR$ THEN 710
720 IF A$="£" THEN 700
730 GOTO 220
735 REM ******************************
736 REM *                            *
737 REM *        PLOT A POINT        *
738 REM *                            *
739 REM ******************************
740 POKE P,PT
750 PP=PA*64+PY*3+INT(PX/8)
760 POKE PP,PEEK(PP) OR 2↑(7-(PX-INT(PX
/8)*8))
770 PX=FNX(PX+1):GOTO 220
775 REM ******************************
776 REM *                            *
777 REM * NEXT OR PREVIOUS SPRITE  *
778 REM *                            *
779 REM ******************************
780 IF PA<15 OR (PA>31ANDPA<63) OR(PA>1
27ANDPA<255) THEN PA=PA+1:GOTO570
790 GOTO 220
800 IF(PA>13ANDPA<16) OR (PA>32ANDPA<64
) OR(PA>128ANDPA<256) THEN PA=PA-1:GOTO
570
810 GOTO 220
815 REM ******************************
816 REM *                            *
817 REM *        CLEAR SPRITE        *
818 REM *                            *
819 REM ******************************
820 FOR X=0 TO 63:POKE PA*64+X,0:NEXT:G
OSUB 850:RETURN
825 REM ******************************
826 REM *                            *
827 REM *      REVERSE SPRITE        *
828 REM *                            *
829 REM ******************************
830 FOR X=0 TO 63:POKE PA*64+X,255-PEEK
(PA*64+X):NEXT:GOSUB 850:RETURN
845 REM ******************************
846 REM *                            *
847 REM * DISPLAY SPRITE ON SCREEN *
848 REM *                            *
849 REM ******************************
```

continued on page 32

```
850 POKE 251,PA*64-INT(PA/4)*256:POKE 2
52,PA/4
860 SYS 32526:PRINT"[WHITE][HOME]":RETU
RN
865 REM ******************************
866 REM *                            *
867 REM *  INITIALIZE SPRITE DATA    *
868 REM *                            *
869 REM ******************************
870 POKE G+21,0:POKE 2040+SP,PA:POKE G+
39,C
880 POKE G+SP*2,SX:POKE G+SP*2+1,SY:POK
E G+23,Y1:POKE G+29,X1
890 POKE G+16,2↑SP:POKE G+21,2↑SP:RETUR
N
895 REM ******************************
896 REM *                            *
897 REM *  EXPAND/CONTRACT SPRITE    *
898 REM *                            *
899 REM ******************************
900 SX=17:IF X1=0 THEN SX=30
910 SY=138:IF Y1=0 THEN SY=150
920 RETURN
925 REM ******************************
926 REM *                            *
927 REM *  DISPLAY EDITING SCREEN    *
928 REM *                            *
929 REM ******************************
930 PRINT "[HOME][DOWN][DOWN][DOWN]"TAB
(29)"|          |":PRINT TAB(29)"| PAGE
 |"
940 PRINT TAB(29)"└──────────┘":PRINT "[U
P][UP]"TAB(34)PA
950 RETURN
960 PRINT "[HOME]■ ──────────────────────
──────── [WHITE]──────────"
970 PRINT " ■[REV]  1234567890123456789
01234 [OFF] [REV][WHITE]  SPRITE  [DOWN
][LEFT][LEFT][LEFT][LEFT][LEFT][LEFT][L
EFT][LEFT][LEFT][LEFT]  EDITOR  [HOME][
DOWN]"
980 FOR X=1 TO 21
990 PRINT " ■[REV]"RIGHT$(" "+STR$(X),2
)"[OFF][WHITE]"DOT$"■[REV] [OFF]":NEXT
1000 PRINT " [REV]
      [OFF][WHITE]"
1001 PRINT LEFT$(V$,18)TAB(30)"COLOURS"
:PRINTTAB(30)"────────"
1002 PRINT TAB(29)"BRDR:":GOSUB 3000
1004 PRINT TAB(29)"BKGD:":GOSUB 3010
1006 PRINT TAB(29)"SPRT:":GOSUB 3020:IF
 MR=0 THEN RETURN
1008 GOSUB 3030
1010 RETURN
```

```
1015 REM ****************************
1016 REM *                          *
1017 REM *    PROMPT FOR NEW PAGE   *
1018 REM *                          *
1019 REM ****************************
1020 IF IN$="" THEN PA=AP:GOTO 180
1030 LI=10:COL=31:LE=3:MSG$="[OFF]PAGE[
OFF]":GOSUB 1700:AP=PA:PA=VAL(IN$)
1040 PRINTLEFT$(V$,LI)TAB(COL-1)"
   "
1050 IF IN$="" OR PA<13 OR (PA>15ANDPA<
32)OR (PA>63ANDPA<128) OR PA>255 THEN P
A=AP
1060 GOTO 180
1065 REM ****************************
1066 REM *                          *
1067 REM * DISPLAY RANGE OF SPRITES *
1068 REM *                          *
1069 REM ****************************
1070 POKE G+21,0:POKE G+16,0:POKE G+23,
0:POKE G+29,0:GOSUB 1250
1080 LI=8:COL=6:LE=3:MSG$="[OFF]FROM PA
GE:[OFF]"
1090 GOSUB 1700:GP=PG:PG=VAL(IN$):IFIN$
="" THEN PRINT "[CLEAR]":GOTO 170
1100 IF PG<0 OR PG>255 OR (PG=0 AND IN$
<>"0") THEN 1080
1110 LI=8:COL=22:MSG$="[OFF]TO PAGE:[OF
F]"
1120 GOSUB 1700:PH=VAL(IN$):IF PH<PG OR
 PH>255 THEN 1110
1130 SW=PG
1140 SUM=0:EN=SW+7:IF EN>PH THEN EN=PH:
IF SW>PH THEN 170
1150 GOSUB 1250:FOR I=SW TO EN:K=I-SW:M
=K:IF M>3 THEN M=M-4
1160 POKE 2040+K,I:SUM=SUM+21^K
1170 POKE G+K*2,M*48+104:L=98:L1=10:IF
K>3 THEN L=155:L1=17
1180 POKE G+K*2+1,L:POKE G+39+K,1:POKE
G+21,SUM
1190 PRINT LEFT$(V$,L1)TAB(M*6+9)I
1200 NEXT
1210 PRINTLEFT$(V$,20)TAB(7)"
      "
1220 PRINTTAB(7)"PRESS [REV]SPACE BAR[O
FF] TO CONTINUE"
1230 GET A$:IF A$<>" "THEN 1230
1240 POKE G+21,0:PRINT "[CLEAR]":SW=EN+
1:GOTO 1140
1250 PRINT "[CLEAR]
                    "
1260 PRINT "          [REV] DISPLAY S
PRITES [OFF]":RETURN
```

continued on page 34

```
1265 REM ******************************
1266 REM *                            *
1267 REM *   SAVE SPRITES TO DISK     *
1268 REM *                            *
1269 REM ******************************
1270 POKE G+21,0:PRINT "[CLEAR]
     ____________________________"
1280 PRINT "          [REV] SAVE SPRITES
 TO DISK [OFF]"
1290 LI=8:COL=6:LE=3:MSG$="[OFF]FROM PA
GE:[OFF]"
1300 GOSUB 1700:PG=VAL(IN$):IF IN$="" T
HEN PRINT "[CLEAR]":GOTO 170
1310 IF PG<13 OR (PG>15 AND PG<32) OR (
PG>63 AND PG<128) OR PG>255 THEN 1290
1320 LI=8:COL=22:MSG$="[OFF]TO PAGE:[OF
F]":GOSUB 1700:PH=VAL(IN$)
1330 IF PH<PG OR (PH>15 AND PH<32) OR (
PH>63 AND PH<128) OR PH>255 THEN 1320
1340 BEG=PG*64:EN=PH*64+63:HI=INT(BEG/2
56):LO=BEG-HI*256
1350 LI=10:COL=12:LE=16:MSG$="[OFF]FILE
NAME:[OFF]":GOSUB 1700:FL$="0:"+IN$+",P
RG,WRITE"
1360 OPEN 1,8,15,"I0":GOSUB 1810
1370 OPEN 2,8,1,FL$:GOSUB 1810
1380 PRINT#2,CHR$(LO);CHR$(HI);
1390 FOR I=BEG TO EN:X=PEEK(I):PRINT#2,
CHR$(X);:NEXT:GOSUB 1810
1400 CLOSE 2
1410 CLOSE 1
1420 PRINT "[CLEAR]":GOTO 170
1425 REM ******************************
1426 REM *                            *
1427 REM *  LOAD SPRITES FROM DISK    *
1428 REM *                            *
1429 REM ******************************
1430 POKE G+21,0:PRINT "[CLEAR]
     ____________"
1440 PRINT "               [REV] DISK LO
AD [OFF]"
1450 LI=10:COL=12:LE=16:MSG$="[OFF]FILE
NAME:[OFF]":GOSUB 1700:FL$="0:"+IN$
1460 IF IN$="" THEN PRINT "[CLEAR]":GOT
O 170
1470 OPEN1,8,15,"I0"
1480 OPEN2,8,0,FL$:GOSUB 1810:CLOSE 2:C
LOSE 1
1490 LOAD FL$,8,1  1500 END
1504 REM ******************************
1505 REM *                            *
1506 REM *  COPY SPRITES TO ANOTHER   *
1507 REM *       AREA IN MEMORY       *
1508 REM *                            *
1509 REM ******************************
```

```
1510 POKE G+21,0:PRINT "[CLEAR]
   ____________________"
1520 PRINT "          [REV] COPY SPRIT
E DATA [OFF]"
1530 PRINT LEFT$(V$,4)" ____________
____":PRINT " [REV]** SOURCE PAGES **"
1540 LI=7:COL=4:LE=3:MSG$="[OFF]FROM PA
GE:[OFF]"
1550 GOSUB 1700:PG=VAL(IN$):IF IN$="" T
HEN PRINT "[CLEAR]":GOTO 170
1560 IF PG<13 OR (PG>15 AND PG<32) OR (
PG>63 AND PG<128) OR PG>255 THEN 1540
1570 LI=9:COL=4:LE=3:MSG$="[OFF]..TO PA
GE:[OFF]":GOSUB 1700:PH=VAL(IN$)
1580 IF PH<PG OR (PH>15 AND PH<32) OR (
PH>63 AND PH<128) OR PH>255 THEN 1570
1590 PRINT LEFT$(V$,4)TAB(21)"_________
_________":PRINT TAB(21)"[REV]** TARGET
 PAGES **"
1600 LI=7:COL=24:LE=3:MSG$="[OFF]FROM P
AGE:[OFF]":GOSUB 1700:PD=VAL(IN$)
1610 IF PD<13 OR (PD>15 AND PD<32) OR (
PD>63 AND PD<128) OR PD>255 THEN 1600
1620 PE=PD+PH-PG:IF PE>255 THEN PE=255
1630 PRINTLEFT$(V$,9)TAB(23)"..TO PAGE:
"PE
1640 IF PD>PG AND PD<=PH THEN 1670
1650 FOR I=PD TO PE:PRINT LEFT$(V$,12)T
AB(12)"COPYING PAGE"I
1660 FOR J=0 TO 63:POKE I*64+J,PEEK((PG
+I-PD)*64+J):NEXT:NEXT:PRINT "[CLEAR]":
GOTO 170
1670 FOR I=PE TO PD STEP -1:PRINT LEFT$
(V$,12)TAB(12)"COPYING PAGE"I
1680 FOR J=0 TO 63:POKE I*64+J,PEEK((PG
+I-PD)*64+J):NEXT:NEXT:PRINT "[CLEAR]":
GOTO 170
1690 END
1695 REM ******************************
1696 REM *                            *
1697 REM *      INPUT ROUTINE         *
1698 REM *                            *
1699 REM ******************************
1700 Y9=2:IN$="":UC=0:UB$=LEFT$(BL$,LE)
:GOSUB 1800:UB$=" ":UC=3
1710 UT=TI
1720 GET Z9$:IF Z9$="" THEN 1780
1730 IF Z9$=CR$ THEN Y9=2:GOSUB 1800:PR
INT "[LEFT][LEFT] ":RETURN
1740 IF Z9$=DE$ THEN ON -(LEN(IN$)=0) G
OTO 1780:IN$=LEFT$(IN$,LEN(IN$)-1):GOTO
1780
1750 IF (ASC(Z9$)AND127)<32 OR Z9$=CHR$
(34) THEN 1780
1760 IF LE=LEN(IN$) THEN 1780
1770 IN$=IN$ + Z9$
```

continued on page 36

REVISED SPRITE EDITOR—continued from page 35

```
1780 GOSUB 1800:IF TI-UT<10 THEN 1720
1790 Y9=3-Y9:GOTO 1710
1800 PRINT LEFT$(V$,LI)TAB(COL-1)MID$(M
SG$,Y9)UB$IN$MID$("[REV] [OFF]",Y9,UC)"
 ";:RETURN
1805 REM ******************************
1806 REM *                            *
1807 REM *  CHECK FOR DISK ERRORS     *
1808 REM *                            *
1809 REM ******************************
1810 INPUT#1,A$,B$,C$,D$
1820 IF VAL(A$)=0 THEN RETURN
1830 PRINT"[CLEAR][DOWN][DOWN][RIGHT][R
EV]DISK ERROR:[OFF] "B$
1840 CLOSE2
1850 END
1855 REM ******************************
1856 REM *                            *
1857 REM *  VIEW SPRITE IN MOTION     *
1858 REM *                            *
1859 REM ******************************
1860 POKE G+21,0:PRINT "[CLEAR]":POKE G
+16,0
1870 X=INT(RND(1)*100)+155:Y=INT(RND(1)
*100)+75:DX=2:DY=1:X2=0:Y2=0
1880 POKE G,X:POKE G+1,Y:POKE G+21,1
1890 B$=" ":X=X+DX:Y=Y+DY:GET A$:IF A$<
>"" THEN 1950
1900 IF X>255 THEN X=255:DX=-DX
1910 IF Y>200 THEN Y=200:DY=-DY
1920 IF X<65 THEN X=65:DX=-DX
1930 IF Y<75 THEN Y=75:DY=-DY
1940 GOTO 1880
1950 IF A$="+" THEN DX=DX+SGN(DX):DY=DY
+SGN(DY):DX=DX-(DX=0):DY=DY-(DY=0)
1960 IF A$="-" THEN DX=DX-SGN(DX):DY=DY
-SGN(DY):DX=DX+(DX=0):DY=DY+(DY=0)
1970 IF A$="+" OR A$="-" THEN 1890
1980 IF A$="[F3]" THEN B=(B+1)AND15:POKE 5
3281,B
1990 IF A$="[WHITE]" THEN E=(E+1)AND15:
POKE 53280,E
2000 IF A$=">" THEN C=(C+1)AND15:POKE G
+39,C
2010 IF A$="[F5]" THEN X2=1-X2:POKE G+29,X 2
2020 IF A$="[F7]" THEN Y2=1-Y2:POKE G+23,Y
2
2030 IF A$=CR$ THEN POKE G+21,0:RETURN
2040 IF A$<>B$ THEN 1890
2050 GET B$:A$=B$:IF B$<>" " THEN 1980
2060 GOTO 1890
2095 REM ******************************
2096 REM *                            *
2097 REM * DISPLAY COLOUR REGISTERS   *
2098 REM *                            *
2099 REM ******************************
```

```
3000 PRINT LEFT$(V$,20)TAB(34)CO$(E):RE
TURN
3010 PRINT LEFT$(V$,21)TAB(34)CO$(B):RE
TURN
3020 PRINT LEFT$(V$,22)TAB(34)CO$(C):RE
TURN
3030 A1$="RG-0:"+CO$(M0):A2$="RG-1:"+CO
$(M1):IFMR=0THEN A1$="            ":A2$=A
1$
3040 PRINT LEFT$(V$,23)TAB(29)A1$:PRINT
 TAB(29)A2$"[HOME]":RETURN
3095 REM ******************************
3096 REM *                            *
3097 REM * FLIP SPRITE UPSIDE DOWN    *
3098 REM *                            *
3099 REM ******************************
3100 FOR I=0 TO 9:Z1=PA*64+I*3:W1=(PA+1
)*64-I*3-4
3110 FOR J=0 TO 2:Z2=PEEK(Z1+J):POKE Z1
+J,PEEK(W1+J):POKE W1+J,Z2:NEXT J
3120 NEXT I:GOSUB 850:GOTO 220
3195 REM ******************************
3196 REM *                            *
3197 REM *   FLIP SPRITE SIDEWAYS     *
3198 REM *                            *
3199 REM ******************************
3200 FOR I=0 TO 20:Z1=PA*64+I*3
3210 FOR J=0 TO 2:X4=PEEK(Z1+J):GOSUB 3
240:X(J)=X5:NEXT J
3220 FOR J=0TO 2:POKE Z1+J,X(2-J):NEXT
J
3230 NEXT I:GOSUB 850:GOTO 220
3240 X5=0:FOR L=0 TO 7:IF X4ANDTWO(L) T
HEN X5=X5+TWO(7-L)
3250 NEXT:RETURN
3495 REM ******************************
3496 REM *                            *
3497 REM * SAVE SPRITES TO CASSETTE   *
3498 REM *                            *
3499 REM ******************************
3500 POKE G+21,0:PRINT "[CLEAR]
                              "
3510 PRINT "    [REV] SAVE SPRITES T
O CASSETTE [OFF]"
3520 LI=8:COL=7:LE=3:MSG$="[OFF]FROM PA
GE:[OFF]"
3530 GOSUB 1700:PG=VAL(IN$):IF IN$="" T
HEN PRINT "[CLEAR]":GOTO 170
3540 IF PG<13 OR (PG>15 AND PG<32) OR (
PG>63 AND PG<128) OR PG>255 THEN 3520
3550 LI=8:COL=23:MSG$="[OFF]TO PAGE:[OF
F]":GOSUB 1700:PH=VAL(IN$)
3560 IF PH<PG OR (PH>15 AND PH<32) OR (
PH>63 AND PH<128) OR PH>255 THEN 3550
3570 BEG=PG*64:EN=PH*64+63:HI=INT(BEG/2
56):LO=BEG-HI*256
```

continued on page 38

REVISED SPRITE EDITOR—continued from page 37

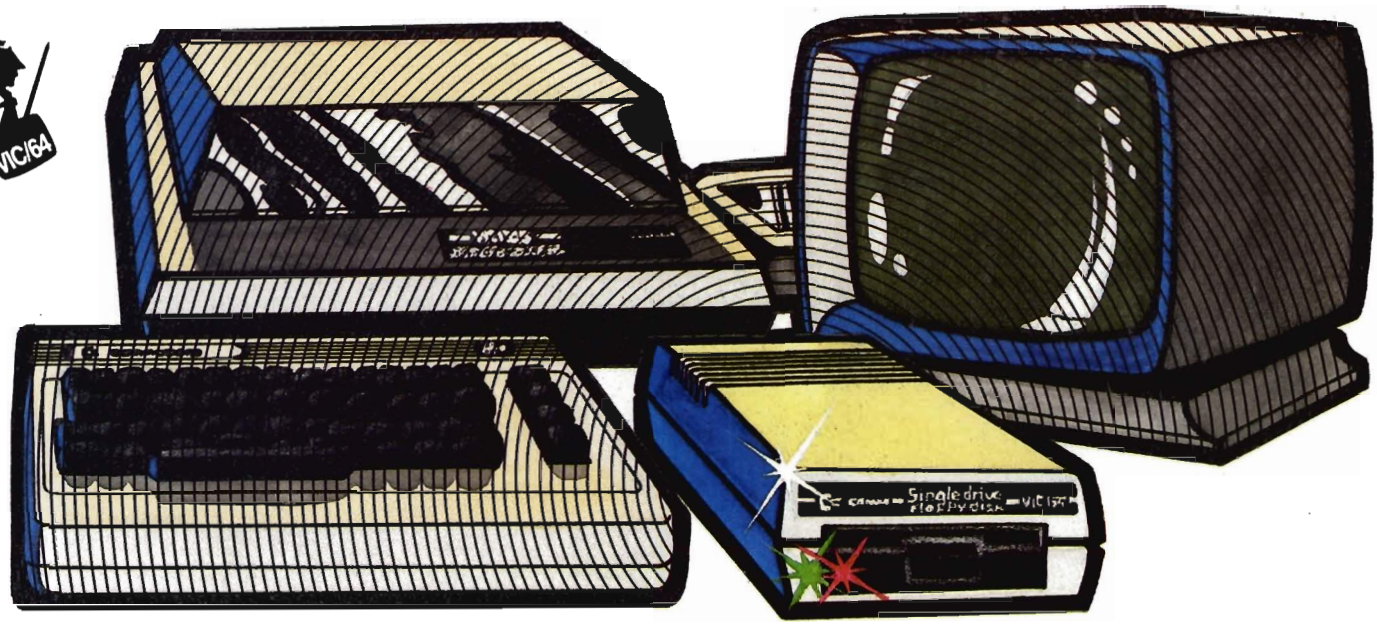
```
3575 HZ=INT(EN/256):LZ=EN-HZ*256
3580 LI=10:COL=12:LE=16:MSG$="[OFF]FILE
NAME:[OFF]":GOSUB 1700:FL$=IN$
3585 PRINT"[DOWN][DOWN][RIGHT][RIGHT][R
IGHT][RIGHT][RIGHT][RIGHT]PRESS  RECORD
 & PLAY ON TAPE[DOWN][DOWN]":PRINTTAB(1
1)"THEN PRESS [REV]RETURN[OFF]"
3588 GET A$:IF A$<>CHR$(13)THEN 3588
3590 OPEN 2,1,1,FL$
3600 PRINT#2,CHR$(LO);CHR$(HI);CHR$(LZ)
;CHR$(HZ);
3610 FOR I=BEG TO EN:X=PEEK(I):PRINT#2,
CHR$(X);:NEXT
3620 CLOSE 2
3630 PRINT "[CLEAR]":GOTO 170
3795 REM ********************************
3796 REM *                              *
3797 REM * LOAD SPRITES FROM CASSETTE *
3798 REM *                              *
3799 REM ********************************
3800 POKE G+21,0:PRINT "[CLEAR]
                        "
3810 PRINT "          [REV] CASSETTE
LOAD [OFF]"
3820 LI=7:COL=13:LE=16:MSG$="[OFF]FILEN
AME:[OFF]":GOSUB 1700:FL$=IN$
3830 IF IN$="" THEN PRINT "[CLEAR]":GOT
O 170
3833 PRINT TAB(12)"[DOWN][DOWN][DOWN][D
OWN]INSERT DATA TAPE[DOWN]":PRINT TAB(1
1)"PRESS PLAY ON TAPE[DOWN]"
3835 PRINT TAB(11)"THEN PRESS  [REV]RET
URN[OFF]"
3838 GET A$:IF A$<>CHR$(13) THEN 3838
3840 OPEN 2,1,0,FL$
3850 GET#2,X$,Y$,Z$,W$:BEG=ASC(X$+CHR$(
0))+256*ASC(Y$+CHR$(0))
3860 EN=ASC(Z$+CHR$(0))+256*ASC(W$+CHR$
(0))
3870 FOR I=BEG TO EN:GET#2,X$:POKE I,AS
C(X$+CHR$(0)):NEXT
3880 CLOSE 2:PRINT "[CLEAR]":GOTO 170

READY.
||||||||||||||||||||||0 POKE53281,6:GOTO7
1 READL,I,S,E:PRINT"[WHITE][CLEAR][DOWN
][DOWN][DOWN]";MID$(STR$(L),2);" DATA "
;
2 PRINTMID$(STR$(PEEK(S)),2);
3 S=S+1:T=T+1:IFS>=ETHENPRINT:PRINT"[BL
UE]PRINTCHR$(147)CHR$(5)":GOTO6
4 IFT<16THENPRINT",";:GOTO2
5 L=L+I:PRINT"[BLUE][HOME][DOWN][DOWN]0
DATA";L;"[LEFT],";I;"[LEFT],";S;"[LEFT]
,";E:PRINT"[HOME][DOWN][DOWN][DOWN][DOW
```

```
N][DOWN]RUN";
6 PRINT"[HOME]";:POKE631,13:POKE632,13:
POKE633,13:POKE634,13:POKE198,4:END
7 PRINT"[CLEAR][WHITE]"TAB(4)"_________
_____________________"
8 PRINTTAB(4)"[REV] SPRITE DATA TO DATA
 STATEMENTS "
9 INPUT"[DOWN][DOWN][DOWN][RIGHT][RIGHT
][RIGHT][RIGHT]STARTING LINE NUMBER ";L
:INPUT "[DOWN][RIGHT][RIGHT][RIGHT][RIG
HT]INCREMENT ";I:L=L-I
10 INPUT"[DOWN][RIGHT][RIGHT][RIGHT][RI
GHT]STARTING PAGE ";S:S=S*64:INPUT"[DOW
N][RIGHT][RIGHT][RIGHT][RIGHT]ENDING PA
GE ";E:E=(E+1)*64
11 PRINT"[CLEAR]":GOTO5
READY.
```

Circle No. 10

Circle No 38



So You Bought a Disk Drive!?

by R.G. Partner
Federal Way, WA

The Commodore 1540/1541 disk drive is a remarkable piece of equipment. It is one of the few disk drives on the market which, when plugged in, allows the computer mainframe to retain all the memory that was available without the disk drive.

How can that be, you ask? The 1540/1541 disk drive could be called a Smart Disk Drive. It has its own microprocessor, ROM (read only memory) and RAM (random access memory) built in. In other words it is self-contained. No extra cables to make it work. No expansion chassis to make it work. When you buy the 1540/1541 disk drive it is ready to go with whatever memory your machine has.

It has a total storage capacity of 174848 bytes of data on a 5¼ inch diskette. When storing sequential files and relative files you have slightly less capacity. The disk will hold 664 blocks of information. Each block contains 255 bytes of information.

In case you're counting, that does not come out to 174848 bytes. There is a track taken up with directory information and something called Block Availability Map or BAM (more about this later).

I must preface the Load and Save operation with a brief discussion about disk drive operation. There are several important items that need to be covered before we actually perform Load and Save.

One of the most important things is that each disk must or should have a Unique ID number. It may save you hours of frustration later. **Non- Unique ID's** may become a real problem later on.

Commodore has chosen the word **Format** to mean the process of **Formatting a disk. Industry often uses the term Initialize.** What does **Formatting** mean? Formatting is the process used to allocate specific spaces on the diskette where information will be stored. Six hundred sixty four blocks of 255 Bytes are marked electromagnetically so the disk drive can go to any one track and sector and retrieve data. There are three basic types of disk drives used. They are:

1) Soft sectored
2) Hard sectored
3) Hard disks

Type one, the Soft Sectored disk, is the type used by the 1540/1541 disk drive and other Commodore Disk Drives.

A phonograph record could be used to illustrate what the Sectored Disk actually is. A phonograph record is **Hard Sectored.** You cannot change the placement of the music (or tracks) on the record. The Producers planned for a song or track to occupy a specific place on the record. They **Formatted** the record to contain specific information. We cannot change that format without damaging the record. So it is with **Hard Sectored Disks**. The data and programs can be erased but you cannot change the location of the individual sectors. The Hard Sectored diskette contains a series of holes around the disk. The holes mark the beginning and ending of individual tracks and sectors. Disk drives which require **Hard Sectored Disks** contain a sensor to check for the additional holes. The manufacturer has defined the placement of information on the disk and you cannot change it. The Soft Sectored disk on the other hand allows you, the user, to define, within the limits of the disk, and operating system, where you will put information. With this type of disk system we must **Format** the disk (i.e., we must define the sectors). This is done for us by the internal **Disk Operating System (DOS)** when we give the command.

Commodore packages with the 1540/1541 disk drive a disk titled

Test/Demo. One program on the disk is a **Performance Test** which checks the Disk drive for proper operation. It cannot check for all possible disk drive problems but it does cover many of the common problems that can arise.

Also on the disk is a program titled **Wedge.** This is perhaps one of the most useful programs to the beginner and experienced user alike. The instruction book that comes with the 1540 disk drive does not tell how to use this program. In fact the book does not even acknowledge the program is present on the **Test/Demo** disk. The 1541 book does mention it briefly. We will be covering the use of this nifty little program a little later so hold on.

Let's go through some of the operations of the disk so you can become familiar with how to **Format** a disk and how to **Load** and **Save** a program. **Formatting** a new disk or reformatting an old disk. **Caution: Formatting a disk will erase all programs, files, and records.**

It is a simple process with the 1540/1541 disk drive.

We type in the command:

```
OPEN 15,8,15,"N:?????????,nn":
CLOSE 15
```

We are telling the disk drive to **Format** a new disk (or an old one we want to re-use). The N is short for **New.** The question marks represent the name we wish to give the new disk (up to 16 characters). The nn represents the **Disk ID** number. This can be two letters, two numbers or a combination.

We need to assign to each disk a special and **Unique ID Number.** When you insert a disk in the drive and load a program into the computer the **Disk Operating System (DOS)** goes to the directory information on the disk. It checks to see if the program you seek is actually there. It also stores the directory information in memory in the disk drive.

It also stores something called Block Availability Map (**BAM** for short) in disk memory. When we want to load another program, the computer looks in disk memory for the information. What does the directory contain? Information about on which track and in which sector the program you want resides. **BAM** (Block Availability Map) is a Disk memory representation of available and allocated space on the **Disk. BAM is checked by DOS** for available space to save a program or data to. The **BAM** is updated each time a program or data is stored on the disk. A separate and unique ID number should be assigned to each disk. If you have two disks with the same ID number and put the first one into the drive, **Load** a program then insert a second disk and load another program, reinsert the first disk and load a program from the first disk. There is a chance that the DOS (Disk Operating System) could get them confused and try to load a program that is not there.

Worse yet is the possibility that you would **Save** a program after changing disks and the **DOS**, thinking the first disk was still in place, would **Save** the program based on the **Old Directory** and BAM information.

This could result in the program you are trying to save writing over another program thereby destroying not one but two programs. This can easily happen if you do not assign a Unique ID number to each disk. A solution to prevent the directory mixup is to type:

```
OPEN15,8,15,"i":CLOSE15
```

each time you change diskettes. This commands the disk operating system to get the directory and put it into the disk memory. Now you have current directory information and the BAM is update.

Right at the beginning decide on a plan of action for labeling your disks. eg. UA to UZ for disks containing **Utility** programs. GA to GZ for **Games,** MA to MZ for **Math** programs WA to WZ for wordprocessors. That allows 26 **Unique** and individual ID numbers for each disk category. That should be more than enough for a long time and many many disks full of data.

We will look at another way of doing this with the **Wedge** and save some typing. If you have a program saved on tape, now is the time to load it into the computer. The next step will be to save that program to the newly formatted disk. Got it loaded? Good,

now let's save it to the disk. Type the following:

SAVE"program name",8

The red disk light should come on and you should hear the disk motor running. When the program is saved the disk drive motor should shut off. The red light on the disk drive should go out. Before you get too far in storing programs to your disk why not read on for some short cuts!

Directory

Caution: Typing in the following command will erase what you have in memory!

How do I know what is on my disk? Easy, type the command:

LOAD"$",8 press RETURN

the disk will spring into action. When the computer signals **READY** type:

LIST

There you have information about what is on the disk you now have in the disk drive. What does the information mean? The number to the far left represents the number of blocks your program occupies on the disk. Remember that one block contained 255 bytes of information? To determine how long your program takes multiply the number of blocks the program takes up by 255 and presto you have the length of the program. The next thing you see is the name of the program. Just under the program name you should see the abbreviation "prg" for program. If this was a sequential date file it would say SEQ. There are some other words and symbols that may appear. An important one is the * which indicates that the file was not properly closed.

Short Cut with VIC Wedge

Take your disk out of the drive and insert the **Test/Demo** disk. Type:

LOAD"VICWEDGE",8
RETURN

When the computer signals **READY** type **RUN**. This will bring up an unusual display which is quite cryptic. Don't despair. If you will follow closely

you will save a lot of time typing later. This is the short cut I promised.

VIC Wedge will look something like the following:

```
        VIC-20 VERSION 2.6
                        DISK STATUS
@                    OR COMMANDS
  $0                    DIRECTORY
@$0
/FILENAME                    LOAD
```

Let's start at the top of the command list. The first is >

DISK STATUS

With the **Wedge** in memory, this symbol ' > ' can be typed when we have any kind of a **Disk Error**. When the **RED** light on the disk drive is flashing typing ' > ' will tell you what the error message is. Your operator's manual has a list of these error messages in it and should be consulted to determine the specific error code.

The next symbol:

@ OR COMMANDS

Using this symbol allows you to execute all the disk commands except SAVE. Some of the commands are as follows:

NEW
SCRATCH
INITIALIZE
RENAME
VALIDATE

The nice thing about this is that you do not have to use the full word when issuing a command. An example of formatting a new disk would be:

@N:disk name,ID

Note the absence of quotation marks.

That's all there is to it. Isn't that simpler than typing:

OPEN 15,8,15,"N:program name,ID":CLOSE15?

The other commands can be abbreviated by using just the **FIRST** initial or letter of the word. eg:@I will initialize the disk. To **Re-name** a program on the disk type the following: @R:new program name=old program name (Renames a program on the disk). Just that simple. Using the long way we would type:

OPEN 15,8,15,"R:new program name=old program name":
CLOSE 15

The next command is:

$0 DIRECTORY

Just as the command implies, typing ' > $' will get you the directory and it will **Not** load over a program you presently have in computer memory. If you use the long way shown earlier LOAD"$",8 you are loading the directory into memory and you will overwrite whatever is or was in memory!

Either command can be used for DIRECTORY $0 or @$0 actually you can ignore the '0' on the end unless you are using two disk drives or a dual drive with your computer. Thus you would type: $ or @$

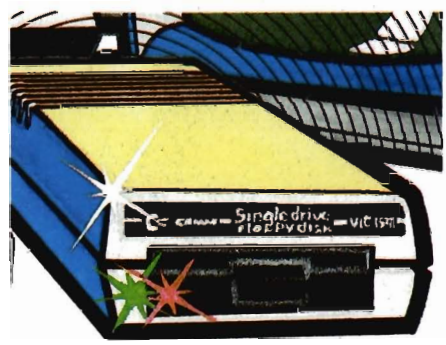
The last command is the /FILENAME LOAD. This does just what it implies, **LOADS** a program from the disk into computer memory.

Type: /(and the program or file name) and **presto** the disk springs into action, finds the file and loads it.

One exception is loading an **ABSOLUTE** file/program. That is a file which must load somewhere else in memory other than at the start of BASIC. Usually this will be a Machine language program or perhaps a utility routine that resides out of the normal basic area. Those programs will still require you to type.

LOAD"program name",8,1

There are many more disk commands. Some can be used to load individual tracks and sectors into disk memory and allow you to execute them like they were in the computer mainframe memory. The book explains these but it will take study on your part to understand how to use them. Perhaps we can cover some of these another time. **Happy Disking.**□

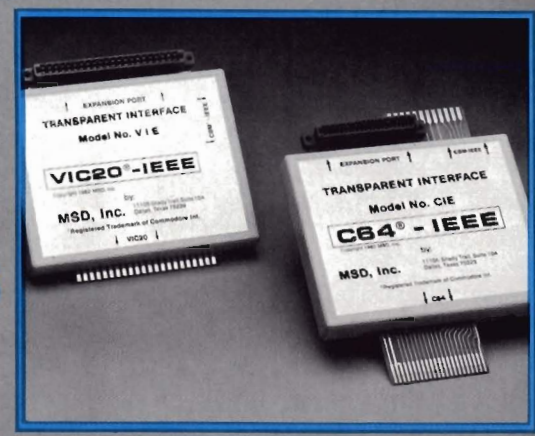


INTERBUS

INTERFACES FOR VIC-20 AND COMMODORE 64

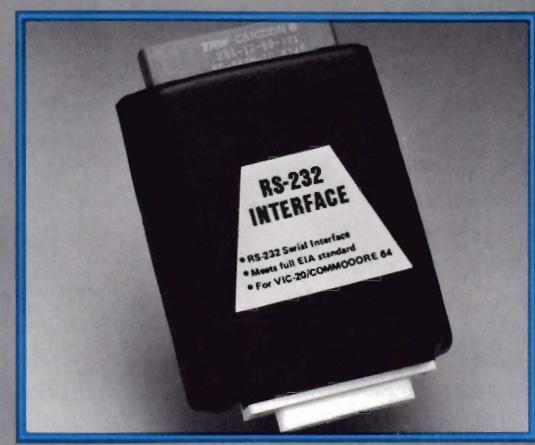
GROW WITH US!

EXPANSION PRODUCTS FOR YOUR MICROCOMPUTER



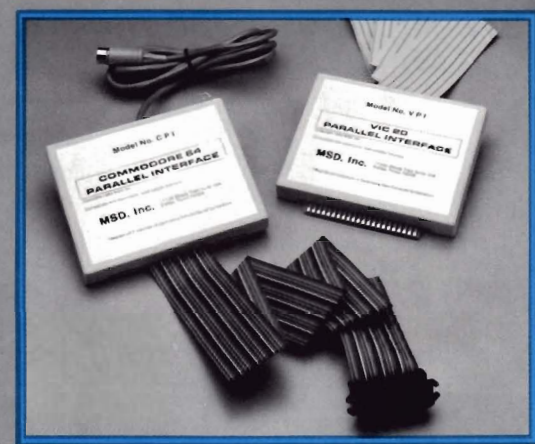
The Interbus Series includes three interfaces for the VIC-20 and Commodore 64; one for IEEE 488, one for RS-232 and one for Parallel.

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 RS-232 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 vice versa, addresses printer to device 4, 5, 6, or 7, and allows normally unprintable Commodore characters to be printed in a recognizable form.



Dealer Inquiries: 1-800-527-5285

Win a FREE trip to Hawaii. Join our "Grow With Us" Club. The more you buy the better your chances of winning. Ask your dealer for details.

MICRO SYSTEMS DEVELOPMENT, INC.
(214) 241-3743
11105 Shady Trail • Suite 104 • Dallas, Texas 75229

Screen Save: A Simple Utility

by Howard Rotenberg
Scarborough, Ontario, Canada

This is an article that will present two short and simple screen save routines that I hope you will find useful. They will both save a complete screen of text or graphics in a flash. For the best speed, the routines were done in assembler. There are many ways that you may want to utilize these routines; however, I originally used them for a fast recall of a menu driven program.

There are two programs as I mentioned before, the first called switch1.src and the second called switch2.src. They are similar in the way they work; however, they are functionally different. They are both called with a SYS 634.

Switch1.src will save or recall a screen depending on a flag that is set in the program. If the program did not save the screen the last time it was called, then it will do so. On the other hand, if the screen was saved, then it will be recalled.

Switch2.src is a little different in that it will toggle two screens back and forth. On each call to the subroutine, the current screen will be saved and the saved screen recalled.

Both the programs were originally written for a 40 column computer and then changed to be used on a 80 column. The only difference is the number of pages or blocks that will be saved or recalled. A 40 column computer has only 1K of screen RAM or 4 blocks to be concerned with, while an eighty column computer has 2K of screen RAM or 8 blocks. In the comments in the program it explains that to use this on a forty column computer you just have to change the CPX #8 to CPX #4. This is of course assuming that you are using a 2001 computer with the screen RAM, and memory addresses in the same place.

If you are using a VIC or a Commodore 64 then you will have to change the screen address where I initialize the pointers. The memory locations that I use in zero page and the load address will also have to be altered to the ones you have available to you. One other consideration will be to add a similar routine that will save the color table along with the screen. It can be done in the same way.

For those of you who do not have an assembler, you may enter the monitor with a SYS 4 and enter the code like this. I will enter the code for switch2.src, however the same method may be used for switch1.src.

```
.m 027a 02a5
.: 027a a9 00 85 01 a9 80 85 02
.: 0282 a9 00 85 b1 a9 70 85 b2
.: 028a a0 00 a2 00 b1 01 48 b1
.: 0292 b1 91 01 68 91 b1 c8 d0
.: 029a f3 e6 02 e6 b2 e8 e0 08
.: 02a2 d0 ea 60 00 00 00 00 00
.x
READY.
```

Conclusion:

I hope that you will be able to use this routine as is, or alter it to your own needs. It is short and very simple to use and if it helps any of the readers with a problem, or gives a little more insight into assembler, then this article was not written in vain. □

Program 1

```
LINE#  LOC   CODE       LINE
00001  0000             ;PUT"@0:SWITCH1.SRC"
00002  0000             ;+++++++++++++++++++++++++++++++++++++++++++++
00003  0000             ;+  PROGRAM TO STORE THE CURRENT SCREEN     +
00004  0000             ;+         AT $7000 IN MEMORY.             +
00005  0000             ;+   EACH SUCESSIVE CALL EITHER SAVES       +
00006  0000             ;+       OR RECALLS THE LAST SCREEN.        +
00007  0000             ;+   FOR A 40 COLUMN SCREEN CHANGE THE      +
00008  0000             ;+     CPX #8 TO CPX#4 ON THE SAVE AND      +
00009  0000             ;+          RESET SUBROUTINES.              +
00010  0000             ;+                                          +
00011  0000             ;+             PROGRAM BY                   +
00012  0000             ;+           HOWARD ROTENBERG               +
00013  0000             ;+           TORONTO ONTARIO                +
00014  0000             ;+++++++++++++++++++++++++++++++++++++++++++++
00015  0000             ;
00016  0000                    *=$027A          ;LOAD ADDRESS
```

```
00017  027A                    ;
00018  027A                    LOWPTR = $01            ;LOW PTR FOR CRT
00019  027A                    HIPTR  = $02            ;HIGH PTR FOR CRT
00020  027A                    SAVLOW = $B1            ;LOW PTR FOR STORING
00021  027A                    SAVHI  = $B2            ;HIGH PTR FOR STORING
00022  027A                    ;
00023  027A                    ;           INITIALIZE POINTERS
00024  027A                    ;
00025  027A  A9 00                    LDA #0          ;GET LOW BYTE OF
                                                        SCREEN ADDRESS
00026  027C  85 01                    STA LOWPTR;STORE IT
00027  027E  A9 80                    LDA #$80        ;GET HIGH BYTE OF
                                                        SCREEN ADDRESS
00028  0280  85 02                    STA HIPTR       ;STORE IT
00029  0282  A9 00                    LDA #00         ;GET LOW BYTE OF
                                                        STORAGE AREA
00030  0284  85 B1                    STA SAVLOW      ;STORE IT
00031  0286  A9 70                    LDA #$70        ;GET HIGH BYTE OF
                                                        STORAGE AREA
00032  0288  85 B2                    STA SAVHI       ;SAVE IT
00033  028A                    ;
00034  028A                    ;CHECK TO SEE IF SCREEN WAS SAVED LAST CALL
00035  028A                    ;
00036  028A  A6 21                    LDX $21         ;GET STATUS OF FLAG
00037  028C  E0 01                    CPX #1          ;WAS SCREEN SAVED
00038  028E  F0 19                    BEQ RESET       ;YES
00039  0290                    ;
00040  0290                    ;ROUTINE TO SAVE THE CURRENT SCREEN
00041  0290                    ;
00042  0290  A9 01                    LDA #1
00043  0292  85 21                    STA $21         ;STORE FLAG TO INDI
                                                        CATE SCREEN SAVED
00044  0294  A0 00                    LDY #0          ;INITIALIZE INDEX
                                                        COUNTERS
00045  0296  A2 00                    LDX #0
00046  0298  B1 01            SAVE    LDA (LOWPTR),Y  ;GET CHR FROM CRT
00047  029A  91 B1                    STA (SAVLOW),Y  ;STORE IT IN MEMORY
00048  029C  C8                      INY             ;FIRST PAGE DONE ?
00049  029D  D0 F9                    BNE SAVE
00050  029F  E6 02                    INC HIPTR       ;YES INCREMENT PTR'S
00051  02A1  E6 B2                    INC SAVHI
00052  02A3  E8                      INX
00053  02A4  E0 08                    CPX #8          ;EIGHT PAGES DONE
00054  02A6  D0 F0                    BNE SAVE        ;YES
00055  02A8  60                      RTS             ;BACK TO BASIC
00056  02A9                    ;
00057  02A9                    ;ROUTINE TO PRINT SAVED SCREEN TO CRT
00058  02A9                    ;
00059  02A9  A9 00            RESET   LDA #0          ;
00060  02AB  85 21                    STA $21         ;RESET FOR NEXT SAVE
00061  02AD  A2 00                    LDX #0          ;REINITIALIZE COUNTER
00062  02AF  A0 00                    LDY #0          ;REINITIALIZE INDEX
00063  02B1  B1 B1            GET     LDA (SAVLOW),Y  ;GET SAVED CHR
00064  02B3  91 01                    STA (LOWPTR),Y  ;WRITE TO CRT
00065  02B5  C8                      INY
00066  02B6  D0 F9                    BNE GET         ;FIRST PAGE FINISHED
```

continued on page 76

COMPUKIDS

"It's fun. It's not work. It's more like play." Five enthusiastic kids crowded around my tape recorder to talk excitedly about their experiences with computers. The kids, four boys and one girl, ages 9-11, are part of the upper level class at Compukids, a Calfornia based computer school. I visited the West Los Angeles site of Compukids for an afternoon session. The kids took charge immediately. Games and student-written programs were soon up and running. The class before had left a colorful cowboy with a perfect lariat in his hand on one screen. All the kids were eager to show what they could do.

But first, the lesson for the day. A semi-circle formed around teacher and co-founder of Compukids, Ellen Newman. The lesson was on graphics and the students paid close attention as Ms. Newman led them through commands and concepts printed on a flip chart. Her questions prompted the class members to discover the logic of the program and to visualize its operation. Next, students eagerly approached their computers to begin the process of writing code and debugging until their programs were ready to RUN.

I tried to compete with the excitement and collect some personal observations about working with computers. The youngest boy, Robert, gave the best description possible of how it is for young people the first time they sit in front of a keyboard and see their input come up on a screen:

> "First time when I came here I hardly knew what a computer was. I couldn't do anything with it. I looked at the computer and turned it on. I started letting my fingers bounce around on the keys and all this stuff started coming up on the screen and then I just wrote something. And then something weird happened. I pushed the shift button and it gave me graphics. It was neat!"

According to Dr. Julie Chan, professor of education and cofounder of Compukuds, most kids are like Robert when it comes to approaching the computer. They are fearless and open. Adults fear that they will do the wrong thing; kids jump right in.

Dr. Chan and Ms. Newman founded Compukids in 1981. Both educators, they believed they had found a learning tool that offered something to learners of all ages. They set out to create a computer literacy program that would maximize the excitement about computers and meet high educational standards. Since that first summer afternoon in July in a storefront office with twelve PETs, Chan and Newman have seen hundreds of kids and adults have their first experiences with computers. Compukids now has three locations in Southern California and franchises are being opened elsewhere in the state.

An expert in learning theory and developmental reading, Dr. Chan has seen the benefits of computer instruction for students as young as five years old. Her first piece of advice to parents is: "Get them involved early . . . as young as possible. By the time they are 12 and older, they don't do as well on the computer. They are too busy with their lives and they have the same hesitancy as adults. About age 14 is

the cut-off for the ideal time to learn about computers."

What makes it so easy for kids? Chan says, "Most kids have played computer games. When they come to Compukids or anywhere else to learn about computers, they are in a learning mode. They are open to learning. They have a mindset that programming a computer is going to be fun. They don't expect anything academic or dull."

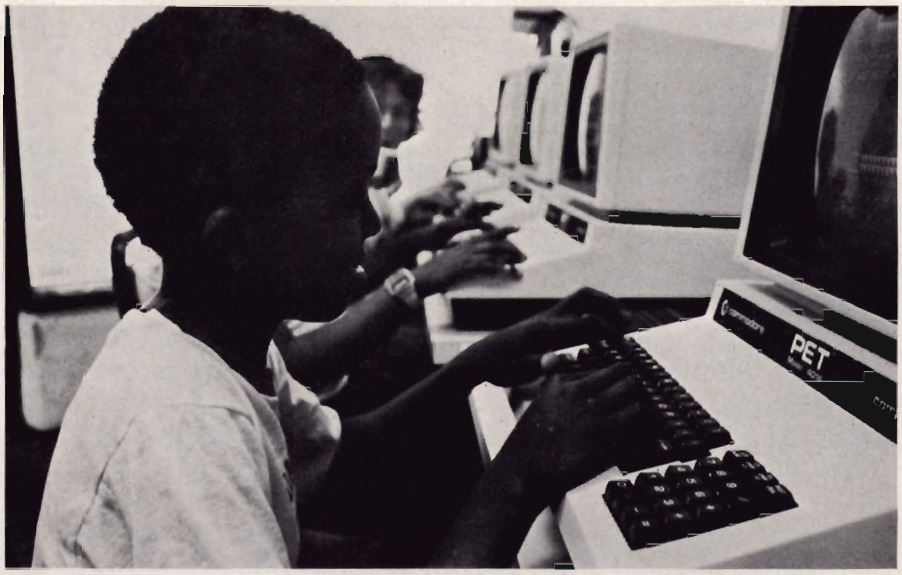
What keeps them coming back for more? Chan: "Kids often have little control over anything in their lives except the family pet. Computers have brought children and adults to the same level. When kids command a computer, they have a sense of status, power, and prestige. The kids want their parents to come in and see what they've learned."

I was curious about what kind of child does well with computer instruction. Again, Dr. Chan's experience provided the answer. "Every kid does well on computers. I had one kid brought to me that I was told to handle with kid gloves. I made up my mind to treat him just like everyone else and I was right." I met Greg, the young man Dr. Chan described. He was assisting Ellen Newman in the afternoon class! Greg, 14, had reportedly been in a learning handicapped class in school when he first came to Chan. He is in now in gifted classes in his junior high school.

"His parents say his whole sense of well-being changed when he began to learn about computers," says Chan. He's the youngest of eight kids. Suddenly he knew something no one else in the family knew."

Chan tells another story of a child diagnosed as hyperactive who sat at the computer one summer afternoon as his parent watched through the open door from her car. After the lesson she told Chan that she had never seen her child concentrate like that or sit in one place for so long a time. She was delighted.

Chan and Newman also teach classes for adults, Compufolks; for families, Compufamily; and for business people, Compubiz. The parents in these classes often report



improvement in their children's homework habits or concentration as a result of coming to computer classes.

Besides the confidence, patience, determination, and enjoyment that seem to be obvious results of learning to program, Chan offers an easy way to remember other benefits. She calls it **PIES** . . . Personal, Intellectual, Educational and Social benefits of computer instruction. Chan feels that she will someday be able to show that even a few hours of computer use can change the way a child thinks. She says that on the first day of classes, there is an observable difference between how students think when they come in and when they leave. In only four hours they have learned to plan and organize differently than they ever have before. They have developed an ability to analyze a problem while working to find errors and debug programs.

Use of computers encourages hypothesizing, estimating, and creative problem solving. Since there are several possible solutions to every computer problem, students can try things, make mistakes, and try another approach. They discover as they learn.

The computer is also a good teaching tool for reading skills. Students learn the importance of vocabulary and syntax. They learn to find main ideas and predict outcomes. They come to know the importance of attending to every detail, recognizing key words, and spelling each word correctly. Reading and following directions are essential skills for good programming.

Not only do students learn how to operate a computer and how to type. In the process, they collaborate, cooperate, and interact. They gain acceptance, prestige, and status.

Do the kids know what they're getting? As eleven-year-old Paul tells me, "There will be a lot of computers in the future . . . a lot of jobs with computers. I want to know all about it so I can be prepared and maybe get a job. Besides I like the kids here and it's fun."

Compukids is not unique in its effort to reach kids this way. There are over two hundred computer schools across the country. There are classes in schools and computer camps for summer vacation. Every program has the same goal: computer literacy for the masses. What is it? Well, despite different definitions and course objectives, an overall approach seems to lead to giving people a sense of power with the computer. Young and old alike need to know that they are in control of the computer, not slaves to it. Programming, or speaking the computer's language, is one way to gain control over the computer.

At Compukids, Dr. Chan uses the Madelyn Hunter learning model as a basis for planning curriculum and lessons. She believes that: "To teach computer literacy, you need structure and a scope and sequence just as you do for any other part of the curriculum. Left to their own devices with a computer, kids usually get so far in the tutorial and then they need help. Also, those who learn on their own often develop bad habits that can waste time later when they're programming."

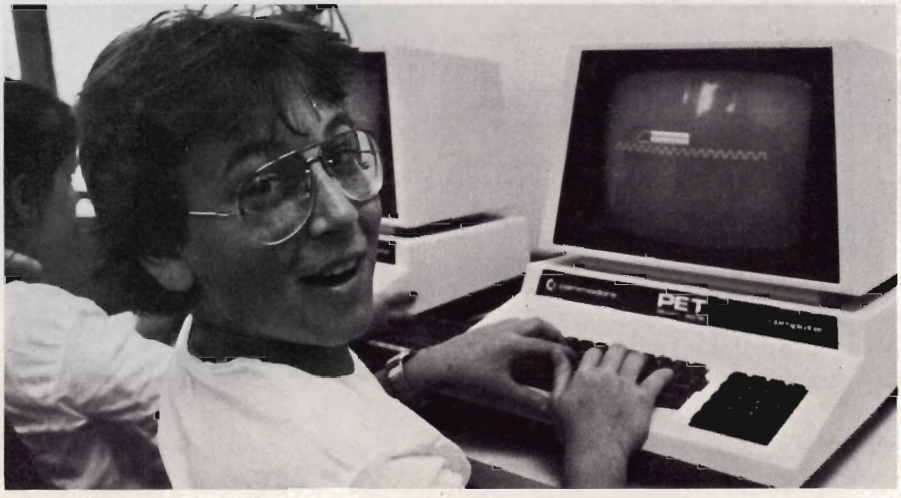
Chan feels that most computer classes taught in schools do not provide follow-up beyond the basic level of understanding. Compukids has four levels of classes and students typically are enrolled in a nine month series of courses. The approach follows a stairstepped BASIC. There is structure and follow-up. In schools, Chan suggests that computer literacy classes be taught across the curriculum as part of every class.

Classes at Compukids usually begin with a sponge activity like playing computer games or showing off software designed the day before. Some of the kids come in early to attempt to do their homework on the computers. In their very first class, they get 10 lessons, starting right on the machine. They go through a program, copy a program, make changes in the program and create their own personalized computer letter. All the activities are practical and applicable to their real lives.

I asked Dr. Chan about her use of PETs. "People who know the Commodore love it. The keyboard graphics are great. What you see is what you get! The editing function is terrific. The stop button is so easy to use. You don't have to press shift to get certain characters. The clear screen command is easier than on other machines. We find that kids do better on the Commodore than on other machines we use!"

My attention back on the kids, I saw some of the results of the day's work. Dr. Chan's advice to parents and teachers about the readiness of children to learn computers and the benefits of well-planned instruction seemed to be borne out by the kids themselves. "I love it here!," said one of them with a smile, on her way out the door. The other kids had not moved from their machines even though the class was over.

"We have to throw them out of here," says Chan with a twinkle in her eye. "Say, whose mother's got the red Mercedes? Better get on out there."□



A Problem of Sorts

by Roy MacLean and Tim Parker

Sorting, a routine problem to most programmers, is the arrangement of a series of elements in some order. Examples of sorted files are numerous. They include the telephone book, which is sorted into alphabetical order, to car license plates, which can be a mixture of both alphanumeric and numeric ordering. While sorting may be common, efficiency of the sorting procedure varies widely according to the type of sort used. A good indication of the speed and efficiency of a sort is by finding the number of comparisons required to completely sort a file. The fewer the number of comparisons, the faster the sort is likely to be.

Within the sphere of sorting techniques there is some common terminology, that will need introduction. When data processing, you sort a **file of records** by a selected **key**. An excellent example is the personnel office of a company. They maintain a **file** of data on all of the firm's employees. If you seek the information on one particular employee, then you are looking for his **record**. The employee's record will contain several **fields;** for example, name, address, telephone number, employee number, social security number, and so on. If we were to sort the file according to the employee's telephone number, then the telephone number is the **key**. If we sort the file according to the employee's name and then by the employee number, for cases where more than one employee has the same name, then we require a **compound sort key**.

There are essentially two general types of sorts. The first is where an entire file is stored in memory. These are termed **internal** sorts. When the file is large, it may not be possible to keep the records in memory. Therefore the file is kept on a storage medium, a floppy disk for example, with limited methods of access. Sorts of this type are defined as **external** sorts.

There are numerous types of sorts. For our purposes, we will concentrate on internal sorts though the methods may be applied to external sorts in most cases.

Many different sorting procedures have gained popularity for microcomputers, due to the limited memory available, and the ease of programming. The most popular sorts are the simple selection sort, exchange or bubble sort, insertion sort, and the more modern quick sort.

The selection sort is the slowest of the sorts and most closely mimics a method you might use to sort a list. It consists of searching a list for the smallest value in it. This value is then copied to another new list while it is scratched out (or replaced with a value deemed not under consideration) on the original list. The original list is then searched to find the smallest value remaining on the list and then that number is copied to the end of the new list while eliminating it in the old list. This process is repeated until the old list is emptied. The new list is now a sorted copy of the old list.

Using this method, the original list could just as easily have been sorted for a largest first (ascending) order in the new list by merely always selecting the largest remaining value in the original list.

With the selection sort, the process of selection must be repeated once for each entry in the original list. In each pass of the original list, the first entry is taken as the smallest and is compared with the remaining entries replacing it only if a smaller value is encountered. Thus the number of comparisons is (n-1) for each of n passes, where n is the number of records to be sorted. Note that for each record, this method requires another complete pass over the file when sorting it. This results in n(n-1) comparisons per sorted file where n is the number of records in the file regardless of the original order of the file. This algorithm can be improved to the point where about half of the previous comparisons are executed by using a single list and storing the partially sorted entries at the start of the same list. This is because the number of entries to scan for the smallest decreases by one after each pass.

Generally, the selection sort is used on small files or files that are not to be frequently sorted. The forte of selection process is that you need not alter the original. Thus, when using an external sort with this method, you have a backup copy of the file should the system falter during the sort. In addition, this sort is simple to write and use. So, if you are in a rush to write a sort program for your purposes, the simple selection sort is about the fastest around when it comes to programming time. Should you want to run a selection sort on a moderate to large file, start it, then get a coffee, danish, and read Ravings of a Madman.

The exchange or bubble sort involves comparing pairs of entries in the list. If the entries are out of order, then they are interchanged. This is done repeatedly until no interchanging occurs in a pass through the list being sorted.

Suppose we have an array with N elements. We now do an exchange sort on the array. What happens? First we compare Array(1) and Array(2) and arrange them in the desired order depending on if ascending or descend-

ing order is required. We will use ascending order in this example. If Array(1) is larger than Array(2), then we interchange the values, placing the value of Array(1) in Array(2) and Array(2) in Array(1). We then do the same process on Array(2) and Array(3), interchanging the values if Array(2) is greater than Array(N), arranging them so that the value in Array(N-1) is less than that at Array(N). This is repeated until one pass is completed without any interchanging occurring. This will result in the smallest element "bubbling" to the top of the list.

A loop is used to repeat the exchange/comparison process for each pair of elements in the array. We can determine the maximum number of comparisons that will occur in a particular exchange sort by noting that during the first pass, N-1 pairs must be examined and compared. Each successive pass will cause the next largest entry to move into position near the end of the list. Therefore the maximum number of pairs for every following pass can be reduced by one. Then the maximum number of comparisons that will be done is;

$$(N-1)+(N-2)+(N-3)+.....+3+2+1 = N(N-1)/2$$

The maximum number of comparisons would occur only if the smallest element in the array were located at the bottom (last element) of the array, giving it the longest possible distance to "bubble" up. However, if the array was already sorted, then this method would require only N-1 comparisons, much better than the N(N-1) for the simple selection sort.

The exchange or bubble sort is considerably quicker than the simple selection sort. The bubble sort is still rather inefficient, but maintains ease of programmability. The bubble sort is probably the most often used sort for small files (under 500 records). With large files, go for coffee and nibble on that danish.

The insertion sort requires that elements be taken sequentially from the original list and be placed immediately in their correct relative place in the new list. Suppose we have a set of elements in Oldlist of 7,5,4,6,8,7 which we will describe as Oldlist(7,5,4,6,8,7) while we have a new file which is empty, Newlist(-,-,-,-,-,-). The Newlist is first given the initial entry of 7 from Oldlist. To insert the second entry, we must shift the 7 in Newlist if we want the file in ascending order. Thus we have Newlist(-,7,-,-,-,-) after which the 5 is transferred to give Newlist(5,6,-,-,-). To insert the 4 into Newlist, we must move over the 5 and 7 to give Newlist(-,5,7,-,-,-), then transfer the 4, giving Newlist(4,5,7,-,-,-). This continues until we have the final sorted list of Newlist(4,5,6,7,7,8) where the last 7 is placed after the location of the first.

The required number of comparisons needed to determine the position for each entry will change as the size of the new list increases. However, it is reasonable to assume that you will go through half the new list for each entry. Therefore, to insert the j-th entry, you would have to do (j-1)/2 comparisons in the new list. Then for N entries in the old list, the number of comparisons would be, on average;

$$(0+1+2+3+...+(N-2)+(N-1))/2 = N(N-1)/4$$

The maximum number of comparisons would be required only if the file to be sorted is in reverse order. If the list is already sorted, then only N-1 comparisons are required. This method is an improvement over the simple selection sort and the bubble sort for most applications.

The insertion sort has one advantage similar to the selection sort. It allows for an automatic backup file should the worst happen. The biggest drawback to the insertion sort, when doing internal sorts, is having to keep two identical files in storage at the same time. It is a waste and will limit the size of file you can sort with this method. This is also true of the simple selection sort. When using the insertion sort on large to moderate files, I suggest you just get a coffee and skip the danish.

The most rapid sorting algorithm we will discuss at this time is one devised by C.A.R. Hoare. It is a recent development and is termed "Quicksort" for reasons which will become obvious as we continue. This method is a variation on the exchange/insertion idea as at each stage it succeeds in placing at least one value correctly in the final sorted list. The fact that this element is correctly positioned in the final list is used to lower the total number of comparisons needed to place future entries.

The logic behind Quicksort is to rearrange or partition the file relative to a specific entry called the **pivot**. Thus in the final partitioned list, all entries before the pivot are less than or equal to the pivot while all entries after the pivot are less than or equal the pivot. This ensures that the pivot has the correct final position in the file. The same idea is applied to the two partitions on each side of the now correctly positioned pivot. This means that the original list is gradually reduced to several smaller sublists of length one that are sorted relative to each other.

Consider the example of the series 7,5,4,6,8,7. First we choose the first 7 as the pivot. The series might be then partitioned as 4,6,5,7pivot,8,7 where everything before the pivot (7) is less than 7 and all of that after the pivot is greater than or equal to the pivot. When the list is ultimately sorted, the pivot 7 should be in exactly the same place with three entries before it and two after it. The partitioning process is then applied to the two sublists of 4;6,5 and 8,7. It is not necessary to select the first entry as the pivot element, but its choice makes the sort easier.

After the pivot entry is chosen, the quicksort algorithm scans the list from each end of the file, exchanging smaller entries near the right with larger entries on the left. When the two scans meet, the pivot is positioned. This maintains relative positioning.

To better show the partitioning taking place, we will take another larger example. Suppose we want to sort the file consisting of fifteen numeric entries. The file is composed of the following data, 58, 17, 60, 99, 25, 98, 35, 73, 50, 23, 59, 69, 76, 85, and 78, in that order. For the purpose of describing what happens, we will 'number' the numbers with a position label. The first

entry, 58 will be position 1 while the last entry, 78, will be position 15. Thus, the entry 73 would be in position 8 and so on. Initially, Quicksort chooses the value of 58 (in position 1) as the pivot value. The list is then scanned from each end. The scan from left to right searches for an entry greater than or equal to the pivot (58) while the scan from right to left on the file searches for values less than or equal to the pivot. The scan starting on the left will start at position 2 (the designated pivot is in position 1) and stop at position 3 where it finds a value of 60 which is greater than the pivot. The right to left scan starts at position 15 and scans until it stops at position 10 where the value 23 is located. We can now see that all of the entries before position 3 are less than or equal to 58 while all of those after position 10 are greater than or equal to the pivot. To maintain this trend, we exchange the values in position 3 and 10. Thus we now have 58, 17, 23, 99, 25, 98, 35, 73, 50, 60, 59, 69, 76, 85, and 78. The scans then continue with the left to right scan stopping at position 4 and the right to left scan stopping at position 9. The values of 99 and 50 are then exchanged. The scans continue again until the value of 98 is found by the left to right scan and the value of 35 is found by the right to left scan. These are then exchanged giving the file a new order of 58, 17, 23, 50, 25, 35, 98, 73, 99, 60, 59, 69, 76, 85, and 78. The scan continues, resulting in the left to right scan stopping at position 7 (value 98) and the right to left scan ending in position 6 (value 35). We now have all of the entries before position 7 less than or equal to the pivot and all of those after position 6 having a value greater than or equal to the pivot value of 58. We now exchange the pivot with position 6 as the pivot lies on the side with the values less than or equal to the pivot. Thus we now have 35, 17, 23, 50, 25, 58pivot, 98, 73, 99, 60, 59, 69, 76, 85, and 78. At this point, we have the value 58 (position 6) positioned in its final location. This now creates two partitions. One composed of entries 35, 17, 23, 50, 25 and another composed of 98, 73, 99, 60, 59, 76, 85, and 78. These partitions then follow the same process until all of the partitions are of length one or less, at which point the sort is complete.

How fast is Quicksort? The statistics required to compute the average number of comparisons is rather lengthy and boring, however, it can be shown that the average number of comparisons is about 2N(log(N)). From this, we know that Quicksort is much faster than any of the previously discussed sorts. The bad side of Quicksort is apparent when you sort an already sorted file. This is Quicksort's worst case. It requires N(N-1)/2 comparisons to sort an already sorted file, where N is the number of records in the file to be sorted. Thus, it is best to use Quicksort on poorly sorted files or completely unordered files.

Quicksort is a more involved programming task. For this reason, a wise idea is to make a backup of the file to be sorted before you run your Quicksort. This method of sorting has not yet achieved widespread use, however, it is most useful for ordering poorly sorted files regardless of their size. When running, you have just enough time to get a glance at Ravings before the file is sorted.

The final question is of course which sort to use when. For short lists of less than 30 records, the Quicksort is considerably faster. Below is a list of estimates for the efficiency of the sorts covered where N is the number of records in a file to be sorted.

FIGURE 1

Another good indication of what to choose can be found off of the table listed below. The number in the " brackets indicate the number of records to be sorted while the numbers under them indicate the number of comparisons done on the file to sort it on average.

FIGURE 2

When you are sorting a file, it is not necessary to manipulate the entire record at one time. You can just work with the key. If the file contains many fields, then it becomes important to limit the amount of data you must manipulate or your sort will slow from a fast rocket to a slow turtle. One method to use when working with records containing many fields is to maintain a list of pointers which are manipulated and are symbolic of the remainder of the records. During the actual sort, it is the pointers that are moved, since the same record may be moved many times before it arrives in its ultimate location. The pointer list will point to the original location of the record and is moved with the key field for that record. After the sort, the pointers pull the record back to the sorted field. This will result in the remaining unsorted fields combining with the key field to form a complete record, but will allow a more rapid sort. However, this is another story for another time.

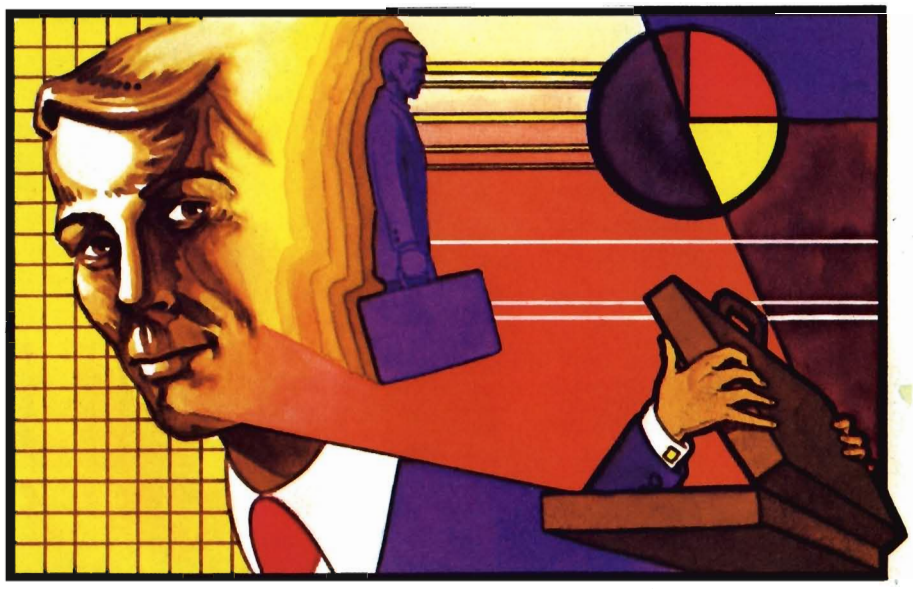
In a forthcoming article, examples of the above sorting techniques will be presented with program listings. (They are not included this month as there would be very little left in the magazine after its inclusion!) Good luck and happy sorting. □

Figure 1

Method	Average	Maximum	Minimum
Selection	N(N-1)	N(N-1)	N(N-1)
Exchange	((N(N-1)/2	N(N-1)/2	N-1
Insertion	N(N-1)/4	N(N-1)/2	N-1
Quicksort	2Log(N)	N(N-1)/2	0.51/N(N)

Figure 2

Method	16	64	256	1024
Selection	240	4032	65280	1047552
Exchange	120	2016	8128	523776
Insertion	60	243	4064	261888
Quicksort	90	224	1255	14336



Living with PractiCalc

by Colin F. Thompson
Santa Monica, CA

VisiCalc
is a registered trademark of Visicorp

Everyone **should** have a spreadsheet for his or her computer. If you don't have a spreadsheet, I'm going to tell yu what you are missing out on.

PractiCalc™ is easily the most powerful and versatile program I use. This is a program that lets you use your imagination. Most programs have specific jobs to perform. They process words, print labels, do your taxes, play games or teach your kids to spell. A good spreadsheet is not limited to a single application. Instead, it will perform just about any task you define. The key word here is define. You are in control, and the spreadsheet does the calculations you define.

Calc Wars

I bought my first PractiCalc on the weight of a friend's rave review. I wasn't expecting much. I expected a MiniCalc, but got a MaxiCalc. PractiCalc compares favorably with the grand-daddy of Calcs, VisiCalc(r). I'm very familiar with VisiCalc, and didn't expect to see anything with its power for the VIC.

Rows and Columns

The language of spreadsheets is cryptic. See the sample spreadsheet in Figure 1. The first thing to learn is the difference between Rows and Columns. Columns run vertically and are labeled by number. Rows go across the page and are labeled by letters. A blank spreadsheet is really a large sheet of blank paper awaiting your entries. By looking through the window of your TV screen, you can view any portion of the entire screen. The intersection of each row and column forms a cell. Cells can hold Titles, Labels, Data, Numbers, Bar Graphs or Formulas. By manipulating all these various entries, PractiCalc will yield answers to the problems you pose. As an example, let's use a common problem.

Checks and Balances

Anyone with a checking account already has a manual spreadsheet. It's called a Check Register. Take your checkbook out and look at the register. The top Row has a Title in each Column. The titles should be Date, Check Number, Description, Other (+ −), Amount of Deposit, Amount of Payment or Withdrawal and under Balance Forward should be your calculated balance. Each of the Rows below the Title Row hold information concerning one check or transaction. Think about what you do when you record a check. You write into separate Cells the Date, Number, Payee and Amount. Finally, you subtract the check's dollar amount from the previous balance and enter it in the far right Column. A simple spreadsheet could do the same thing.

What If?

If you have noticed the ads for various spreadsheets, you probably spotted the phrase **what if** mentioned repeatedly. **What if** accurately describes the job most spreadsheets are used for. A typical business application would have PractiCalc project sales and profits into the future, based on a fixed cost and sales rate. What if sales go up by 32% and the cost of production goes down by 15%? PractiCalc will instantly calculate the results. The spreadsheet concept has been attributed by some to be the foundation of the fabled Information Age. No management tool ever conceived can supply so many critical answers so fast.

Specs

PractiCalc is available from Computer Software Associates in three versions. The original version is PractiCalc 20. With its extra RAM memory, the 64's version is more versatile. PractiCalc 64 (P64) has a maximum capacity of 100 columns or 250 rows. The latest VIC version, PractiCalc Plus, has all the P64 features but fewer total cells available. P20 and Plus require a minimum of 16K expansion, but may use up to 24K. This will yield from 600 to 2000 cells available. All three versions have the following features:

- Available on disk and tape
- Will print a sheet to a printer
- Will print all formulas to a printer
- Sort any column on either numeric or alphanumeric data
- Prompt a cell for key entry
- Replicate any formula, data or cell format

• Perform all basic mathematic functions

• Has high level math functions—count, sum, average, square roots, logarithims, etc.

• Makes good use of the function keys, sound and color

• Fix titles on the screen

• Move a column or row

• Column width may vary between 3 and 38 characters

The Sort function is an unexpected bonus. To sort a VisiCalc® sheet you must pay at least $100 extra for the privilege. If you plan on using an RS-232 printer, order the special version of PractiCalc designed to handle that interface. Files generated on one VIC version are compatible with the other VIC version. PractiCalc is compatible with almost any printer. Several versions are available for different printers.

PractiCalc Unleashed

If you are an original owner of PractiCalc 20, you may upgrade the PractiCalc Plus for a small charge. Contact Computer Software Associates for details.

Plus and P64 have some useful enhancements. They will allow one column, anywhere in the sheet, to be a different width than the rest. Titles are right justified, making a more legible printout. Plus and P64 will Search the entire sheet for a specified letter, word or number. The results of any mathematic calculation may be displayed in a cell by a bar graph instead of the actual number. The bar may be printed on your Commodore 1515/1525 printer using the graphics mode. Other printers will print a number of asterisks equal to the resulting number. (See Figure 2)

PractiCalcly Perfect

The original PractiCalc 20 is still being sold, but I wouldn't recommend buying it. For only $10 more, PractiCalc Plus offers much more flexibility. The variable column width on Plus is worth $10 alone. Plus and P64 are $49.95 on cassette and $54.95 on disk.

In the months I've been using PractiCalc, I've found some very handy uses for it. I have replaced my Home Inventory program with PractiCalc. My Checking Account program also lies unused. Weekly travel expenses are now entered and printed in record time (without the math errors I usually cause). The best template I've done so far is a household budget planner. I really haven't made a dent in PractiCalc's abilities. Template? It's another cryptic spreadsheet term. When setting up a sheet to do a specific job, you must enter the titles, labels, data fields, and formulas in proper cells. When completed, the result is a template. Templates are like programs. They allow you to enter data to be manipulated.

On the Horizon

In the near future, CSA will release some templates for use with the PractiCalcs. CSA's Kate Nolan tells me they are also working on something called a programmable spreadsheet. I twisted her arm but she wouldn't divulge any details. When I find out more, I'll tell you about it. Watch this space. Kate is the wizard responsible for PractiCalc's great user manuals. These manuals are a picture of perfection. Someday, I hope all manuals will be as well written and illustrated as these. Special thanks go out to Sandy Ruby, Kate Nolan and Sue Robbins for their help.□

Figure 1—Sample Spreadsheet

	TER	NAME	SALES	TER.TTL	SALEDIF	3MSALES	DAYSALE	%TSALE
C	A	SMITH	500					5
D		JONES	700					7.77778
E		HOGAN	900	2100		6300	84	10
G	B	NELSON	1000					11.1111
H		PARSONS	1200					13.3313
I		ANDREWS	500	2700	600	8100	108	5.55556
K	C	ALLEN	800					8.88889
L		SHEA	400					4.44444
M		CONNORS	600	1800		5400	72	6.66667
O	D	BENNETT	900					10
P		RYAN	700					7.77778
Q		EDWARDS	800	2400	600	7200	96	9
S		12	9000				90	
T		PEOPLE	TSALES				AVDSALE	
V		2700						
W		HIGHEST						
X		1800						
Y		LOWEST						
	0	1	2	3	4	5	6	7

Figure 2—Graphics with PractiCalc

Low resolution graphics depict the histograms with a series of asterisks(*) in the appropriate columns.

	NAME	UNITS	COST
C	JONES	*****	630.00
D	SMITH	***	315.00
E	ALLEN	******	735.00
F	NELSON	*****	520.00
G	ANDREWS	**	210.00
H	RYAN	**********	940.00
J	TOTAL	32	3360.00
	0	1	2

High resolution graphics show bar graphics comprised of shaded rectangular areas that are representative of numeric quantities.

	NAME	%SALES
C	JONES	
D	SMITH	
E	ALLEN	
F	NELSON	
G	ANDREWS	
H	RYAN	
J	TOTAL	100
	O	1

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

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 shows off the sound and graphic capabilities of your VIC. PARATROOPER IS OUR #1 SELLING ARCADE GAME, you've got to see this game to believe it. **$19.95**

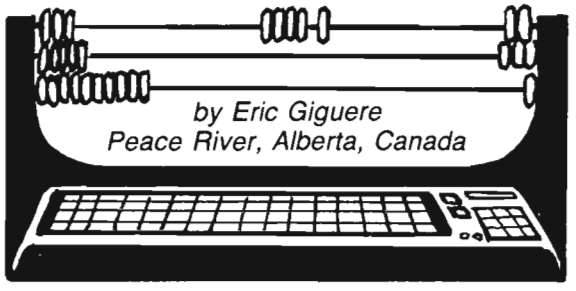
Order your copy today or see your dealer.
Will you get the top score?

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 1664T . Lake Havasu City, Arizona 86403

An Introduction to Assembly Language Programming on the Vic-20 Part VII Simple Math



by Eric Giguere
Peace River, Alberta, Canada

Over the last few months we've gone from the basic concepts of assembly language programming to loops. Now we start to get a bit more complicated as we progress to new, more powerful instructions. Math can be very useful but at the same time complicated, so we are going to approach it a piece at a time.

Simple Math

At one time in our life we have to use the math we learned at school. We may not use the more complex concepts learned in high school, but we usually do use the basic arithmetic learned in grade school. The 6502 chip in the Commodore machines can be thought of as a little pupil in the elementary grades: it uses only basic arithmetic. There are no instructions for complex functions. Instead, we have to "teach it how," using a combination of simpler instructions, to simulate those functions. This is what makes math in assembly language such a pain: what we take almost for granted on a calculator is hard to program. If you ever try disassembling the ROM math routines in your computer, you will know what I mean. Yet there are times when your basic arithmetic can come in handy, so we will take a look at these this month.

Adding

Try and stretch your memory back to your first years in school. After the basic concept of numbers and numbering systems what did they teach you? You guessed it—the concept of addition, that 1 + 1 = 2. Assembly language has an addition instruction, so this is probably the best place to start. The instruction looks like this:

```
ADC #$FF
```

ADC stands for Add with Carry and will only work with the accumulator (in fact, no math instructions work with other registers). Notice the "with Carry". Remember the C (carry) bit in the status register that we talked about before? This bit is added at along with the value you wish to add, so if the carry is 1 the actual number will be one more than expected. Unless you want this (you'll find it useful later on) you should always make sure that the carry bit is cleared (zeroed) with this instruction:

```
CLC
```

CLC stands for Clear the Carry bit, and it does so without disturbing any registers or memory locations. Once cleared, you can proceed with your addition since the carry won't make a difference anymore (adding zero is the same as not adding anything). An example is as follows:

```
CLC
LDA $00
ADC #$01
STR $00
```

Following the logic, the program first clears the carry bit, loads the accumulator from memory location $00, adds $01 (1) to it, and then stores it back in $00. ADC will work only on the accumulator itself and you have to use a STA to store your data if you so wish. ADC also has other addressing modes (as many as LDA), but remember: these are used only to get the value to be added, as such:

```
CLC
LDA $1E2F
ADC $02,X
STA $1E2F
```

This example will **not** add the value found in $1E2F to memory location $02,X but instead will add the value it finds in 02,X to the **present value in the accumulator** (which we just loaded from $1E2F). Because of this, most of the different addressing modes available are seldom used.

What happens if the result from the addition is greater than 255 ($FF)? The carry bit in the status register is set and then the byte rolls over (turns back to zero) and is incremented the remaining amount. For example, if you add $01 to $FF, the carry will be set and the result will be $00. Adding two to $FF would result in an answer of $01. This will work for any value if the total exceeds $FF. (Question: adding $FF to any value is like subtracting $01—can you figure out why?).

Two-byte additions

Adding a number to one byte is simple—just use ADC. But what if you want to add to a two-byte number? The answer lies in the carry bit. Remember that a number greater than $FF has to be stored in two bytes, with the low-order byte first and the high-order byte second. An example:

FF 01

This number reads as $01FF hex or 511 decimal (multiply high-order byte by 256 and add the low-order byte, so 1*256 + 255 = 511). If we want to properly add a value to this number we'll also have to watch out for the high-order byte ($01) and add one to this as necessary. As an example let's add one to the previous number. The number is now stored as

00 02

or $0200 (512). Two things have changed—the low-order byte rolled over from $FF to $00 and the high-order byte increased by one. If the high byte hadn't been increased when the low byte rolled over, the number would be $0100, 256 less than the true value. This is the importance of keeping a lookout for the high byte. The way you take care of this problem is as follows:

```
CLC
LDA VALLO
ADC #$01
STA VALLO
LDA VALHI
ADC #$00
STA VALHI
```

The first four lines we saw before: a value is loaded from location VALLO (the low lyte of a two-byte number), one is added to it, and then the new number is stored back in VALLO. The last three lines take care of the high-order byte: the high byte is loaded from location VALHI, zero is added, and the result is stored back in VALHI. If we are simply adding zero why should we even worry about the high byte? The secret lies in the instructions' name: add with carry. If the first addition causes a rollover, the carry bit will be set. It will then be included in the next addition unless cleared, so adding 0 with the carry bit on is the same as adding 1 (and adding with it clear is the same as adding 0). This then is the solution to our little problem. The logic is as follows:

1. Clear Carry bit.
2. Load the accumulator with the low-order byte.
3. Add to the acc. and if there is a rollover set the carry.
4. Store new low-order byte.
5. Load the acc. with the high-order byte.
6. Add zero to it in case the carry bit is set.
7. Store it back into memory.

If you follow these steps you should have no problems in your additions. If you want you can also code it a different way:

```
    . . .
    . . .
    CLC
ADDLO LDA VALLO
    ADC #$80
    STA VALLO
    BCC SKIP
ADDHI CLC
    LDA VALHI
    ADC #$01
    STA VALHI
SKIP . . .
       . . .
```

If, after adding to the lower byte, the carry is clear (meaning there was no rollover), the program will branch to SKIP. Otherwise it will clear the carry and add 1 to the high byte. It's easy to follow but very inefficient.

Subtraction

The reverse of addition is subtraction and the 6502 has an instruction for this too, called Subtract with Carry:

SBC #$02

Again, this instruction has as many addressing modes as ADC and works only on the accumulator, but has one little difference concerning the carry: instead of clearing it before subtracting, you **get it** by using the instruction SEC (Set the Carry bit). Why? Because during the subtraction SBC will also subtract the **complement** of the carry bit from the accumulator (the complement of 1 is 0 and vice-versa). This means that if the carry is clear (0) one more will be subtracted. An example of a subtraction:

```
SEC
LDA VALUE
SBC #$05
STA VALUE
```

This has the effect of subtracting 5 from the memory location VALUE. Now then, can you figure out how to subtract from a two-byte number? Write one down and check it against this:

```
SEC
LDA VALLO
SBC #$80
STA VALLO
LDA VALHI
SBC #$00
STA VALHI
```

It's just like the last three lines of the addition example except that the ADC #$00 is replaced with SBC #$00. If the low-order byte rolled over from $00 to $FF (backwards) the carry would be cleared and this would have the effect of subtracting 1 from VALHI later on (remember: you subtract the complement). Otherwise the carry would stay set and VALHI wouldn't be affected by the subtraction.

Adding and Subtracting by 1's

Before leaving there are two other instructions I wish to discuss: INC and DEC. These instructions are used to INCrement or DECrement a memory location by 1, instead of having to use ADC or SBC. They must be followed by an address and have the form

```
INC $FB
DEC $14DF,X
```

Note: INC and DEC only work on memory locations, not on the accumulator. They resemble INX, DEX, INY and DEY which affect the X- and Y-registers. Also, the status registers are affected by any changes to a memory location brought about by INC or DEC (ex: decrementing a location to 0 sets the Z flag). These instructions are useful for things that need to be incremented or decremented by regularly, such as a countdown timer or perhaps a screen color register.

Next Month

As you can see, I wasn't kidding you when I said things would get more complicated. Yet we've really only scratched the surface of math. Next month we'll go a little deeper with some multiplication and division. See you then!□

Machine Language I/O: Part Three of Three

by Howard N. Rotenberg
Toronto, Canada

This is the last part of this series of articles. In Part One we discussed the opening of a disk file and an all purpose input routine. Part Two of the article took us into the realm of a simple PET and ASCII terminal package. Now that we have come to the last part, I have included as a sample program, a CBM or ASCII terminal package that includes all the aspects we dealt with earlier. There are just a few routines that I have used that were not discussed before. The others are either the same or variations of the first routines to make them more flexible.

To just use the terminal package is quite simple since all of the instructions are included in the program and given if warranted. The terminal will talk to another computer in either PET ASCII or regular ASCII. It will also send files to or from your disk. It has the capability of pausing the transmission or receiving of files and then resuming. Lastly, it may receive an ASCII file and store it in PET format or visa-versa. An example of the instructions you would receive if you are going to use PET ASCII is the possibility of sending a sequential file or a program file. This question would not appear if you were using ASCII since you would be restricted to sequential files.

Since we have been through most of the routines in the preceding articles I will just step through the program according the remarks outlining the different routines. I will lightly mention some of the routines that have not been discussed before.

All variables and constants are declared first. I have chosen to put the program at $7000, however, if you are using a 16K computer you may change it to wherever you have 2K of room available. The buffers that will be

```
LINE# LOC   CODE        LINE
00001  0000             ;PUT"@0:IEEE.TERM.SRC"
00002  0000             ;++++++++++++++++++++++++++++++++++++++
00003  0000             ;+            IEEE TERMINAL           +
00004  0000             ;+             VERSION 1.0            +
00005  0000             ;+   IEEE MODEM PROGRAM FOR BASIC 4   +
00006  0000             ;+        40 OR 80 COLUMN CRT'S       +
00007  0000             ;+           OCTOBER 3,1982           +
00008  0000             ;+         (C) COPYRIGHT 1982         +
00009  0000             ;+    PROGRAM BY HOWARD ROTENBERG     +
00010  0000             ;++++++++++++++++++++++++++++++++++++++
00011  0000             ;
00012  0000             FNUM   = $D2          ;HOLDS FILE NUMBER
00013  0000             FNLEN  = $D1          ;HOLDS FILE NAME LENGTH
00014  0000             FNPTR  = $DA          ;FILE NAME POINTER
00015  0000             OPEN   = $F563        ;ROM DEPENDENT 2.0 = $F524
00016  0000             OPENI  = $F7AF        ;ROM DEPENDENT 2.0 = $F770
00017  0000             OPENO  = $F7FE        ;ROM DEPENDENT 2.0  =$F7BC
00018  0000             SETOUT = $FFC9        ;SET OUTPUT DEVICE
00019  0000             SETIN  = $FFC6        ;SET INPUT DEVICE
00020  0000             CLOSEA = $F2A2        ;ROM DEPENDENT 2.0 = $F26E
00021  0000             CLOSE  = $F2E2        ;ROM DEPENDENT 2.0 = $F2AE
00022  0000             DFAULT = $FFCC        ;RESTORE I/O DEVICE
00023  0000             CLEAR  = $F2A6        ;ROM DEPENDENT 2.0 = $F284 ?
00024  0000             WRITE  = $FFD2        ;PRINT A CHR
00025  0000             GETCHR = $FFE4        ;GET A CHR
00026  0000             PRMSG  = $BB1D        ;ROM DEPENDENT 2.0 = $CA1C
00027  0000             CASE   = $E84C        ;FOR UPPER OR LOWER CASE
00028  0000             DISKDS = $FFBD        ;GET DS# DISK STATUS
00029  0000             DEV    = $D4          ;DEVICE NUMBER
00030  0000             SECADR = $D3          ;SECONDARY ADDRESS
00031  0000             STATUS = $96          ;STATUS FOR FILES
00032  0000             TPBUFF = $B4          ;TAPE BUFFER LEADING CHR
00033  0000             FORMAT = $B2          ;ASCII OR PET FLAG
00034  0000             FNPTRM = $B5          ;MLM FLAG-FILE NAME PTR
00035  0000             SAVELA = $B3          ;SAVE LOGICAL ADDRESS
00036  0000             TCHPAR = $B1          ;TAPE CHARACTER PARITY
00037  0000             OUTDEV = $D0          ;FLAG FOR DISK OR NOT
00038  0000             LFEED  = $B8          ;LINE FEED FLAG
00039  0000             LASTCH = $0A          ;TEMP STORAGE FOR CHR
00040  0000             MAXCHR = $0A          ;MAXIMUM CHRS FOR FILE NAME
00041  0000             DRIVE  = $21          ;FLAG FOR DRIVE 0 OR 1
00042  0000             DISK   = $3C          ;FLAG FOR SEND TO DISK
00043  0000             DISK02 = $3D          ;PERMENENT FLAG FOR DISK READ
00044  0000             CLOSEF = $3E          ;FLAG FOR INPUT FILE CLOSED
00045  0000             DISKER = $3F          ;FLAG FOR DISK ERROR
00046  0000             DISKO  = $54          ;FLAG FOR READ FROM DISK
00047  0000             TYPE   = $55          ;FLAG FOR PRG OR SEQ FILE
00048  0000             TEMPFL = $56          ;HOLDS  FILE LENGTH
00049  0000             FLAG1  = $57          ;FLAG FOR LINE FEEDS
00050  0000             PFLAG  = $58          ;FLAG FOR PRG OR SEQ
00051  0000             TYPE2  = $59          ;FLAG FOR READ OR WRITE
00052  0000             BUF1   = $027A        ;11 CHRS LONG - $0284
00053  0000             BUF2   = $0285        ;16 CHRS LONG - $0294
00054  0000             BACK   = $9D          ;BACKSPACE
00055  0000             DEL    = $14          ;DELETE
00056  0000             SPACE  = $20          ;SPACE
00057  0000             INST   = $94          ;INSERT
00058  0000             CURSUP = $91          ;CURSOR UP
00059  0000             CUDOWN = $11          ;CURSOR DOWN
00060  0000             CULEFT = $9D          ;CURSOR LEFT
00061  0000             CRIGHT = $1D          ;CURSOR RIGHT
00062  0000             HOME   = $13          ;HOME
00063  0000             CR     = $0D          ;CARRIAGE RETURN
00064  0000             LNFEED = $0A          ;LINE FEED
00065  0000             ;
00066  0000             ;       IEEE SUBS AND LOCATIONS
00067  0000             ;
00068  0000             IEEIS  = $E821        ;CONTROL REG'R A
00069  0000             IEEI   = $E820        ;INPUT BUFFER FOR IEEE BUS
00070  0000             VIA    = $E840        ;FIRST VIA ADDRESS
00071  0000             TIH    = $E845        ;TIMER 1 HI
00072  0000             IFR    = $E84D        ;INTERUPT FLAG REGISTER
00073  0000             ;
00074  0000                    * = $7000       ;LOAD ADDRESS
```

used for the input routine are cleared right at the start. This will contain the name of the files that we may wish to send or receive. At this point I should mention that you may open both files to send as well as receive.

The signon message is printed at the top of the screen using a useful routine that only requires that you have the low byte of the address in the accumulator and the high byte in the Y register. You then just have to JSR to PRMSG which will print your message. The program is then set up for PET or ASCII and will put the screen into either graphic or upper/lower case mode respectively. If you chose ASCII then you have the option of sending linefeeds.

The next thing that is done is to determine if you wish to send a file from your disk. If you are conversing in ASCII then you may only send a sequential file, otherwise you are asked if the file is a program or sequential. If you have chosen to send a file, then the name is asked for and after it is given it is transferred from out temporary buffer to the one we utilized to put in the s,r, or ,p,r. The file is then opened using the information in our buffer as opposed to the file in the first article that was hard coded into the program. If the file is to be a PET file, then we decide if it will be sent in PET or ASCII format. Following this, if we had earlier decided to send line feeds, then we are given the option to send a line at a time rather than a continuous character string.

We now determine if we will send a file to our disk. Once again we may choose a program or sequential file, ONLY if we are using PET ASCII. The default once again for ASCII is a sequential file. The file is opened with the appropriate error checking as in the first OPEN and the appropriate instructions are then displayed.

For transmitting from disk, these two messages are displayed:

PRESS CURSOR-LEFT TO
TRANSMIT

PRESS CURSOR-RIGHT TO
PAUSE

For receiving a file, these two messages are displayed:

```
00075  7000                    ;
00076  7000                    ;       CLEAR BUF1 AND BUF2
00077  7000                    ;  27 CHRS LONG AND CONTIGUOUS
00078  7000                    ;   AND CLEAR 178 (#B2) TO 182 (#B6)
00079  7000                    ;
00080  7000  A0 00             ;       LDY #0          ;SET INDEX
00081  7002  A9 20             ;       LDA #SPACE      ;CHR TO CLEAR BUFFER
00082  7004  99 7A 02    CLR   ;       STA BUF1,Y      ;CLEAR BUFFER
00083  7007  C8               ;       INY             ;INC INDEX
00084  7008  C0 1B             ;       CPY #27         ;ALL POSITIONS CLEARED
00085  700A  D0 F8             ;       BNE CLR         ;IF NOT LOOP BACK
00086  700C  A0 00             ;       LDY #0          ;RESET INDEX
00087  700E  A9 00             ;       LDA #0          ;CHR TO CLEAR WORK AREA
00088  7010  99 B2 00    CLR2  ;       STA $B2,Y       ;CLEAR MEMORY LOCATION
00089  7013  C8               ;       INY             ;INC INDEX
00090  7014  C0 05             ;       CPY #5          ;WERE ALL POSTIONS CLEARED
00091  7016  D0 F8             ;       BNE CLR2
00092  7018                    ;
00093  7018                    ;         SIGNON MESSAGE
00094  7018                    ;
00095  7018  A9 6E             ;       LDA #<MSG17
00096  701A  A0 75             ;       LDY #>MSG17
00097  701C  20 1D BB          ;       JSR PRMSG
00098  701F  A9 8C             ;       LDA #<MSG18
00099  7021  A0 75             ;       LDY #>MSG18
00100  7023  20 1D BB          ;       JSR PRMSG
00101  7026  A9 AA             ;       LDA #<MSG19
00102  7028  A0 75             ;       LDY #>MSG19
00103  702A  20 1D BB          ;       JSR PRMSG
00104  702D                    ;
00105  702D                    ;   SET UP FOR PROGRAM ASCII OR PET
00106  702D                    ;
00107  702D  A9 00             ;       LDA #0
00108  702F  85 57             ;       STA FLAG1       ;CLEAR FLAG FOR LINE FEEDS
00109  7031  85 54             ;       STA DISK0       ;CLEAR FROM DISK FLAG
00110  7033  85 3D             ;       STA DISK02      ;CLEAR FROM DISK FLAG
00111  7035  85 3C             ;       STA DISK        ;CLEAR TO DISK FLAG
00112  7037  A9 01             ;       LDA #1
00113  7039  85 B5             ;       STA FNPTRM      ;MLM FLAG,FILE PTR
00114  703B  85 58             ;       STA PFLAG       ;SET TO 1 FOR PRG OK
00115  703D  A9 CC             ;       LDA #<MSG4      ;PRINT MESSAGE TO
00116  703F  A0 73             ;       LDY #>MSG4      ;ASK IF FORMAT IS
00117  7041  20 1D BB          ;       JSR PRMSG       ;ASCII OR PET
00118  7044  20 E4 FF    INPUT2 ;       JSR GETCHR      ;GET ANSWER
00119  7047  F0 FB             ;       BEQ INPUT2      ;IF NOTHING LOOP BACK
00120  7049  C9 50             ;       CMP #'P         ;IS IT PET
00121  704B  F0 0C             ;       BEQ UPCASE      ;(UPCASE)   PUT INTO UPPER CASE
00122  704D  C9 41             ;       CMP #'A         ;IS IT ASCII
00123  704F  D0 F3             ;       BNE INPUT2      ;WRONG LETTER ENTERED
00124  7051  A0 0E             ;       LDY #14         ;GET NUMBER TO
00125  7053  8C 4C E8          ;       STY CASE        ;PUT INTO LOWER CASE
00126  7056  4C 5E 70          ;       JMP DISP1       ;SKIP UPPER CASE
00127  7059  A0 0C       UPCASE ;       LDY #12         ;GET NUMBER TO
00128  705B  8C 4C E8          ;       STY CASE        ;PUT INTO UPPER CASE
00129  705E  20 D2 FF    DISP1  ;       JSR WRITE       ;PRINT CHR
00130  7061  C9 50             ;       CMP #'P         ;IF PET THEN
00131  7063  F0 1F             ;       BEQ SENDFL      ;GOTO SENDFILE
00132  7065  A9 01             ;       LDA #1          ;SET UP FOR ASCII
00133  7067  85 B2             ;       STA FORMAT
00134  7069                    ;
00135  7069                    ;         QUERY FOR LINE FEEDS
00136  7069                    ;
00137  7069  C6 58             ;       DEC PFLAG       ;CLEAR  NO PRG BECAUSE ASCII
00138  706B  A9 F2             ;       LDA #<MSG5      ;ASK IF
00139  706D  A0 73             ;       LDY #>MSG5      ;LINE FEEDS
00140  706F  20 1D BB          ;       JSR PRMSG       ;WANTED
00141  7072  20 E4 FF    INPUT3 ;       JSR GETCHR      ;GET ANSWER
00142  7075  F0 FB             ;       BEQ INPUT3
00143  7077  C9 4E             ;       CMP #'N         ;IF NOT
00144  7079  F0 06             ;       BEQ DISP2       ;GOTO SENDFILE
00145  707B  C9 59             ;       CMP #'Y         ;IF NOT YES
00146  707D  D0 F3             ;       BNE INPUT3      ;GO BACK FOR GOOD ANSWER
00147  707F  E6 57             ;       INC FLAG1       ;INPUT LINE FEEDS(SAVE FLAG)
00148  7081  20 D2 FF    DISP2  ;       JSR WRITE       ;PRINT CHR
00149  7084                    ;
00150  7084                    ;   DETERMINE IF SENDING FROM DISK
00151  7084                    ;
00152  7084  A9 0D       SENDFL LDA #<MSG6      ;PRINT MESSAGE TO
00153  7086  A0 74             ;       LDY #>MSG6      ;ASK IF FILE IS TO
00154  7088  20 1D BB          ;       JSR PRMSG       ;BE SENT FROM DISK
00155  708B  20 E4 FF    INPUT4 ;       JSR GETCHR      ;GET A CHR
00156  708E  F0 FB             ;       BEQ INPUT4      ;NOTHING SO LOOP BACK
00157  7090  C9 4E             ;       CMP #'N         ;IF NOT THEN
00158  7092  F0 04             ;       BEQ DISP4       ;SKIP AND PRINT CHR
00159  7094  C9 59             ;       CMP #'Y         ;YES SO SET UP
00160  7096  D0 F3             ;       BNE INPUT4      ;INVALID GO BACK FOR ANSWER
00161  7098  20 D2 FF    DISP4  ;       JSR WRITE       ;PRINT CHR
00162  709B  C9 4E             ;       CMP #'N
00163  709D  F0 34             ;       BEQ SKIP2       ;SKIP AND GOTO OUT1 (TO DISK)
00164  709F                    ;
00165  709F                    ;    SET UP FOR SENDING FROM DISK
00166  709F                    ;
00167  709F  A5 58             ;       LDA PFLAG       ;IS I/O PET OR ASCII
00168  70A1  C9 01             ;       CMP #1          ;IF PET (1)
00169  70A3  F0 07             ;       BEQ ASKPA       ;THEN BRANCH
```

continued on page 64


FROM
COMM*DATA
ARMSTRONG

PEGASUS ODYSSEY

PRESS CURSOR-UP TO ENGAGE
THE DISK
PRESS CURSOR-DOWN TO
DISENGAGE

The last two messages displayed:
PRESS INST TO QUIT
YOU ARE ON LINE

At this point the program opens the file to the modem and JSR's to the main driver routine. When it returns it will come back to this point to close all the files. The next section should be familiar since it is the input routine that we discussed in Part One. It is followed by a simple disk error routine that will display *** DISK ERROR *** if one occurs. An alternative could have been to actually take the address contained in ($0e) and the length in $0d and display the actual disk error message. I found this caused me a problem since after the message was displayed and the program would try to continue I would get a "file not open" error.

The routine is to fill the buffer for the file name with the appropriate drive number and commas, etc. It also retrieves the information in our temporary buffer that contains our file names.

This brings us to the last routines in this section that will simply open the command channel and get the drive number that is asked for when opening files.

The Driver:

At this point we go into the meat of the terminal package. I will only briefly mention the routines that are used since a full explanation would constitute another article.

The modem is set up to receive, and the cursor is turned on. If we are using PET ASCII, we will do our checking for its special characters next. The next routine is used for ASCII to PET ASCII conversion. This is a fairly standard way to do the conversions. It is not the same one that I used in the second article of this series. We now set the

MACHINE LANGUAGE—continued from page 61

```
00170  70A5  A9 53           LDA #'S          ;ELSE
00171  70A7  85 55           STA TYPE         ;MUST BE SEQ FILE
00172  70A9  4C C5 70        JMP PSKIP        ;SKIP QUERY FOR PRG OR SEQ
00173  70AC  A9 31     ASKPA LDA #<MSG7       ;PRINT MESSAGE TO
00174  70AE  A0 74           LDY #>MSG7       ;ASK TO PRG OR
00175  70B0  20 1D BB        JSR PRMSG        ;SEQ FILE
00176  70B3  20 E4 FF  INPUT5 JSR GETCHR      ;GET A CHR
00177  70B6  F0 FB           BEQ INPUT5       ;NOTHING LOOP BACK
00178  70B8  C9 50           CMP #'P          ;IS IT PET
00179  70BA  F0 04           BEQ STYPE        ;YES SAVE TYPE
00180  70BC  C9 53           CMP #'S          ;IS IT SEQ FILE
00181  70BE  D0 F3           BNE INPUT5
00182  70C0  85 55     STYPE STA TYPE         ;SAVE P OR A FOR WRITE
00183  70C2  20 D2 FF        JSR WRITE        ;PRINT CHR
00184  70C5  A9 56     PSKIP LDA #<MSG8       ;PRINT MESSAGE TO
00185  70C7  A0 74           LDY #>MSG8       ;ASK THE INPUT
00186  70C9  20 1D BB        JSR PRMSG        ;FILE NAME
00187  70CC  20 81 72        JSR INPUT        ;GET FILE NAME
00188  70CF  84 56           STY TEMPFL       ;STORE FILE LEN FROM Y
00189  70D1  C0 00           CPY #0           ;IF LENGTH = 0
00190  70D3  F0 47     SKIP2 BEQ OUT1         ;NO NAME ENTERED S GET OUT
00191  70D5  E6 54           INC DISKO        ;SET WRITE FROM DISK
00192  70D7  E6 3D           INC DISKO2       ;SET WRITE FROM DISK
00193  70D9  A9 52           LDA #'R          ;PUT R FOR READ FILE
00194  70DB  85 59           STA TYPE2        ;IN THE BUFFER
00195  70DD  20 FF 72        JSR FILLB        ;FILL BUFFER
00196  70E0  20 39 73        JSR CMDCH        ;OPEN COMMAND CHANNEL
00197  70E3                 ;
00198  70E3                 ;      OPEN OUTPUT FILE FROM DISK
00199  70E3                 ;
00200  70E3  A9 07           LDA #7           ;GET FILE NUMBER
00201  70E5  85 B3           STA SAVELA       ;SAVE L.A.
00202  70E7  85 D2           STA FNUM         ;STORE FILE NUMBER
00203  70E9  A9 08           LDA #8           ;GET DEVICE
00204  70EB  85 D4           STA DEV          ;STORE IT
00205  70ED  A9 07           LDA #7           ;STORE
00206  70EF  09 60           ORA #$60         ;SECONDARY
00207  70F1  85 D3           STA SECADR       ;ADDRESS
00208  70F3  A5 56           LDA TEMPFL       ;GET FILE LENGTH
00209  70F5  85 D1           STA FNLEN        ;STORE IT
00210  70F7  A9 85           LDA #<BUF2       ;GET LOW BYTE OF FILE NAME
00211  70F9  85 DA           STA FNPTR        ;STORE AT FILENAME POINTER
00212  70FB  A9 02           LDA #>BUF2       ;GET HIGH BYTE
00213  70FD  85 DB           STA FNPTR+1      ;AND STORE IT
00214  70FF  A9 00           LDA #0
00215  7101  85 96           STA STATUS       ;SET STATUS TO 0
00216  7103  20 63 F5        JSR OPEN         ;OPEN FILE
00217  7106  20 E5 72        JSR ERRCHK       ;CHECK FOR DISK ERROR
00218  7109  A2 07           LDX #7           ;GET FILE NUMBER
00219  710B  20 AF F7        JSR OPENI        ;OPEN FOR INPUT
00220  710E  A2 07           LDX #7
00221  7110  20 A6 F2        JSR CLEAR
00222  7113  A5 3F           LDA DISKER       ;GET DISK ERROR
00223  7115  C9 01           CMP #1           ;IF AN ERROR
00224  7117  F0 03           BEQ OUT1         ;GOTO SEND TO DISK
00225  7119  4C 23 71        JMP SKIPEX       ;SKIP EXIT
00226  711C  A0 00     OUT1  LDY #0           ;CLEAR FLAG FOR
00227  711E  84 3D           STY DISKO2       ;SEND FROM DISK
00228  7120  4C 69 71        JMP TODISK       ;GOTO SEND TO DISK
00229  7123  A5 58     SKIPEX LDA PFLAG       ;IF FLAG IS FOR
00230  7125  C9 01           CMP #1           ;THE PET THEN
00231  7127  F0 40           BEQ TODISK       ;GOTO SEND TO DISK
00232  7129  A9 79           LDA #<MSG9       ;PRINT MESSAGE TO
00233  712B  A0 74           LDY #>MSG9       ;ASK FOR FILE
00234  712D  20 1D BB        JSR PRMSG        ;FORMAT
00235  7130  20 E4 FF  INPUT6 JSR GETCHR      ;GET A CHR
00236  7133  F0 FB           BEQ INPUT6
00237  7135  C9 41           CMP #'A          ;IF ASCII THEN
00238  7137  F0 0A           BEQ DISP5        ;PRINT IT
00239  7139  C9 50           CMP #'P          ;IF NOT PET
00240  713B  D0 F3           BNE INPUT6       ;INVALID SO ASK AGAIN
00241  713D  48              PHA              ;SAVE P
00242  713E  A9 01           LDA #1           ;ELSE POKE
00243  7140  85 B6           STA FNPTRM+1     ;MLM FLAG/COUNTER/F PTR
00244  7142  68              PLA              ;RESTORE TO WRITE
00245  7143  20 D2 FF  DISP5  JSR WRITE       ;PRINT CHR
00246  7146  A5 57           LDA FLAG1        ;GET LINE FEED FLAG
00247  7148  C9 01           CMP #1           ;IF LINE FEEDS (1)
00248  714A  F0 1D           BEQ TODISK       ;THEN SKIP
00249  714C  A9 9F           LDA #<MSG10      ;PRINT MESSAGE TO
00250  714E  A0 74           LDY #>MSG10      ;ASK IF A LINE AT A
00251  7150  20 1D BB        JSR PRMSG        ;TIME SHOULD BE SENT
00252  7153  20 E4 FF  INPUT7 JSR GETCHR      ;GET A CHR
00253  7156  F0 FB           BEQ INPUT7       ;NOTHING GO BACK
00254  7158  C9 4E           CMP #'N          ;NO SO EXIT
00255  715A  F0 0A           BEQ DISP7        ;AND PRINT IT
00256  715C  C9 59           CMP #'Y          ;YES
00257  715E  D0 F3           BNE INPUT7       ;INVALID SO LOOP BACK
00258  7160  48              PHA              ;SAVE CHR
00259  7161  A9 00           LDA #0           ;POKE WITH
00260  7163  85 B5           STA FNPTRM       ;0
00261  7165  68              PLA              ;RESTORE CHR FOR PRINTING
00262  7166  20 D2 FF  DISP7  JSR WRITE       ;PRINT CHR
```

modem to be the output device and check if line feeds are being used. You must keep in mind that we are constantly jumping back to INIT to utilize the IEEE routines that take care of the bus. These routines are at the end of the program. We set up to send a file from disk now, and if the end of file is reached, we close it. The next set of routines is used to detect if any of the keys to initiate the disk commands are used, and the appropriate action is taken if the tests do not fail. The RVS key is also checked at this time to see if we want to send any control characters. Now it is time to do the PET ASCII to ASCII conversions and then we may send our characters.

The last set of subroutines is standard IEEE routines that you will probably see time and time again for manipulating the bus. These routines are outlined in the book called Programming the PET/CBM by Raeto Collin West, quite well.

I have commented the program listing quite intensively so that each routine should be almost self explanatory. I have included a cross reference of all variables and labels used in the program for easy access. I would like to give my sincere thanks to Jim Butterfield for his guidance and the use of some of his routines used in the driver of this program.

Conclusion:

Well, this brings us to the end of my Machine Language I/O articles. This, however, only brings us all to the start of getting more involved in this interesting and complex part of programming. This is something that many of you will have to get involved with when speed and precision is of the most importance. With patience and practice, there should be nothing that you cannot do in this fashion. All listings have been assembled using Commodore's assembler.□

```
00263  7169                    ;
00264  7169                    ;    DETERMINE IF FILE GOES TO DISK
00265  7169                    ;
00266  7169  A9 71     TODISK  LDA #<MSG1      ;PRINT MESSAGE TO
00267  716B  A0 73             LDY #>MSG1      ;ASK IF YOU WANT TO
00268  716D  20 1D BB          JSR PRMSG       ;SEND FILE TO DISK
00269  7170  20 E4 FF  INPUT1  JSR GETCHR      ;GET A CHR
00270  7173  F0 FB             BEQ INPUT1      ;IF NOTHING LOOP BACK
00271  7175  C9 59             CMP #'Y         ;IS IT A Y
00272  7177  F0 04             BEQ DISP3       ;GOTO  DISPLAY CHR
00273  7179  C9 4E             CMP #'N         ;IS IT NO
00274  717B  D0 F3             BNE INPUT1      ;INVALID INPUT LOOP BACK
00275  717D  20 D2 FF  DISP3   JSR WRITE       ;PRINT CHR
00276  7180  C9 59             CMP #'Y         ;IF Y
00277  7182  F0 03             BEQ PORA        ;YES SO ASK FOR A OR P
00278  7184  4C FE 71          JMP PINST       ;NO SO SKIP OPEN
00279  7187                    ;
00280  7187                    ;       ASK FOR PRG OR SEQ FILE
00281  7187                    ;
00282  7187  A5 B2     PORA    LDA FORMAT      ;IS I/O PET OR ASCII
00283  7189  C9 00             CMP #0          ;IF PET(0)
00284  718B  F0 07             BEQ ASKAP       ;THEN ASK FOR FILE TYPE
00285  718D  A9 53             LDA #'S         ;ELSE MUST BE SEQ
00286  718F  85 55             STA TYPE        ;SO STORE IT
00287  7191  4C AD 71          JMP INBUF       ;SKIP QUERY FOR PRG OR SEQ
00288  7194  A9 31     ASKAP   LDA #<MSG7      ;PRINT MESSAGE TO
00289  7196  A0 74             LDY #>MSG7      ;ASK IF PRG OR
00290  7198  20 1D BB          JSR PRMSG       ;SEQ FILE
00291  719B  20 E4 FF  INPUT8  JSR GETCHR      ;GET A CHR
00292  719E  F0 FB             BEQ INPUT8      ;NOTHING LOOP BACK
00293  71A0  C9 50             CMP #'P         ;IS IT A PRG
00294  71A2  F0 04             BEQ STYPE2      ;YES SO STORE TYPE
00295  71A4  C9 53             CMP #'S         ;IS IT A SEQ FILE
00296  71A6  D0 F3             BNE INPUT8      ;INVALID SO LOOP BACK
00297  71A8  85 55     STYPE2  STA TYPE        ;SAVE P OR S
00298  71AA  20 D2 FF          JSR WRITE       ;PRINT IT
00299  71AD                    ;
00300  71AD                    ;  GET THE INPUT FILE NAME
00301  71AD                    ;
00302  71AD  A9 93     INBUF   LDA #<MSG2      ;PRINT MESSAGE TO
00303  71AF  A0 73             LDY #>MSG2      ;ASK FOR THE
00304  71B1  20 1D BB          JSR PRMSG       ;INPUT FILE NAME
00305  71B4  20 81 72          JSR INPUT       ;GET FILE NAME
00306  71B7  84 56             STY TEMPFL      ;STORE FILE NAME LEN FROM Y
00307  71B9  C0 00             CPY #0          ;IF LENGTH = 0
00308  71BB  D0 03             BNE GETFL
00309  71BD  4C FE 71          JMP PINST       ;NO NAME ENTERED, SO EXIT
00310  71C0  A9 57     GETFL   LDA #'W         ;PUT W IN THE
00311  71C2  85 59             STA TYPE2       ;BUFFER FOR WRITE
00312  71C4  20 FF 72          JSR FILLB       ;FILL BUFFER
00313  71C7  20 39 73          JSR CMDCH       ;OPEN COMMAND CHANNEL
00314  71CA  A9 08             LDA #8          ;GET FILE NUMBER
00315  71CC  85 D2             STA FNUM        ;STORE IT
00316  71CE  85 D4             STA DEV         ;STORE DEV
00317  71D0  09 60             ORA #$60        ;8,8,8
00318  71D2  85 D3             STA SECADR
00319  71D4  A5 56             LDA TEMPFL      ;GET FILE  LENGTH
00320  71D6  85 D1             STA FNLEN       ;STORE IT
00321  71D8  A9 85             LDA #<BUF2      ;GET LOW BYTE OF FILE NAME
00322  71DA  85 DA             STA FNPTR       ;STORE AT FILENAME POINTER
00323  71DC  A9 02             LDA #>BUF2      ;GET HIGH BYTE
00324  71DE  85 DB             STA FNPTR+1     ;AND STORE IT
00325  71E0  A9 00             LDA #0
00326  71E2  85 96             STA STATUS      ;SET STATUS TO 0
00327  71E4  20 63 F5          JSR OPEN        ;OPEN FILE
00328  71E7  20 E5 72          JSR ERRCHK      ;CHECK FOR DISK ERROR
00329  71EA  A2 08             LDX #8          ;GET FILE NUMBER
00330  71EC  20 FE F7          JSR OPENO       ;OPEN DEVICE FOR OUTPUT
00331  71EF  A0 08             LDY #8          ;SET 6 IN FLAG
00332  71F1  84 3C             STY DISK        ;FOR SEND TO DISK
00333  71F3  A2 08             LDX #8          ;GET FILE NUMBER
00334  71F5  20 A6 F2          JSR CLEAR       ;AND CLEAR CHANNEL
00335  71F8  A5 3F             LDA DISKER      ;IF A DISK ERROR
00336  71FA  C9 01             CMP #1          ;THEN SKIP NEXT
00337  71FC  F0 14             BEQ PINST2      ;INSTRUCTIONS
00338  71FE                    ;
00339  71FE                    ;     PRINT INSTRUCTIONS
00340  71FE                    ;
00341  71FE  A5 3C     PINST   LDA DISK        ;IF NOT TO DISK
00342  7200  C9 00             CMP #0          ;THEN SKIP FIRST
00343  7202  F0 0E             BEQ PINST2      ;FIRST INSTRUCTIONS
00344  7204  A9 C5             LDA #<MSG11     ;PRINT MESSAGE TO
00345  7206  A0 74             LDY #>MSG11     ;PRESS CURSOR UP TO
00346  7208  20 1D BB          JSR PRMSG       ;ENGAGE DISK LOG
00347  720B  A9 EA             LDA #<MSG12     ;PRINT MESSAGE TO
00348  720D  A0 74             LDY #>MSG12     ;PRESS CURSOR DOWN
00349  720F  20 1D BB          JSR PRMSG       ;TO  DISENGAGE
00350  7212  A5 3D     PINST2  LDA DISK02      ;FIND OUT IF
00351  7214  C9 00             CMP #0          ;SEND FROM DISK
00352  7216  F0 0E             BEQ START       ;WAS INITIATED OR ERROR
00353  7218  A9 0A             LDA #<MSG13     ;PRINT MESSAGE TO
00354  721A  A0 75             LDY #>MSG13     ;PRESS CURSOR LEFT TO
00355  721C  20 1D BB          JSR PRMSG       ;SEND FROM DISK
00356  721F  A9 2A             LDA #<MSG14     ;PRINT MESSAGE TO
```

continued on page 68

Editor's Note—

We at Commander would like to extend our apologies to Mr. David A. Hook, for not giving proper acknowledgement of his review "C64-Link: Review" published in our March/April 1983 issue.

In the May '83 issue, we published a helpful hint titled "Screen Clean-up for the VIC-20 and the 64" without giving proper acknowledgement. With this in mind, we would like to give proper credit, and extend an apology to Public Domain. If you would like more information about Public Domain, please see their advertisement on page 72.□

Dear Editor—

I am a subscriber to your very informative magazine. While using my computer (VIC-20) tonight, I discovered an error in the **Joy Stick Modification** routine of the game program **Gobble!** that was featured in your January and February issues.

The corrections are as follows:

Line 9010 POKE DD,127:
P = PEEK(**P2**) and 128:
J0 = −(P = 0)

Line 9030 J1 = −((P AND 8) = 0):
J2 = −((P AND 16) = 0):
J3 = −((P AND 4) = 0):RETURN

The parts in bold face have been corrected and the modifications now run well.□

Elmer W. McKay

Dear Editor—

Congratulations on a fine publication!

I'm a new owner of a Commodore 64, and I think I can shed some light on the white flashes described by Vincent Mooney, Jr., in your Bits and Pieces section in the March/April issue.

I had a similar problem. The flashes appear as horizontal streaks near text. They are in text color. (Mine are black since I usually use a white background with black characters.) They first showed up on my screen when I started using a word processing program.

My dealer explained that this glitch only occurs in some machines when the computer executes GET statements. You can test this on your own 64 by running a program that fills the screen about half full with any text (numbers seem worse than letters) and then GETs a key from the keyboard.

Unfortunately, the only cure seems to be a new machine. If Mr. Mooney Jr's computer is still under warranty, I suggest he contact his dealer and try the exchange machine for the same glitch before accepting it.□

Sincerely,
Noel Nyman
Seattle, WA

Dear Editor—

I enjoy your magazine very much and find it very informative. I do, however, have a question. How do I get a program that was written on my 64 to list on my 8032? I would like to use the utility ROM in the 8032 to renumber and clean up in general, programs that I have written for the 64.□

Thank you,
Jerry Fellows
Box 114
Ocean City, WA 98569-0114

Dear Editor:

Reference: Review Paper Clip in March/April '83 Issue.

We would like to add our comments to the review of PaperClip by David Hook in your March/April issue. Davis has said it almost all.

However, I would like to emphasize the EXCELLENT support provided by Batteries Included, the vendors of PaperClip. I met Alan Krofchick for the first time at a Commodore Dealer Meeting in Calgary last year.

Since that time, we have sold a lot of PaperClips. Hardly a week goes by without us getting a call from Alan to inquire about any problems or questions. Any questions are dealt with almost immediately, or very shortly after if the answer isn't obvious. We also sell their disksharing Arbiter system, with equal top-of-the-line support and returns of defective units with immediate replacement.□

Kobetek Systems Limited
Sieg Deleu
President

Dear Editor—

Reference: Bits and Pieces White Flashes in March/April '83 Issue.

In response to Bits and Pieces, the white flashes reported by Vincent Mooney on his 64 may well have to do with a heating problem. If the flashes do not occur initially, but only after warm-up, the problem lies in the voltage-regulator. The early 64's exhibited this problem, and we had to build heat-sinks for the regulator to cure the problem.□

Kobetek Systems Limited
Sieg Deleu
President

Dear Readers,

As a result of questions from Mr. Florence, I entered the program as listed in the December issue of Commander, pages 43, 44 which turned up the following errors:

Line 2000 change semicolon to colon.

Line 2060 remove parenthesis before the numeral one and insert a comma after the second E$.

Line 2090 the same change as line 2060.

Line 3120 change the plus sign to a bracket.

Line 3240 the 3 UP should be enclosed in brackets.

Line 3380 add an R before the IGHT$.

Line 3395 add a minus one after the L%.

These lines should read as follows:

```
2000 PRINT"ENTER DATE IN
  FORM 08DEC82":PRINT
2060 E$=RIGHT$(H$,2):IF
  LEFT$(E$,1)="."GOTO 2080
2090 E$=RIGHT$(L$,2):IF
  LEFT$(E$,1)="."GOTO 2110
3240 D(I)=VAL(D$(I)):IF
D(I)<>D(I-1)THEN PRINT"{3
  UP}"TAB(X);M$
3380 L=(VAL(LEFT$(LS(1),J-1))+
  VAL INT(L+1):J=1
3395 IF H%=L%-1 GOTO
  3500
```

□

Sincerely,
Claud E. Cleeton

Miscellaneous—

Have you ever wanted to 'remember' the X,Y coordinates of the location where the next PRINT will occur in your VIC-20 programs? Have you ever wanted to be able to set the coordinates back again after having printed in another part of the screen? The following functions and subroutine will allow you to do this:

```
10 DEFFNA(X)=PEEK(214)*22+
  PEEK(211)
20 DEFFNX(A)=A-INT(A/22)*22
30 DEFFNY(A)=INT(A/22)-1
100 REM Your Program
110 PRINT'[clear][10 down][10
  right]';
120 X=FNA(0)
130 PRINT'[3 right][3 down]*'
140 GOSUB1000:PRINT'1':END
1000 POKE214,FNY(X):PRINT:
  POKE211,FNX(X):RETURN
```

When run, the "*" will print under the "1", where it would have printed earlier, just like you wanted it to!

Scott C. Jensen, 1359 Palace Ave., St. Paul, MN 55105.□

```
00357  7221  A0 75        LDY #>MSG14     ;PRESS CURSOR RIGHT
00358  7223  20 1D BB     JSR PRMSG       ;TO PAUSE
00359  7226  A9 47     START  LDA #<MSG15     ;PRINT MESSAGE
00360  7228  A0 75        LDY #>MSG15     ;PRESS INST
00361  722A  20 1D BB     JSR PRMSG       ;TO QUIT
00362  722D  A9 5C        LDA #<MSG16     ;PRINT MESSAGE
00363  722F  A0 75        LDY #>MSG16     ;TO INFORM THAT
00364  7231  20 1D BB     JSR PRMSG       ;USER IS ON LINE
00365  7234              ;
00366  7234              ;       OPEN MODEM
00367  7234              ;
00368  7234  A9 05        LDA #5          ;GET FILE NUMBER
00369  7236  85 D2        STA FNUM        ;STORE IT
00370  7238  85 D4        STA DEV         ;STORE DEV
00371  723A  A9 00        LDA #0          ;NO FILE NAME
00372  723C  85 D1        STA FNLEN
00373  723E  A9 FF        LDA #255        ;NO SECONDARY
00374  7240  85 D3        STA SECADR      ;ADDRESS
00375  7242  A9 00        LDA #0          ;SET STATUS
00376  7244  85 96        STA STATUS      ;TO 0
00377  7246  20 63 F5     JSR OPEN        ;OPEN MODEM
00378  7249  A5 3C        LDA DISK        ;GET DISK 8 OR 0(NO INPUT)
00379  724B  85 D0        STA OUTDEV      ;STORE AT DEVICE
00380  724D  A9 00        LDA #0          ;POKE LEN FILE NAME
00381  724F  85 D1        STA FNLEN       ;WITH 0
00382  7251  A5 57        LDA FLAG1       ;GET LINE FEED FLAG
00383  7253  85 B8        STA LFEED       ;AND STORE IT
00384  7255  20 E5 75     JSR HTERM       ;GOTO MAIN ROUTINE
00385  7258  A2 08     FCLOSE LDX #8          ;GET FILE NUMBER
00386  725A  20 A6 F2     JSR CLEAR       ;CLEAR CHANNEL
00387  725D  A9 08        LDA #8          ;GET FILE NUMBER
00388  725F  20 E2 F2     JSR CLOSE       ;CLOSE INPUT FILE
00389  7262  A2 07        LDX #7          ;GET FILE NUMBER
00390  7264  20 A6 F2     JSR CLEAR       ;CLEAR CHANNEL
00391  7267  A9 07        LDA #7          ;GET FILE NUMBER
00392  7269  20 E2 F2     JSR CLOSE       ;CLOSE FILE
00393  726C  A2 05        LDX #5          ;GET MODEM FILE
00394  726E  20 A6 F2     JSR CLEAR       ;CLEAR CHANNEL
00395  7271  A9 05        LDA #5          ;GET FILE NUMBER
00396  7273  20 E2 F2     JSR CLOSE       ;CLOSE MODEM
00397  7276  A2 0F        LDX #15         ;GET FILE NUMBER
00398  7278  20 A6 F2     JSR CLEAR       ;CLEAR CHANNEL
00399  727B  A9 0F        LDA #15         ;GET FILE NUMBER
00400  727D  20 E2 F2     JSR CLOSE       ;CLOSE FILE
00401  7280  60        GETOUT RTS             ;RETURN TO BASIC
00402  7281              ;
00403  7281              ;       INPUT ROUTINE
00404  7281              ;
00405  7281  A0 00     INPUT  LDY #0          ;INITIALIZE INDEX
00406  7283  A9 B9     PCURS  LDA #185        ;PRINT CURSOR
00407  7285  20 D2 FF     JSR WRITE       ;CHRACTER
00408  7288  A9 9D        LDA #BACK       ;POSITION CURSOR
00409  728A  20 D2 FF     JSR WRITE       ;OVER IT
00410  728D  98           TYA             ;SAVE INDEX
00411  728E  48           PHA
00412  728F  20 E4 FF     JSR GETCHR      ;GET A CHR
00413  7292  85 0A        STA LASTCH      ;STORE IT
00414  7294  68           PLA
00415  7295  A8           TAY             ;RESTORE INDEX
00416  7296  A5 0A        LDA LASTCH      ;GET CHR SAVED
00417  7298  F0 E9        BEQ PCURS       ;NOTHING LOOP BACK
00418  729A  C9 14        CMP #DEL        ;IS A DELETE
00419  729C  F0 2A        BEQ DELETE      ;GOTO DELETE ROUTINE
00420  729E  C9 0D        CMP #CR         ;A CARRAGE RETURN
00421  72A0  F0 25        BEQ FINISH      ;ENTRY FINISHED
00422  72A2  C9 22        CMP #'"         ;A QUOTE
00423  72A4  F0 DD        BEQ PCURS       ;DONT ALLOW
00424  72A6  99 7A 02     STA BUF1,Y      ;STORE FIRST CHR
00425  72A9  20 D2 FF     JSR WRITE       ;PRINT TO SCREEN
00426  72AC  C8           INY             ;INCREMENT BUFFER INDEX
00427  72AD  C0 0A        CPY #MAXCHR     ;ARE ALL ELEVEN CHRS USED
00428  72AF  F0 03        BEQ WAIT        ;YES SO WAIT FOR CR OR LAST DEL
00429  72B1  4C 83 72     JMP PCURS       ;NO SO GET ANOTHER
00430  72B4  98        WAIT   TYA             ;XFER INDEX
00431  72B5  48           PHA             ;SAVE IT
00432  72B6  20 E4 FF     JSR GETCHR      ;GET A CHR
00433  72B9  85 0A        STA LASTCH      ;STORE IT
00434  72BB  68           PLA             ;RESTORE INDEX
00435  72BC  A8           TAY             ;IN Y REGISTER
00436  72BD  A5 0A        LDA LASTCH      ;GET THE LAST CHR
00437  72BF  C9 14        CMP #DEL        ;IS IT A DELETE
00438  72C1  F0 05        BEQ DELETE      ;YES GO BACK TO DELETE ROUTINE
00439  72C3  C9 0D        CMP #CR         ;IS IT A CARRAIGE RETURN
00440  72C5  D0 ED        BNE WAIT        ;NO GO BACK FOR ANOTHER CHR
00441  72C7  60        FINISH RTS             ;RETURN TO CALLER
00442  72C8              ;
00443  72C8              ;    DELETE ROUTINE
00444  72C8              ;
00445  72C8  C0 00     DELETE CPY #00         ;ANY CHRS TO DELETE
00446  72CA  F0 B7        BEQ PCURS       ;NO SO IGNORE AND LOOP BACK
00447  72CC  A9 9D        LDA #BACK       ;GET BACKSPACE
00448  72CE  20 D2 FF     JSR WRITE       ;POSITION BACK
00449  72D1  A9 20        LDA #SPACE      ;GET SPACE
00450  72D3  20 D2 FF     JSR WRITE       ;WRITE TWICE TO
00451  72D6  20 D2 FF     JSR WRITE       ;DELETE CHR AND CURSOR
00452  72D9  A9 9D        LDA #BACK       ;GET BACKSPACE
00453  72DB  20 D2 FF     JSR WRITE       ;POSITION BACK
00454  72DE  20 D2 FF                     JSR
00455  72E1  88                          DEY
00456  72E2  4C 83 72                     JMP
00457  72E5                          ;
00458  72E5                          ;
00459  72E5                          ;
00460  72E5  A0 00              ERRCHK LDY
00461  72E7  84 3F                       STY
00462  72E9  20 BD FF                     JSR
00463  72EC  A0 00                       LDY
00464  72EE  B1 0E                       LDA
00465  72F0  C9 32                       CMP
00466  72F2  B0 01                       BCS
00467  72F4  60                          RTS
00468  72F5  A9 B6              PERROR LDA
00469  72F7  A0 73                       LDY
00470  72F9  20 1D BB                    JSR
00471  72FC  E6 3F                       INC
00472  72FE  60                          RTS
00473  72FF                          ;
00474  72FF                          ;
00475  72FF                          ;
00476  72FF  20 57 73           FILLB  JSR
00477  7302  A5 21                       LDA
00478  7304  8D 85 02                    STA
00479  7307  A9 3A                       LDA
00480  7309  8D 86 02                    STA
00481  730C  A2 00                       LDX
00482  730E  BD 7A 02           STORE  LDA
00483  7311  9D 87 02                    STA
00484  7314  E8                          INX
00485  7315  E4 56                       CPX
00486  7317  F0 03                       BEQ
00487  7319  4C 0E 73                    JMP
00488  731C  A5 56              FIN    LDA
00489  731E  A8                          TAY
00490  731F  18                          CLC
00491  7320  69 06                       ADC
00492  7322  85 56                       STA
00493  7324  A9 2C                       LDA
00494  7326  99 87 02                    STA
00495  7329  A5 55                       LDA
00496  732B  99 88 02                    STA
00497  732E  A9 2C                       LDA
00498  7330  99 89 02                    STA
00499  7333  A5 59                       LDA
00500  7335  99 8A 02                    STA
00501  7338  60                          RTS
00502  7339                          ;
00503  7339                          ;
00504  7339                          ;
00505  7339  A5 54              CMDCH  LDA
00506  733B  C9 01                       CMP
00507  733D  F0 01                       BEQ
00508  733F  60                          RTS
00509  7340  A9 0F              CMDOK  LDA
00510  7342  85 D2                       STA
00511  7344  A9 08                       LDA
00512  7346  85 D4                       STA
00513  7348  A9 0F                       LDA
00514  734A  09 60                       ORA
00515  734C  85 D3                       STA
00516  734E  A9 00                       LDA
00517  7350  85 D1                       STA
00518  7352  85 96                       STA
00519  7354  85 54                       STA
00520  7356  60                          RTS
00521  7357                          ;
00522  7357                          ;
00523  7357                          ;
00524  7357  A9 CF              GETDRV LDA
00525  7359  A0 75                       LDY
00526  735B  20 1D BB                    JSR
00527  735E  20 E4 FF           INPUT9 JSR
00528  7361  F0 FB                       BEQ
00529  7363  C9 30                       CMP
00530  7365  F0 04                       BEQ
00531  7367  C9 31                       CMP
00532  7369  D0 F3                       BNE
00533  736B  85 21              DISP6  STA
00534  736D  20 D2 FF                    JSR
00535  7370  60                          RTS
00536  7371                          ;
00537  7371                          ;
00538  7371                          ;
00539  7371  0D                 MSG1
00539  7372  53 45
00539  7392  00
00540  7393                          ;
00541  7393  0D                 MSG2
,0
00541  7394  45 4E
00541  73B4  0D
00541  73B5  00
00542  73B6                          ;
00543  73B6  0D                 MSG3
00543  73B7  0D
```

```
   WRITE        ;OVER DELETED CHR
                ;DECREMENT BUFFER INDEX
   PCURS        ;GO BACK FOR ANOTHER

    DISK ERROR ROUTINE

   #0           ;CLEAR DISK ERROR
   DISKER       ;FLAG
   DISKDS       ;GET DS$
   #0           ;GET FIRST CHR
   ($0E),Y      ;AND COMPARE IT
   #$32         ;IS IT LESS THAN 2
   PERROR       ;YES SO ERROR
                ;RETURN TO CALLER
   #<MSG3       ;PRINT A MESSAGE
   #>MSG3       ;TO INFORM USER THAT
   PRMSG        ;A DISK ERROR OCCURED
   DISKER       ;SET DISK FLAG


 FILL BUFFER WITH FILE NAME

   GETDRV       ;GET DRIVE
   DRIVE        ;RETRIEVE DRIVE NO
   BUF2+0       ;STORE IT IN IT'S PROPER
   #':          ;POSITION IN THE BUFFER
   BUF2+1       ;ALONG WITH THE ':'
   #0           ;SET BUFFER INDEX
   BUF1,X       ;GET CHR FROM INPUT BUFFER
   BUF2+2,X     ;STORE IT IN FILE BUFFER
                ;INC INDEX
   TEMPFL       ;ALL NAME TRANSFERED
   FIN          ;YES SO CONTINUE
   STORE        ;LOOP BACK FOR MORE
   TEMPFL       ; GET FILE LENGTH
                ;RESET Y INDEX

   #6           ;COMPENSATE FOR 0:,S,W OR P
   TEMPFL       ;SET PROPER FILE LENGTH
   #',          ;STORE THE ,
   BUF2+2,Y     ;IN THE CORRECT POSITION
   TYPE         ;GET P OR S
   BUF2+3,Y     ;STORE  IT
   #',          ;STORE THE ,
   BUF2+4,Y     ;AND THE R OR
   TYPE2        ;THE W IN THE
   BUF2+5,Y     ;CORRECT POSITIONS
                ;RETURN TO CALLER

 OPEN COMMAND AND DISK FILE

   DISKO        ;WASNT CHANNEL OPENED (SET TO 0)
   #1
   CMDOK        ;NO SO OPEN
                ;RETURN TO CALLER
   ##$0F        ;STORE COMMAND
   FNUM         ;CHANNEL
   #8           ;STORE DEVICE
   DEV          ;NUMBER
   ##$0F        ;STORE
   ##$60        ;SECONDARY
   SECADR       ;ADDRESS
   #0
   FNLEN        ;NO FILE NAME
   STATUS       ;SET STATUS TO 0
   DISKO        ;RESET TO 0 SO NOT REOPENED
                ;RETURN TO CALLER

 GET THE DRIVE NUMBER

   #<MSG20      ;ASK WHICH
   #>MSG20      ;DRIVE THE FILE WILL
   PRMSG        ;BE ON
   GETCHR       ;WAIT FOR CHR
   INPUT9       ;NOTHING SO LOOP BACK
   #'0          ;IS IT DRIVE 0
   DISP6        ;YES SO BRANCH
   #'1          ;IS IT DRIVE 1
   INPUT9       ;INVALID DRIVE SO ASK AGAIN
   DRIVE        ;STORE DRIVE NUMBER
   WRITE        ;PRINT IT
                ;BACK TO CALLER

        MESSAGES

.BYTE $0D,'SEND FILE TO DISK: ENTER Y OR N ',0


.BYTE $0D,'ENTER FILE NAME,10 CHRS MAXIMUM',$0D



.BYTE $0D,$0D,'*** DISK ERROR ***',$0D,0
```

```
00543  73B8  2A 2A
00543  73CA  0D
00543  73CB  00
00544  73CC
00545  73CC  0D          MSG4   .BYTE $0D,'FORMAT: ENTER A FOR ASCII P FOR PET
,0
00545  73CD  46 4F
00545  73F1  00
00546  73F2
00547  73F2  0D          MSG5   .BYTE $0D,'LINE FEEDS: ENTER Y OR N ',0
00547  73F3  4C 49
00547  740C  00
00548  740D
00549  740D  0D          MSG6   .BYTE $0D,'SEND FILE FROM DISK: ENTER Y OR N ',0
00549  740E  53 45
00549  7430  00
00550  7431
00551  7431  0D          MSG7   .BYTE $0D,'ENTER: P FOR PRG OR S FOR SEQ FILE '
0
00551  7432  45 4E
00551  7455  00
00552  7456
00553  7456  0D          MSG8   .BYTE $0D,'INPUT FILE NAME, 10 CHRS MAXIMUM',$0D
,0
00553  7457  49 4E
00553  7477  0D
00553  7478  00
00554  7479
00555  7479  0D          MSG9   .BYTE $0D,'FILE FORMAT: A FOR ASCII, P FOR PET '
,0
00555  747A  46 49
00555  749E  00
00556  749F
00557  749F  0D          MSG10  .BYTE $0D,'SEND A LINE AT A TIME: ENTER Y OR N '
,0
00557  74A0  53 45
00557  74C4  00
00558  74C5
00559  74C5  0D          MSG11  .BYTE $0D,$0D,'PRESS CURSOR-UP TO ENGAGE THE DIS
K',0
00559  74C6  0D
00559  74C7  50 52
00559  74E9  00
00560  74EA
00561  74EA  0D          MSG12  .BYTE $0D,'PRESS CURSOR-DOWN TO DISENGAGE',0
00561  74EB  50 52
00561  7509  00
00562  750A
00563  750A  0D          MSG13  .BYTE $0D,$0D,'PRESS CURSOR-LEFT TO TRANSMIT',0
00563  750B  0D
00563  750C  50 52
00563  7529  00
00564  752A
00565  752A  0D          MSG14  .BYTE $0D,'PRESS CURSOR-RIGHT TO PAUSE',0
00565  752B  50 52
00565  7546  00
00566  7547
00567  7547  0D          MSG15  .BYTE $0D,$0D,'PRESS INST TO QUIT',0
00567  7548  0D
00567  7549  50 52
00567  755B  00
00568  755C

00569  755C  0D          MSG16  .BYTE $0D,'YOU ARE ON LINE',$0D,0
00569  755D  59 4F
00569  756C  0D
00569  756D  00
00570  756E
00571  756E  93          MSG17  .BYTE $93,'          *** IEEE TERM ***',0
00571  756F  20 20
00571  758B  00
00572  758C
00573  758C  0D          MSG18  .BYTE $0D,'              (C) 1982    ',0
00573  758D  20 20
00573  75A9  00
00574  75AA
00575  75AA  0D          MSG19  .BYTE $0D,'     PROGRAM BY HOWARD ROTENBERG',$0
D,$0D,0
00575  75AB  20 20
00575  75CC  0D
00575  75CD  0D
00575  75CE  00
00576  75CF
00577  75CF  0D          MSG20  .BYTE $0D,'DRIVE: ENTER 0 OR 1 ',0
00577  75D0  44 52
00577  75E4  00
00578  75E5
00579  75E5
00580  75E5                     .FIL HTERM.SRC
                          PUT"@0:HTERM.SRC"
00581  75E5  A2 01       HTERM  LDX #$01          ;INITIAL
00582  75E7  86 3E              STX CLOSEF        ;FILE OPEN
00583  75E9  A2 00              LDX ##00          ;INITIALIZE X REG
00584  75EB  A9 00              LDA ##00          ;GET 0 AND
00585  75ED  85 B1              STA TCHPAR        ;STORE IN TAPE CHR PARITY
00586  75EF
00587  75EF                 SETUP FOR MODEM RECIEVE & FILE
00588  75EF
```

continued on page 71

YOUR PROBLEM IS SOLVED!

Now you can rely on *PACE* for ONE STOP shopping for all your Micro Computer needs. We have picked out the BEST 2000 Books, Programs and Accessories—covering all the major brands and put them into one friendly store. And, this is backed up by THOUSANDS of additional items we stock in our central warehouse, ready for overnight shipping to our stores. Magazines? You bet! We carry almost 60 different Micro Magazines on our racks! Plan to visit us soon. Can't visit? Then you can order from our gigantic catalog. Just write for your personal copy today, just $ 3.00 per copy.

pace

For: COMMODORE VIC-20*

RIVER RESCUE Thorn-EMI. Save the explorers from the jungle as you dodge a variety of hazards. 2 variations for 1 or 2 players.
4325-022001 Cartridge *SALE* **$31.95**

GRIDRUNNER HES. Avoid a variety of alien weapons while destroying the advancing legions. Multiple level.
4428-000312 Cartridge *SALE* **$31.95**

SPIDERS OF MARS UMI. You are the Martian Space Fly protecting your home from Web-throwing Martian spiders and Saturian bats, Plutonian dragonflies and Jovian hornets. 256 skill levels!
4850-001604 Cartridge *SALE* **$31.95**

AMOK UMI. Four levels of treacherous passages laced with deadly robots. Save the humans, if you're fast enough!
4850-001611 Cartridge *SALE* **$23.95**

AGGRESSOR HES. Fast paced arcade-style action in the 'Avenger' vein.
4428-000305 Cartridge *SALE* **$31.95**

INTRUDER SCRAMBLER American Peripherals. Avoid the mountains, bomb the targets and avoid the missiles. Multilevel.
4125-000428 Cassette *SALE* **$15.95**

GAME 6 PAC American Peripherals. A set of 6 games for your VIC*: Galaxy Wars; Cat Has 9 Lives; Maze of Dragons; Othello; Ambulance; and Barricade
4125-100006 Cass. (6) *SALE* **$31.95**

DEVELOP-20 French Silk Smooth Ware. The game programmer's toolkit. Includes: Book, Decoder, Editor, Assembler, Loader and Monitor. Requires minimum 5K memory.
4365-004020 Cassette/Book $49.95

MASTERING THE VIC-20 Wiley & Sons. With little knowledge of BASIC, book will teach you to write programs, make music, create pictures and learn to communicate with 6502 machine language
4925-088892 Book, 178 Pgs $14.95

VIC-20* USER GUIDE Osborne/McGraw-Hill. How to operate, including peripherals, programming, color graphics and sound, plus more!
4665-000086 Book, 388 Pgs $14.95

TYPING TUTOR Academy Software. Teach yourself to type with this easy to use, four level program.
4005-000001 Cassette $12.95

DATA MANAGER Micro Spec. Create, write and read files. You can 'browse', search and maintain with this data manager. Requires 16K memory expansion.
4538-000016 Cassette, 11K $19.95

NEW **WORDCRAFT 20** UMI. Great new, inexpensive wordprocessor for the VIC*. Needs 8K Expan.
4850-001101 Cartridge $99.95

VIC* BASIC Prentice-Hall. A user-friendly guide explains how-to-do-it. Make rainbows, music and more!
4690-008378 Book $12.95

VIC-20* PROGRAMMER'S REFERENCE GUIDE Commodore Business Machines. Complete BASIC vocabularly guide, machine language programming, tips and more.
4760-021948 Book, 290 Pgs $16.95

COMPUTE!(s) FIRST BOOK OF VIC* A compilation of articles from COMPUTE! magazine.
4105-000007 Book, 212 Pgs $12.95

KIDS AND THE VIC* Datamost. Written at children, not down to them. Turns kids, (and unsuspecting parents), into computer experts in days! Includes parent's section for help over the 'rougher' parts.
4560-000056 Book, 220 Pgs $19.95

CARDBOARD 6 Cardco. Expansion interface for the VIC-20*. Fuse protected. Will hold up to six cartridges, or up to 35K of additional RAM memory. Allows switching between up to six different games or utilities without shutting off the computer. Also, allows for future expansion by "daisy-chaining" two or more CARDBOARD 6 boards.
4135-000006 Cardboard 6 $99.95

CARDETTE 1 Cardco. Universal cassette interface for the VIC-20* and Commodore 64*. Don't throw away your old cassette player/recorder. This interface simulates all the functions of the data cassettes.
4135-000001 Cardette 1 $29.95

TURTLE GRAPHICS HES. Probably the best, fun way for the novice to learn programming. Easy-to-use language w/over 30 commands. (Many games are written in this programming system.).
4428-000303 Cartridge $39.95

SYNTHESOUND™ HES. State-of-the-art technical musical ability and special effects for the VIC-20* user. Make it a synthesizer!
4428-000306 Cartridge $59.95

6502 PROFESSIONAL DEVELOPMENT SYSTEM HES. A cassette based assembler package offering a one or two pass 6502 assembler that uses standard MOS mnemonics and operand functions.
4428-000101 Cassette $29.95

QUICK BROWN FOX™ One of the quickest, easiest to learn, user friendly word processors available. Full screen editing, supports most 80-column boards, auto reformating of edited text, single-key operation, text moving, boilerplating, tab and margins, right justification, proportional spacing & more.
4702-000100 For VIC-20* $65.00

For: COMMODORE 64*

NEW **THE ELEMENTARY 64** From Datamost. Probably THE BEST book available to date on this superb new computer. Easy to understand and master. For Commodore 64* owners everywhere!
4560-000034 224 Pages $14.95

HES WRITER 64 HES. Word processing cartridge for the Commodore 64* computer. Easy editing, preview output and word wrap-around. Save on tape or disk.
4428-000504 Cartridge $44.95

WORD MACHINE/NAME MACHINE Commodore Business Machines. Perfect easy-to-understand word processing product designed as an entry level item for home. For notes to kids, letters to friends, etc.
4100-064210 Disk $29.95

PET™ EMULATOR™ Commodore Business Machines. An emulator that will allow a high level of existing PET* software to be executed on the Commodore 64*, especially educational materials.
4100-064107 Disk $29.95

SPEECH SYNTHESIZER

TYPE-'N-TALK™ Votrax. Text to speech synthesizer. Self-contained, easy to program. Interfaces w/computer, modem or any RS-232 compatible serial device. Contains: low data rate Votrax* SC01; phoneme-based speech synthesizer CMOS chip w/unlimited vocab; and a microprocessor based text-to-speech algorithm. Operates independently. Has a one-watt audio amplifier. 750 character buffer; data switching capability; Baud (75-9600). 100-hour elevated temperature burn-in, data echo of ASCII characters. Unit requires cables, (sold below).
4900-003900 (Less Cables) $249.00

NOTE: Although TYPE-'N-TALK™ can be used with a serial printer, (on the same port), it cannot be used with a parallel printer, or on a parallel port. In addition, you MUST have the follow equipment to make it operate: 1) Special Card, as noted; 2) An RS-232 Option, or, 3) Expansion Interface AND RS-232 Card.

TYPE-'N-TALK CABLES (ONLY)

4900-001007 For VIC-20 $34.95

JOYSTICK/CONTROLS

JOYSTICK Wico Command Co. Ultimate one hand control. Bat handle. Two firing buttons. For: Atari 2600/400/800*, Sears Arcade Game, and Commodore VIC-20*.
4920-159714 *SALE* **$23.99**

RED BALL Wico Command Co. Ball handle so familiar to arcade game users. 6-leaf switch assembly. Two fire buttons. For: Atari 2600/400/800*, Sears Arcade Game, and Commodore VIC-20*.
4920-159730 *SALE* **$27.99**

TRACK BALL Wico Command Co. A phenolic ball offers the magic of 360 degree movement. Same design as the arcade games. For all Atari®, Sears® Video games, Commodore 64® and VIC-20® computers.
4920-724545 *SALE* **$55.99**

NEW NEW NEW NEW

FROGGEE !!'s easy! Just get your Froggee from the bottom of the screen to the top. Avoid the cars and trucks, hop on the logs and the leaves. Eight levels, with crocodiles, snakes and other neat stuff out to do you in! Uses Joystick. Needs no memory expansion.
4180-020001 For 3K VIC-20* (Cassette) $29.95
4180-064001 For Commodore* 64* (Cassette) $29.95

CENTIPOD Fast paced, decending bugs, falling projectiles, bouncing spiders and more! Quick reactions needed here, just to keep alive! Uses Joystick. No memory expansion needed.
4180-020002 For 3K VIC-20* (Cassette) $29.95

MOTOR MOUSE Up and down the grandfather clock, picking-up the cheese. But, watch out! There are cats hiding in the cheese! 7 progressively harder levels and a time factor to beat. Very fast paced arcade quality game. Uses Joystick. No memory expansion needed.
4180-020003 For 3K VIC-20* (Cassette) $29.95

For: COMMODORE 64*

EASYMAIL 64 Commodore Business Machines. Fully featured name and address program for business, club or organization.
4100-064204 Disk $49.95

6502/6510 PROFESSIONAL DEVELOPMENT SYSTEM HES. A cassette based assembler package. With a one or two pass assembler using standard MOS mnemonics.
4428-000102 Cassette $29.95

TURTLE GRAPHICS II HES. Advanced version of David Malmberg's program, designed for the C-64* graphics system.
4428-000503 Cartridge $59.95

HESWRITER 64 HES. Sophisticated time-saving word processor for the C-64*. Includes full screen editing, justification, centering, page headers and numbering. Compatible with all Commodore printers or any IEEE compatible printer
4428-000504 Cartridge $44.95

TINY BASIC COMPILER Abacus. Gives the benefit of a high-speed compiled language at a modest cost. Supports: IF, THEN, END, INPUT GOSUB, GOTO, LET, PRINT, REM, RETURN, STOP, USR, PEEK and 100% of floating point math and functions. Written in BASIC it is easily modifiable by user. Generates pure 6502 machine relocatable code. Can be tested, and has listing capabilities if desired. With user's manual.
4001-000177 Disk $28.00

QUICK BROWN FOX™ Word processor. See full description under VIC-20*, this ad.
4702-000101 For C-64* $65.00
(Cartridge and Cassette)

DEVELOP-64 French Silk Smooth Ware. Game programmer's toolkit. See full description under VIC-20 in this ad.
4365-064013 Cassette/Book $34.95
4365-064014 Disk/Book $39.95
NEW **PACE** *COMPUTERWARE CATALOG*
0009-198301 (Due Summer) **$ 3.00**

Just Who Is P.A.C.E.?

We want you to have confidence in buying from **PACE**, so, we think that it is important to take this opportunity to explain something about our company.

Drawing from our more than 25 years of merchandising experience, our aim is to provide microcomputer users with a ONE STOP Software Source for all your needs: SOFTWARE, BOOKS, MAGAZINES and ACCESSORIES. No longer will you have to run around to different stores, looking for what you want. It will be all in ONE place, your local **PACE Micro Software Center.**

By the time that you read this ad, our first **PACE** store in the Western Suburbs of Chicago will be open, soon to be followed by many others. Until a **PACE** store opens in your city, you may order direct from the **PACE** central warehouse with confidence, where we stock over 6,000 products.

PACE has been founded by businessmen with impeccable reputations built on over 25 years of experience in the business community and we would be most willing to provide references on request.

PACE will be concentrating on offering your a ONE STOP SOURCE for Software, Books and Accessories covering the following brands of personal computers: Apple®, Atari®, Franklin®, Commodore®, Texas Instruments®, IBM PC®, CP/M® Systems, Timex®/Sinclair® and Radio Shack®.

Our President, John Rhodebeck, demands that our stores and mail-order departments be friendly and informative to all levels of computer users, and he invites you to drop in our first store, or contact us by mail for our latest catalog.

COMMODORE 64

NEW! **The COMMODORE 64* PROGRAMMER'S REFERENCE GUIDE.** Everything you need to know to get started programming Commodore's* newest, and most versatile personal computer. Step by step guides in language that is easy to understand. Tips and a whole lot more! Our most asked for publication!
4760-022056 $19.95

COUPON Please Send Me:

QTY	NUMBER	DESCRIPTION	COST	TOTAL

Illinois Residents Please Add 6% Sales Tax.
Foreign Orders, (All outside Continental US), Add 10% Shipping (Minimum $4.00)
Catalogs Shipped Postage Paid

SHIPPING	$ 2.50
TOTAL	

PAYMENT ENCLOSED: ☐CASH ☐CHECK ☐MONEY ORDER
PLEASE CHARGE TO MY: ☐MASTERCARD ☐VISA (Min. Chg. $25)

CARD NUMBER _____
EXPIRES _____ INTRBNK# _____
SHIP TO _____
STREET ADDRESS _____ APT _____
CITY _____ STATE _____ ZIP _____

RUSH ORDER DEPART.

p.a.c.e.
DEPARTMENT: C-D
Lock Box 328
Bensenville, IL 60106

PRICES & AVAILABILITY SUBJECT TO CHANGE

General Office: 345 East Irving Park Road, Wood Dale, IL 60191
PHONE: (312) 595-0238

```
00589  75EF  A2 00       INIT   LDX #$00        ;GET 0
00590  75F1  86 A7              STX $A7         ;STORE FOR FLASHING CURSOR (1)OF
F
00591  75F3  A2 05              LDX #$05        ;GET FILE NUMBER
00592  75F5  20 C6 FF          JSR SETIN       ;SET MODEM TO INPUT DEVICE
00593  75F8  20 42 77          JSR IEEROU      ;GOSUB IEEE ROUTINES
00594  75FB  A6 96              LDX STATUS      ;GET STATUS
00595  75FD  D0 72              BNE OMODEM      ;IF NO CHR THEN GOTO MODEM
00596  75FF  A6 D1              LDX FNLEN       ;GET FILE NAME LENGTH
00597  7601  F0 0D              BEQ SUB18       ;IF NO FILE THEN BRANCH
00598  7603  48                 PHA             ;SAVE CHR
00599  7604                    ;
00600  7604                    ; SET UP DISK TO RECIEVE & CHECK FORMAT
00601  7604                    ;
30602  7604  A2 08              LDX #$08        ;GET FILE NUMBER
00603  7606  20 C9 FF          JSR SETOUT      ;SET DISK TO OUTPUT
00604  7609  20 D2 FF          JSR WRITE       ;SEND A CHR
00605  760C  20 CC FF          JSR DFAULT      ;RESTORE DEFAULT DEVICE
00606  760F  68                 PLA             ;RESTORE CHR
00607  7610  A6 B2       SUB18  LDX FORMAT      ;IS FORMAT ASCII (1)
00608  7612  F0 57              BEQ PRINT       ;NO SO BRANCH
00609  7614  29 7F              AND #$7F        ;YES SO MASK OUT 8'TH BIT
00610  7616  C9 7F              CMP #$7F        ;IS IT 127
```

IEEE.TERM.SRC.....PAGE 0013

```
LINE# LOC   CODE        LINE

00611  7618  F0 D5              BEQ INIT        ;YES SO BACK TO INIT
00612  761A  C9 1F              CMP #$1F        ;IS IT <= 31 (CTRL)
00613  761C  B0 3B              BCS FASCII      ;YES SO CHANGE TO ASCII
00614  761E                    ;
00615  761E                    ;       PET STUFF
00616  761E                    ;
00617  761E  C9 0D              CMP #CR         ;IS IT A CARRIAGE RETURN
00618  7620  D0 0B              BNE PET1        ;NO SO BRANCH
00619  7622  E6 A7              INC $A7         ;TURN OFF CURSOR
00620  7624  A9 20              LDA #$20        ;GET SPACE CHR
00621  7626  20 D2 FF          JSR WRITE       ;PRINT IT
00622  7629  A9 0D              LDA #CR         ;IS IT A CARRIAGE RETURN
00623  762B  D0 3E              BNE PRINT       ;NO SO BRANCH
00624  762D  C9 08       PET1   CMP #$08        ;IS IT A BACK SPACE
00625  762F  D0 04              BNE PET2        ;NO SO BRANCH
00626  7631  A9 14              LDA #$14        ;IS IT A DELETE
00627  7633  D0 36              BNE PRINT       ;NO SO BRANCH
00628  7635  C9 0C       PET2   CMP #$0C        ;IS IT A FORM FEED
00629  7637  D0 04              BNE PET3        ;NO SO BRANCH
00630  7639  A9 93              LDA #$93        ;GET CLEAR SCREEN CHR
00631  763B  D0 2E              BNE PRINT       ;GO AND PRINT IT
00632  763D  C9 13       PET3   CMP #HOME       ;IS IT A HOME
00633  763F  D0 08              BNE PET4        ;NO SO BRANCH
00634  7641  A9 00              LDA #$00        ;GET 0
00635  7643  85 B4              STA TPBUFF      ;TO CLEAR TAPE BUFFER
00636  7645  85 D1              STA FNLEN       ;NO FILE NAME
00637  7647  F0 A6              BEQ INIT        ;BRANCH ALWAYS
00638  7649  C5 B7       PET4   CMP $B7         ;IS FILE <> 0
00639  764B  F0 A2       TOINT  BEQ INIT        ;YES SO BRANCH
00640  764D  C9 11              CMP #CUDOWN     ;IS IT CURSOR DOWN
00641  764F  D0 9E              BNE INIT        ;NO SO BRANCH
00642  7651  A5 B3              LDA SAVELA      ;GET THE LOGICAL ADDRESS
00643  7653  85 B4              STA TPBUFF      ;STORE IT AT TAPE BUFFER
00644  7655  10 98       TOINIT BPL INIT        ;BRANCH IF LA IS 0 OR POSITIVE
00645  7657  30 96              BMI INIT        ;BRANCH IF NEGATIVE (<↑ BRANCH A
LWAYS
00646  7659                    ;
00647  7659                    ;          ASCII TO PET CONVERSION
00648  7659                    ;
00649  7659  C9 61       FASCII CMP #$61        ;<= ASCII LOWER CASE A
00650  765B  90 04              BCC CHK2        ;NO SO BRANCH
00651  765D  29 5F              AND #$5F        ;NO, CHANGE TO PET UPPER CASE
00652  765F  D0 0A              BNE PRINT       ;BRANCH ALWAYS
00653  7661  C9 41       CHK2   CMP #$41        ;<= PET UPPER CASE A
00654  7663  90 06              BCC PRINT       ;NO SO BRANCH
00655  7665  C9 5B              CMP #$5B        ; >= PET UPPER CASE Z ($5A)
00656  7667  B0 02              BCS PRINT       ;NO SO BRANCH
00657  7669  09 80              ORA #$80        ;NO, CHANGE TO PET LOWER CASE
00658  766B  20 D2 FF    PRINT  JSR WRITE       ;PRINT CHR
00659  766E  4C EF 75    JMPINT JMP INIT        ;GOTO  INIT
00660  7671                    ;
00661  7671                    ;       SET UP MODEM FOR XMIT
00662  7671                    ;
00663  7671  A2 05       OMODEM LDX #$05        ;GET FILE NUMBER
00664  7673  20 C9 FF          JSR SETOUT      ;SET MODEM TO OUTPUT DEVICE
00665  7676  2C 40 E8          BIT VIA
00666  7679  08                PHP             ;SAVE STATUS REG
00667  767A  20 CC FF          JSR DFAULT      ;RESTORE DEFAULT DEVICE
00668  767D  28                PLP             ;RESTORE STATUS REG
00669  767E  50 EE              BVC JMPINT
00670  7680  A6 B9              LDX $B9         ;GET CYCLE COUNTER
00671  7682  F0 0C              BEQ SUB19       ;IF 0 THEN BRANCH
00672  7684  A9 0A              LDA #LNFEED     ;GET LINE FEED CHR
```

continued on page 72


```
00673  7686  E4 B8          CPX LFEED       ;ARE LINE FEEDS BEING USED (1)
00674  7688  F0 02          BEQ SUB20       ;YES SO BRANCH
00675  768A  A9 7F          LDA #$7F        ;GET MASK
00676  768C  C6 B9    SUB20 DEC $B9         ;DECREMENT CYCLE COUNTER
00677  768E  10 2D          BPL SEND        ;BRANCH IF POSITIVE
00678  7690  20 E4 FF SUB19 JSR GETCHR      ;GET A CHR
00679  7693  D0 2B          BNE DSKCMD      ;NOTHING SO CHECK DISK
00680  7695  A6 B4          LDX TPBUFF      ;IF FILE ENDED
00681  7697                ;
00682  7697                ;       SET UP TO SEND FROM DISK
00683  7697                ;
00684  7697  F0 B2    SENDSK BEQ TOINT      ;THEN BRANCH
00685  7699  A6 3E          LDX CLOSEF      ;WAS FILE CLOSED
00686  769B  F0 AE          BEQ TOINT       ;YES SO SKIP
00687  769D  A2 07          LDX #$07        ;GET XMIT FILE NUMBER
00688  769F  20 C6 FF       JSR SETIN       ;SET INPUT DEVICE
00689  76A2  20 E4 FF       JSR GETCHR      ;GET A CHR
00690  76A5  48             PHA             ;SAVE IT
00691  76A6  20 CC FF       JSR DFAULT      ;RESTORE DEFAULT DEVICE
00692  76A9  68             PLA             ;RESTORE CHR
00693  76AA  A6 96          LDX STATUS      ;GET STATUS
00694  76AC  F0 0B          BEQ SUB22       ;ZERO SO BRANCH
00695  76AE  A2 00          LDX #$00        ;GET FLAG FOR NO FILE
00696  76B0  86 B3          STX SAVELA      ;STORE AT L.A
00697  76B2  86 B4          STX TPBUFF      ;STORE FILE ENDED FLAG
00698  76B4  48             PHA             ;SAVE A
00699  76B5  20 91 77       JSR CLOSE7      ;END OF FILE SO CLOSE
00700  76B8  68             PLA             ;RESTORE A
00701  76B9  A6 B6    SUB22 LDX FNPTRM+1    ;GET MLM FLAG/COUNTER/F PTR
00702  76BB  D0 30          BNE SUB29       ;IF NOT 0 BRANCH
00703  76BD  4C 22 77 SEND  JMP MODEMO       ;GOTO SEND TO MODEM
00704  76C0                ;
00705  76C0                ;     CHECK FOR DISK COMMANDS
00706  76C0                ;
00707  76C0  C9 9D    DSKCMD CMP #CULEFT     ;START TO XMIT
00708  76C2  D0 06          BNE DCMD1       ;NO CHECK FOR TODISK
00709  76C4  A5 B3          LDA SAVELA      ;GET L.A
00710  76C6  85 B4          STA TPBUFF      ;STORE AT TAPE BUFFER
00711  76C8  10 81    TOINT1 BPL TOINT      ;POSITIVE GOTO INIT
00712  76CA  C9 91    DCMD1 CMP #CURSUP     ;ENGAGAGE DISK FOR RECV
00713  76CC  D0 06          BNE DCMD2       ;NO CHECK FOR PAUSE
00714  76CE  A5 D0          LDA OUTDEV      ;GET FLAG FROM DISK 8 OR NOT 0
00715  76D0  85 D1          STA FNLEN       ;STORE AT FN LEN
00716  76D2  10 F4          BPL TOINT1      ;IF TO DISK(8) GOTO INIT
00717  76D4  C9 1D    DCMD2 CMP #CRIGHT     ;PAUSE ON XMIT FROM DISK
00718  76D6  D0 06          BNE DCMD3       ;NO CHECK FOR DISENGAGE DISK
00719  76D8  A9 00          LDA #$00        ;YES GET FLAG
00720  76DA  85 B4          STA TPBUFF      ;STORE IT AT TAPE BUFFER
00721  76DC  F0 B9    SUB24 BEQ SENDSK      ;BRANCH ALWAYS TO INIT
00722  76DE  C9 11    DCMD3 CMP #CUDOWN     ;CURSOR DOWN
00723  76E0  D0 06          BNE DCMD4       ;NO CHECK FOR INST (QUIT)
00724  76E2  A9 00          LDA #$00        ;YES STORE FLAG
00725  76E4  85 D1          STA FNLEN       ;TO DISENGAGE DISK
00726  76E6  F0 F4          BEQ SUB24       ;BRANCH ALWAYS TO INIT
00727  76E8  C9 94    DCMD4 CMP #INST       ;INST
00728  76EA  D0 01          BNE SUB29       ;NO BRANCH
00729  76EC  60             RTS             ;EXIT TO CLOSE FILES AND END
00730  76ED  A6 B2    SUB29 LDX FORMAT      ;IS IT ASCII
00731  76EF  F0 31          BEQ MODEMO      ;YES SO SEND TO MODEM
00732  76F1  A6 B1          LDX TCHPAR      ;GET VALUE FROM BEGINING
00733  76F3  F0 07          BEQ CTRL        ;ZERO SO BRANCH
00734  76F5  A8             TAY             ;SAVE IN Y REG
00735  76F6  29 1F          AND #$1F        ;MASK TOP 3 BITS,(<32 FOR CTRL)
00736  76F8  46 B1          LSR TCHPAR      ;MULTIPLY BY TWO
00737  76FA  10 26          BPL MODEMO      ;ON PLUS SEND TO MODEM
00738  76FC  C9 12    CTRL  CMP #$12        ;IS IT THE RVS KEY (CTRL)
00739  76FE  D0 05          BNE TOASC       ;NO GOTO CONVERT TO ASCII
00740  7700  E6 B1          INC TCHPAR      ;INCREMENT
00741  7702  4C EF 75       JMP INIT        ;GOTO INIT
00742  7705                ;
00743  7705                ;          PET TOO ASCII CONVERSION
00744  7705                ;
00745  7705  C9 0D    TOASC CMP #CR         ;CARRIAGE RETURN
00746  7707  F0 13          BEQ MODEMO      ;GOTO SEND TO MODEM
00747  7709  C9 41          CMP #$41        ;<= PET LOWER CASE A
00748  770B  90 06          BCC MASK        ;NO SO NO CONVERT
00749  770D  C9 5B          CMP #$5B        ;>= PET LOWER CASE Z (5A)
00750  770F  B0 02          BCS MASK        ;NO SO NO CONVERT
00751  7711  09 20          ORA #$20        ;CONVERT TO ASCII UPPER CASE
00752  7713  29 7F    MASK  AND #$7F        ;MASK OUT 7'TH BIT
00753  7715  C9 20          CMP #$20        ;IS IT <= SPACE
00754  7717  B0 09          BCS MODEMO      ;YES SO BRANCH TO SEND
00755  7719  C9 14          CMP #$14        ;IS IT A PET DELETE
00756  771B  F0 03          BEQ SKIP1       ;YES SO BRANCH TO CHANGE
00757  771D  4C 6B 76       JMP PRINT       ;GOTO PRINT CHR
00758  7720  A9 08    SKIP1 LDA #$08        ;TO ASCII BACKSPACE
00759  7722                ;
00760  7722                ;          SET UP FOR MODEM XMIT
00761  7722                ;
00762  7722  A2 05    MODEMO LDX #$05       ;GET FILE NUMBER
00763  7724  20 C9 FF       JSR SETOUT      ;SET MODEM TO OUTPUT
```

continued on page 73

```
00764  7727  85 B7           STA $B7         ;STORE FILE NUMBER
00765  7729  20 D2 FF        JSR WRITE       ;PRINT CHR
00766  772C  C9 0D           CMP #CR         ;IS IT A CARRIAGE RETURN
00767  772E  D0 0C           BNE SUB17       ;NO SO BRANCH
00768  7730  A6 B5           LDX FNPTRM      ;GET FILE PTR
00769  7732  F0 06           BEQ SUB28       ;IF ZERO STORE AT TPBUFF
00770  7734  A6 B8           LDX LFEED       ;GET FLAG FOR LINEFEEDS
00771  7736  86 B9           STX $B9         ;STORE IT CYCLE COUNT
00772  7738  10 02           BPL SUB17       ;IF LINE FEEDS (1) BRANCH
00773  773A  86 B4     SUB28 STX TPBUFF      ;STORE AT TAPE BUFF
00774  773C  20 CC FF  SUB17 JSR DFAULT      ;RESET DEFAULT DEVICE
00775  773F  4C EF 75        JMP INIT        ;GOTO INIT
00776  7742                ;
00777  7742                ;          IEEE SUBROUTINES
00778  7742                ;
00779  7742  A9 34     IEEROU LDA #$34       ;SET ATN IN LOW AND NDAC
00780  7744  8D 21 E8        STA IEEIS       ;OUT LOW, 'DATA NOT ACCEPTED'
00781  7747  AD 40 E8        LDA VIA         ;SET NRFD OUT HIGH
00782  774A  09 02           ORA #$02        ; IE. (FALSE)
00783  774C  8D 40 E8        STA VIA         ;'READY FOR DATA'
00784  774F  A9 28           LDA #$28        ;GET DEVICE FOR DISK
00785  7751  8D 45 E8        STA TIH         ;STORE IT IN TIMER
00786  7754  2C 4D E8  TIMOUT BIT IFR        ;USES VIA TIMER TO DETECT
00787  7757  70 1F           BVS NOTRDY      ;TIMEOUT 1 FOR WRITE 2 FOR READ
00788  7759  2C 40 E8        BIT VIA
00789  775C  30 F6           BMI TIMOUT
00790  775E  AD 40 E8        LDA VIA         ;SET NRFD OUT LOW
00791  7761  29 FD           AND #$FD        ;IE. (TRUE)
00792  7763  8D 40 E8        STA VIA         ;'NOT READY FOR DATA'
00793  7766  AD 20 E8  GETCH  LDA IEEI       ;GET CHR FROM INPUT BUFFER
00794  7769  49 FF           EOR #$FF        ;COMPLEMENT IT FOR USUAL FORMAT
00795  776B  48              PHA             ;SAVE CHR
00796  776C  A9 3C           LDA #$3C        ;SET ATN IN LOW & NDAC OUT
00797  776E  8D 21 E8        STA IEEIS       ;HIGH, 'DATA ACCEPTED'
00798  7771  2C 40 E8  DELAY  BIT VIA
00799  7774  10 FB           BPL DELAY
00800  7776  30 14           BMI EXIT
00801  7778  AD 40 E8  NOTRDY LDA VIA         ;SET NRFD OUT LOW
00802  777B  29 FD           AND #$FD        ;IE. (TRUE)
00803  777D  8D 40 E8        STA VIA         ;'NOT READY FOR DATA'
00804  7780  2C 40 E8        BIT VIA
00805  7783  10 E1           BPL GETCH
00806  7785  A5 96           LDA STATUS      ;GET STATUS
00807  7787  09 01           ORA #$01        ;SET FOR TIMEOUT ON WRITE
00808  7789  85 96           STA STATUS      ;AND STORE IT
00809  778B  48              PHA             ;SAVE CHR
00810  778C  20 CC FF  EXIT   JSR DFAULT      ;RESET DEFAULT DEVICE
00811  778F  68              PLA             ;RESTORE CHR
00812  7790  60              RTS             ;RETURN TO CALLER
00813  7791  A2 07     CLOSE7 LDX #7          ;GET FILE NUMBER
00814  7793  20 A6 F2        JSR CLEAR       ;CLEAR CHANNEL
00815  7796  A9 07           LDA #7          ;GET FILE NUMBER
00816  7798  20 E2 F2        JSR CLOSE       ; CLOSE IT
00817  779B  A9 00           LDA #$0
00818  779D  85 3E           STA CLOSEF      ;SET FILE TO CLOSED
00819  779F  60              RTS             ;BACK TO CALLER
00820  77A0                  .END

ERRORS = 00000

SYMBOL TABLE

SYMBOL VALUE
 ASKAP    7194    ASKPA    70AC    BACK     009D    BUF1     027A
 BUF2     0285    CASE     E84C    CHK2     7661    CLEAR    F2A6
 CLOSE    F2E2    CLOSE7   7791    CLOSEA   F2A2    CLOSEF   003E
 CLR      7004    CLR2     7010    CMDCH    7339    CMDOK    7340
 CR       000D    CRIGHT   001D    CTRL     76FC    CUDOWN   0011
 CULEFT   009D    CURSUP   0091    DCMD1    76CA    DCMD2    76D4
 DCMD3    76DE    DCMD4    76E8    DEL      0014    DELAY    7771
 DELETE   72C8    DEV      00D4    DFAULT   FFCC    DISK     003C
 DISKDS   FFBD    DISKER   003F    DISKO    0054    DISKO2   003D
 DISP1    705E    DISP2    7081    DISP3    717D    DISP4    7098
 DISP5    7143    DISP6    736B    DISP7    7166    DRIVE    0021
 DSKCMD   76C0    ERRCHK   72E5    EXIT     778C    FASCII   7659
 FCLOSE   7258    FILLB    72FF    FIN      731C    FINISH   72C7
 FLAG1    0057    FNLEN    00D1    FNPTR    00DA    FNPTRM   00B5
 FNUM     00D2    FORMAT   00B2    GETCH    7766    GETCHR   FFE4
 GETDRV   7357    GETFL    71C0    GETOUT   7280    HOME     0013
 HTERM    75E5    IEEI     E820    IEEIS    E821    IEEROU   7742
 IFR      E84D    INBUF    71AD    INIT     75EF    INPUT    7281
 INPUT1   7170    INPUT2   7044    INPUT3   7072    INPUT4   708B
 INPUT5   70B3    INPUT6   7130    INPUT7   7153    INPUT8   719B
 INPUT9   735E    INST     0094    JMPINT   766E    LASTCH   000A
 LFEED    00B8    LNFEED   000A    MASK     7713    MAXCHR   000A
 MODEMO   7722    MSG1     7371    MSG10    749F    MSG11    74C5
 MSG12    74EA    MSG13    750A    MSG14    752A    MSG15    7547
 MSG16    755C    MSG17    756E    MSG18    758C    MSG19    75AA
 MSG2     7393    MSG20    75CF    MSG3     73B6    MSG4     73CC
 MSG5     73F2    MSG6     740D    MSG7     7431    MSG8     7456
 MSG9     7479    NOTRDY   7778    OMODEM   7671    OPEN     F563
```

continued on page 74

MACHINE LANGUAGE—continued from page 73


```
SYMBOL TABLE

SYMBOL VALUE
 OPENI    F7AF    OPENO    F7FE    OUT1     711C    OUTDEV   00D0
 PCURS    7283    PERROR   72F5    PET1     762D    PET2     7635
 PET3     763D    PET4     7649    PFLAG    0058    PINST    71FE
 PINST2   7212    PORA     7187    PRINT    766B    PRMSG    BB1D
 PSKIP    70C5    SAVELA   00B3    SECADR   00D3    SEND     76BD
 SENDFL   7084    SENDSK   7697    SETIN    FFC6    SETOUT   FFC9
 SKIP1    7720    SKIP2    70D3    SKIPEX   7123    SPACE    0020
 START    7226    STATUS   0096    STORE    730E    STYPE    70C0
 STYPE2   71A8    SUB17    773C    SUB18    7610    SUB19    7690
 SUB20    768C    SUB22    76B9    SUB24    76DC    SUB28    773A
 SUB29    76ED    TCHPAR   00B1    TEMPFL   0056    TIH      E845
 TIMOUT   7754    TOASC    7705    TODISK   7169    TOINIT   7655
 TOINT    764B    TOINT1   76C8    TPBUFF   00B4    TYPE     0055
 TYPE2    0059    UPCASE   7059    VIA      E840    WAIT     72B4
 WRITE    FFD2

END OF ASSEMBLY


CROSS REFERENCE.....

  ASKAP   $7194   284   288
  ASKPA   $70AC   169   173
  BACK    $009D    54   408   447   452
  BUF1    $027A    52    82   424   482
  BUF2    $0285    53   210   212   321   323   478   480   483   494   496
                  498   500
  CASE    $E84C    27   125   128
  CHK2    $7661   650   653
  CLEAR   $F2A6    23   221   334   386   390   394   398   814
  CLOSE   $F2E2    21   388   392   396   400   816
  CLOSE7  $7791   699   813
* CLOSEA  $F2A2    20
  CLOSEF  $003E    44   582   685   818
  CLR     $7004    82    85
  CLR2    $7010    88    91
  CMDCH   $7339   196   313   505
  CMDOK   $7340   507   509
  CR      $000D    63   420   439   617   622   745   766
```

```
  CRIGHT  $001D    61   717
  CTRL    $76FC   733   738
  CUDOWN  $0011    59   640   722
  CULEFT  $009D    60   707
  CURSUP  $0091    58   712
  DCMD1   $76CA   708   712
  DCMD2   $76D4   713   717
  DCMD3   $76DE   718   722
  DCMD4   $76E8   723   727
  DEL     $0014    55   418   437
  DELAY   $7771   798   799
  DELETE  $72C8   419   438   445
  DEV     $00D4    29   204   316   370   512
  DFAULT  $FFCC    22   605   667   691   774   810
  DISK    $003C    42   111   332   341   378
  DISKDS  $FFBD    28   462
  DISKER  $003F    45   222   335   461   471
  DISKO   $0054    46   109   191   505   519
  DISKO2  $003D    43   110   192   227   350
  DISP1   $705E   126   129
  DISP2   $7081   144   148
  DISP3   $717D   272   275
  DISP4   $7098   158   161
  DISP5   $7143   238   245
  DISP6   $736B   530   533
  DISP7   $7166   255   262
  DRIVE   $0021    41   477   533
  DSKCMD  $76C0   679   707
  ERRCHK  $72E5   217   328   460
  EXIT    $778C   800   810
  FASCII  $7659   613   649
* FCLOSE  $7258   385
  FILLB   $72FF   195   312   476
  FIN     $731C   486   488
  FINISH  $72C7   421   441
  FLAG1   $0057    49   108   147   246   382
  FNLEN   $00D1    13   209   320   372   381   517   596   636   715   725
  FNPTR   $00DA    14   211   213   322   324
  FNPTRM  $00B5    34   113   243   260   701   768
  FNUM    $00D2    12   202   315   369   510
  FORMAT  $00B2    33   133   282   607   730
  GETCH   $7766   793   805
  GETCHR  $FFE4    25   118   141   155   176   235   252   269   291   412
                  432   527   678   689
  GETDRV  $7357   476   524
  GETFL   $71C0   308   310
* GETOUT  $7280   401
  HOME    $0013    62   632
  HTERM   $75E5   384   581
  IEEI    $E820    69   793
  IEEIS   $E821    68   780   797
  IEEROU  $7742   593   779
  IFR     $E84D    72   786
  INBUF   $71AD   287   302
  INIT    $75EF   589   611   637   639   641   644   645   659   741   775
  INPUT   $7281   187   305   405
  INPUT1  $7170   269   270   274
  INPUT2  $7044   118   119   123
  INPUT3  $7072   141   142   146
  INPUT4  $708B   155   156   160
  INPUT5  $70B3   176   177   181
  INPUT6  $7180   235   236   240
  INPUT7  $7153   252   253   257
  INPUT8  $719B   291   292   296
  INPUT9  $735E   527   528   532
  INST    $0094    57   727
  JMPINT  $766E   659   669
  LASTCH  $000A    39   413   416   433   436
  LFEED   $00B8    38   383   673   770
  LNFEED  $000A    64   672
  MASK    $7713   748   750   752
  MAXCHR  $000A    40   427
  MODEMD  $7722   703   731   737   746   754   762
  MSG1    $7371   266   267   539
  MSG10   $749F   249   250   557
  MSG11   $74C5   344   345   559
  MSG12   $74EA   347   348   561
  MSG13   $750A   353   354   563
  MSG14   $752A   356   357   565
  MSG15   $7547   359   360   567
  MSG16   $755C   362   363   569
  MSG17   $756E    95    96   571
  MSG18   $758C    98    99   573
  MSG19   $75AA   101   102   575
  MSG2    $7393   302   303   541
  MSG20   $75CF   524   525   577
  MSG3    $73B6   468   469   543
  MSG4    $73CC   115   116   545
  MSG5    $73F2   138   139   547
  MSG6    $740D   152   153   549
  MSG7    $7431   173   174   288   289   551
  MSG8    $7456   184   185   553
  MSG9    $7479   232   238   555
  NOTRDY  $7778   787   801
```

continued on page 76

MACHINE LANGUAGE—continued from page 75


```
OMODEM $7671   595   663
OPEN   $F563    15   216   327   377
OPENI  $F7AF    16   219
OPENO  $F7FE    17   330
OUT1   $711C   190   224   226
OUTDEV $00D0    37   379   714
PCURS  $7283   406   417   423   429   446   456
PERROR $72F5   466   468
PET1   $762D   618   624
PET2   $7635   625   628
PET3   $763D   629   632
PET4   $7649   633   638
PFLAG  $0058    50   114   137   167   229
PINST  $71FE   278   309   341
PINST2 $7212   337   343   350
PORA   $7187   277   282
PRINT  $766B   608   623   627   631   652   654   656   658   757
PRMSG  $BB1D    26    97   100   103   117   140   154   175   186   234
               251   268   290   304   346   349   355   358   361   364
               470   526
PSKIP  $70C5   172   184
SAVELA $00B3    35   201   642   696   709
SECADR $00D3    30   207   318   374   515
SEND   $76BD   677   703
SENDFL $7084   131   152
SENDSK $7697   684   721
SETIN  $FFC6    19   592   688
SETOUT $FFC9    18   603   664   763
SKIP1  $7720   756   758
SKIP2  $70D3   163   190
SKIPEX $7123   225   229
SPACE  $0020    56    81   449
START  $7226   352   359
STATUS $0096    31   215   326   376   518   594   693   806   808
STORE  $730E   482   487
STYPE  $70C0   179   182
STYPE2 $71A8   294   297
SUB17  $773C   767   772   774
SUB18  $7610   597   607
SUB19  $7690   671   678
SUB20  $768C   674   676
SUB22  $76B9   694   701
SUB24  $76DC   721   726
SUB28  $773A   769   773
SUB29  $76ED   702   728   730
TCHPAR $00B1    36   585   732   736   740
TEMPFL $0056    48   188   208   306   319   485   488   492
TIH    $E845    71   785
TIMOUT $7754   786   789
TOASC  $7705   739   745
TODISK $7169   228   231   248   266
* TOINIT $7655   644
TOINT  $764B   639   684   686   711
TOINT1 $76C8   711   716
TPBUFF $00B4    32   635   643   680   697   710   720   773
TYPE.  $0055    47   171   182   286   297   495
TYPE2  $0059    51   194   311   499
UPCASE $7059   121   127
VIA    $E840    70   665   781   783   788   790   792   798   801   803
               804
WAIT   $72B4   428   430   440
WRITE  $FFD2    24   129   148   161   183   245   262   275   298   407
               409   425   448   450   451   453   454   534   604   621
               658   765
```

SCREEN SAVE—continued from page 47


```
LINE# LOC    CODE        LINE
00067  02B8  E6 02       INC HIPTR       ;YES, INCREMENT PTR'S
00068  02BA  E6 B2       INC SAVHI
00069  02BC  E8          INX
00070  02BD  E0 08       CPX #8          ;EIGHT PAGES DONE
00071  02BF  D0 F0       BNE GET         ;NO
00072  02C1  60          RTS             ;BACK TO BASIC
00073  02C2               .END
ERRORS = 000000
SYMBOL TABLE
SYMBOL VALUE
GET      02B1     HIPTR     0002     LOWPTR    0001     RESET     02A9
```

continued on page 77

SCREEN SAVE—continued from page 72

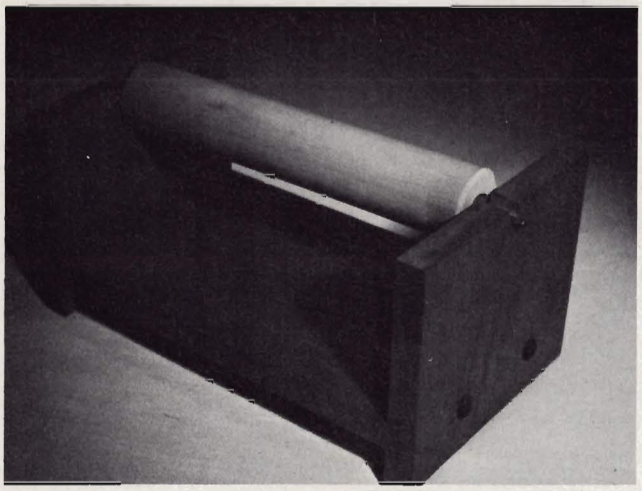
```
 SAVE     0298    SAVHI    00B2    SAVLOW   00B1
END OF ASSEMBLY

SWITCH2.SRC......PAGE 0001

LINE# LOC   CODE        LINE

00001  0000          ;PUT"@0:SWITCH2.SRC
00002  0000          ;++++++++++++++++++++++++++++++++++++++++++++++++++++
00003  0000          ;+                PROGRAM SWITCH2                   +
00004  0000          ;+ PROGRAM TO TOGGLE TWO SCREENS BACK AND FORTH     +
00005  0000          ;+              SCREENS ARE SAVED AT $7000          +
00006  0000          ;+    FOR 40 COLUMNS CHANGE CPX #8 TO CPX #4        +
00007  0000          ;+                                                  +
00008  0000          ;+                   PROGRAM BY                     +
00009  0000          ;+                HOWARD ROTENBERG                  +
00010  0000          ;+                TORONTO ONTARIO                   +
00011  0000          ;++++++++++++++++++++++++++++++++++++++++++++++++++++
00012  0000          ;
00013  0000                 *=$027A
00014  027A          ;
00015  027A          LOWPTR = $01              ;LOW PTR FOR CRT
00016  027A          HIPTR  = $02              ;HIGH PTR FOR CRT
00017  027A          SAVLOW = $B1              ;LOW PTR FOR STORING
00018  027A          SAVHI  = $B2              ;HIGH PTR FOR STORING
00019  027A  A9 00          LDA #0             ;INITIALIZE PTR'S
00020  027C  85 01          STA LOWPTR
00021  027E  A9 80          LDA #$80
00022  0280  85 02          STA HIPTR
00023  0282  A9 00          LDA #00
00024  0284  85 B1          STA SAVLOW
00025  0286  A9 70          LDA #$70
00026  0288  85 B2          STA SAVHI
00027  028A  A0 00          LDY #0             ;INITIALIZE INDEX COUNTERS
00028  028C  A2 00          LDX #0
00029  028E  B1 01  SAVE    LDA (LOWPTR),Y     ;GET CHR FROM CRT
00030  0290  48            PHA                ;SAVE CHR
00031  0291  B1 B1          LDA (SAVLOW),Y     ;GET CHR FROM MEMORY
00032  0293  91 01          STA (LOWPTR),Y     ;WRITE IT TO CRT
00033  0295  68            PLA                ;GET SAVED CHR
00034  0296  91 B1          STA (SAVLOW),Y     ;STORE IT IN MEMORY
00035  0298  C8            INY                ;FIRST PAGE DONE ?
00036  0299  D0 F3          BNE SAVE
00037  029B  E6 02                INC HIPTR          ;YES INCREMENT PTR'S
00038  029D  E6 B2                INC SAVHI
00039  029F  E8                   INX
00040  02A0  E0 08                CPX #8             ;EIGHT PAGES DONE
00041  02A2  D0 EA                BNE SAVE           ;YES
00042  02A4  60                  RTS                ;BACK TO BASIC
00043  02A5                       .END
ERRORS = 00000
SYMBOL TABLE
SYMBOL VALUE
 HIPTR    0002    LOWPTR   0001    SAVE     028E    SAVHI    00B2
 SAVLOW   00B1
 END OF ASSEMBLY
```

Universal Roll Paper Holder, MK II



by Louis F. Sander
Pittsburgh, PA

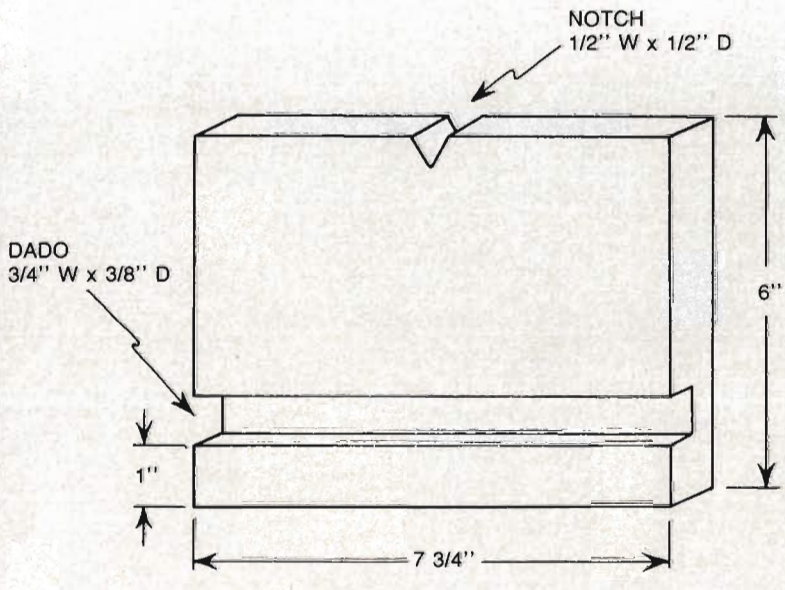
If you built, or are planning to build, the $8 roll paper holder described in the May issue, you might want to consider this improved version, which gives better performance for less money. Of course every improvement has its drawbacks, and the one here is minor—the builder needs to use a power saw. But only two cuts and two dadoes are required, so if you aren't a woodworker yourself, a woodworking friend won't mind spending the twenty minutes it takes to do the sawing.

The holder lets you use inexpensive roll paper with your printer, freeing you to invest in better things than fanfold. As you can see in the photograph, the MK II uses the same Ekco rolling pin as the original model, but only uses one. (There's where the savings comes in!) There's much less friction associated with the MK II, and no tendency for the paper to wrinkle, since the roll's weight is supported from inside.

To make a MK II, get 2 feet of 1"x8" pine or other wood at you local lumberyard. (The actual dimensions will be closer to 3/4"x7 3/4", since lumber measurements are taken before finish planing is performed.) Cut you wood into one piece 11 1/2" long, and two pieces 6" long, preparing the short pieces as shown in the accompanying drawing. The dadoes should be exactly as wide as your lumber, nominally 3/4". The dimensions of the notch are not critical, as long as it is centered on the upper edge. The long piece will work perfectly with a 10 1/2" roller; if yours is a different size, lengthen or shorten this piece accordingly.

Assemble the pieces with woodscrews, or with nails and glue, and you'll have a sturdy paper holder that should last longer than your printer. My son Bill built the unit shown in the photo, and as you can see, he countersunk and plugged the screw holes. He also put rubber feet on the bottom, and stained the whole thing to match my computer desk. The entire project took him two periods in high-school wood shop, earning him an 'A', plus his father's proud appreciation. Your MK II can turn out just as well, and if you built a MKI, you'll even have a leftover roller to return to kitchen service. Bon appetit!□

FIGURE 2—Endpiece Construction Detail



UNIVERSAL ROLL PAPER HOLDER MK II

Reviews for the Vic-20

by Robert L. Foster
Salt Lake City, UT

Apple Panic

Rated ★★★★

A strange name for a strange game! Believe it or not this one has apple monsters! What is an apple monster you ask? It's an unusual little video beast in the shape of an apple. As the game begins the red apples wander about indiscriminately looking for a farmer to pounce on! You are given four farmers to fight these voracious little pouncing creatures. They will devour your farmers unless you have steady nerves, a fast hand and a quick eye. As your farmer moves along the brick roadway he can punch a hole in the brick into which, hopefully, an apple monster will fall. If and when this happens you move the farmer in quickly to pound and mash the monster all the way down into the hole. If you do not hurry the monster can crawl out of the hole and destroy your farmer.

On your first screen there are three red apples to mash into the holes. The second screen has five red apples and the third screen has seven red apples. If your farmer succeeds in mashing all the apples into holes you progress to a much more difficult level where things start to get a bit more complicated.

Now three red apple monsters and a green apple monster appear. The danger now facing your farmer is that the green apple is a pursuer—it will pursue your farmer no matter where he goes, hoping the farmer will drop his vigilance and run into one of the red apples which will pounce on him—and if a red one doesn't it's likely the green one will! In order to destroy the green apple your farmer must smash it through two brick levels, and of course finish off the red apples too! If this should happen you progress to the next level with five red apples and one green apple—then on to the next level with seven red apples and one green apple. However, we've never made it past the second level. Either the green apple gets our farmer or the red ones do. We are still trying though—we never say die! One of these days we'll master all the skill levels.

Apple Panic is an intriguing, exciting video game which will provide many hours of entertainment for the entire family, from any age 6 to 60. Our family was addicted to Apple Panic after playing it only once.

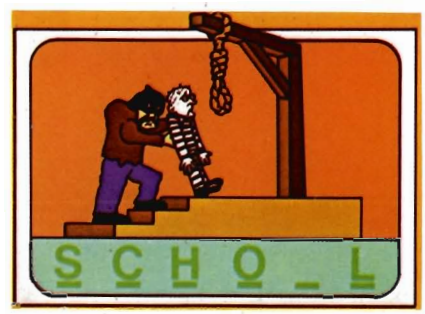
Created by Creative Software, the graphics are excellent as is the sound. Anyone who enjoys a challenge and has the patience to lose fairly often will certainly like Apple Panic, and will keep trying to improve his skill level so he can progress to the next more difficult level.□

Hangman-Hangmath

Rated ★★★

Hangman and Hangmath are video games for entertainment as well as education. Developed by Creative Software, the games are on two-sided cassettes, and were developed for patient people over the age of 12, though small children love the excellent graphics and sound.

Hangman is a video adaptation of the traditional game of Hangman, a game of spelling skill, with many interesting and challenging words to guess—some of them rather difficult! It is fun to win but it is also fun and a bit frustrating when you miss a letter—



put the noose over the doomed man's head, see the trap door release, and watch the very humorous antics of the man at the end of the rope. Smaller children will most certainly hope you don't get the word right so they can see the hangman drop your man through the trap door and laugh at the funny things he does and listen to the interesting sounds which accompany the action!

Hangmath is a game of double-digit multiplication. It is somewhat difficult because the computer gives you only seven guesses for your numbers and you are out. This is a brain twister and quite a challenge. Not for smaller children this game is, however, fun for older children and adults. If you want to lose yourself from the cares of the world in a bit of mathematical fun, and at the same time improve your math skills, this is the game for you.□

Baseball Adversary

Rated ★★★

It's the bottom of the ninth inning, a

tie game; the bases are loaded, with two outs! The man batting eighth is stepping up to the plate. You have three choices, 1. Bunt, 2. Get a pinch-hitter, 3. Let him bat. Since he is batting eighth you decide to put in a pinch-hitter, and he snags a double, scoring two men!

Baseball Adversary is a different look at one of America's favorite games. You take the place of the coach, deciding what to do with the outfield, whether to bring the in-field in, change pitchers, walk the batter, or a combination of any of these.

This new game combines a bit of skill with a certain amount of luck. You are pitted against a worthy adversary, the computer. You have a fairly wide range of choices and a roster keeps track of which batter is "up", so you can decide what strategy you are going to use to win the game.

This game is an armchair coach's delight. You actually imagine yourself in a dug-out at the world series looking grimly at the situation you find your team in. There are few graphics; however, since this is not a graphics and sound game you use a little imagination—and your coaching abilities will eihter win or lose the game.

Developed by Parr Programming, Baseball Adversary is a game of skill, and geared more for an older audience, those who have played or understand the fundamentals of baseball. It will provide many hours of fun and relaxation for you and your friends.□

Treasure of the Bat Cave

Rated ★★★★

Thar's gold in them thar hills, or at least within the caves in the mountain! From your vantage point, as you approach the cave, you can see gold glittering. As you get closer and enter the cave a bat swoops out of nowhere, like the proverbial "bat out of h---!" You put your gunsight on him and fire. A hit! The gold is now yours for the taking.

You find yourself in an endless maze of tunnels with gold treasure scattered about. But the bats, which protect the gold from interlopers, continue to swoop from the dark recesses of the maze trying to stop you. They come from every direction!

This new 3D maze game, developed by Victroy Softwear, is a fast-paced action game, requiring a quick eye for sighting through the cross-hairs, lining up on a bat, and making sure he is shot down before reaching you. The button on your joy stick serves as the trigger.

The game becomes progressively more difficult as you probe deeper into the tunnels in your search for more gold. Bats become much more numerous and fly much, much faster. Your reaction time must become faster and faster. You can only sustain three hits by the bats and that's it—you are finished.

Treasure of the Bat Cave will prove challenging to young and old alike and will provide many hours of fun and relaxation for the entire family.□

Excellent — ★★★★ Good — ★★★

Educational Reviews

by Mary Ann Dodd
Tacoma, WA

Math Softwear Reviews

The answer given most often by families when asked the question, "Why are you buying a personal computer?" is "education for the children." But sometimes the children's idea of education is increasing their dexterity with the joystick and improving their arcade game score. In this month's column I will offer some alternative to the arcade games in the area of math software.□

Math Improvement Six Pack

Commodore
VIC-20 5K Cassette
$59.95

Commodore's Math Improvement Six Pack differs from CAI in that the programs are not designed to teach skills but to provide practice in improving and applying math skills.

The programmers did this in an ingenious way. They disguised the programs as games. In fact, some of the games are so good that if you don't tell the kids they are practicing math skills they just might not know it. The games cover utilization of the four basic operations (addition, subtraction, multiplication and division) and all of the basic facts involved with these operations.

LCM Machine is a slot machine game with three levels of difficulty. On the screen you are presented with a slot machine with a beeping sound. There are numbers rotating. Press "return" to choose the numbers. You have the numbers. What is LCM of these numbers? A choice is made. If correct you win a jackpot and the winnings are added to the Big Dollar Board. Get all ten of the jackpots or $100 on the Big Dollar Board and you get a chance to go to the Jackpot Mine. If the wrong answer is entered—no jackpot and the correct answer is displayed.

Sounds like fun, doesn't it? LCM is an acronym for lowest common multiple. While the kids are winning their jackpots they are practicing finding lowest common multiples which is a necessary skill when performing operations involving fractions such as common denominators and reducing to lowest terms.

Numbowl is loosely based on bowling. The goal of the game is to take three random numbers and combine them by addition, subtraction, multiplication and division and reach as close to thirty as possible without going over. The player is given a choice of two equation patterns. If thirty is exceeded or an illegal operation is performed the computer flashes that the pins are being reset and that the player must try again. A running score is kept in the frame at the bottom of the screen. While playing this game skill is gained in manipulation of numbers within an equation.

Ruler Dueler is a space game with the object being to blast the target on a ruler. This is the only game in the six pack that has optional computer assisted instruction. The computer flashes a fractional number on the screen. The player must choose the corresponding point on the ruler target. A correct response is a hit. If incorrect the laser bounces back and burns the ship. A score is given with the number of right and wrong. The wrong points are left on the screen so that the player can check the mistakes. A rating is given according to the score—study the ruler, good eye or expert.

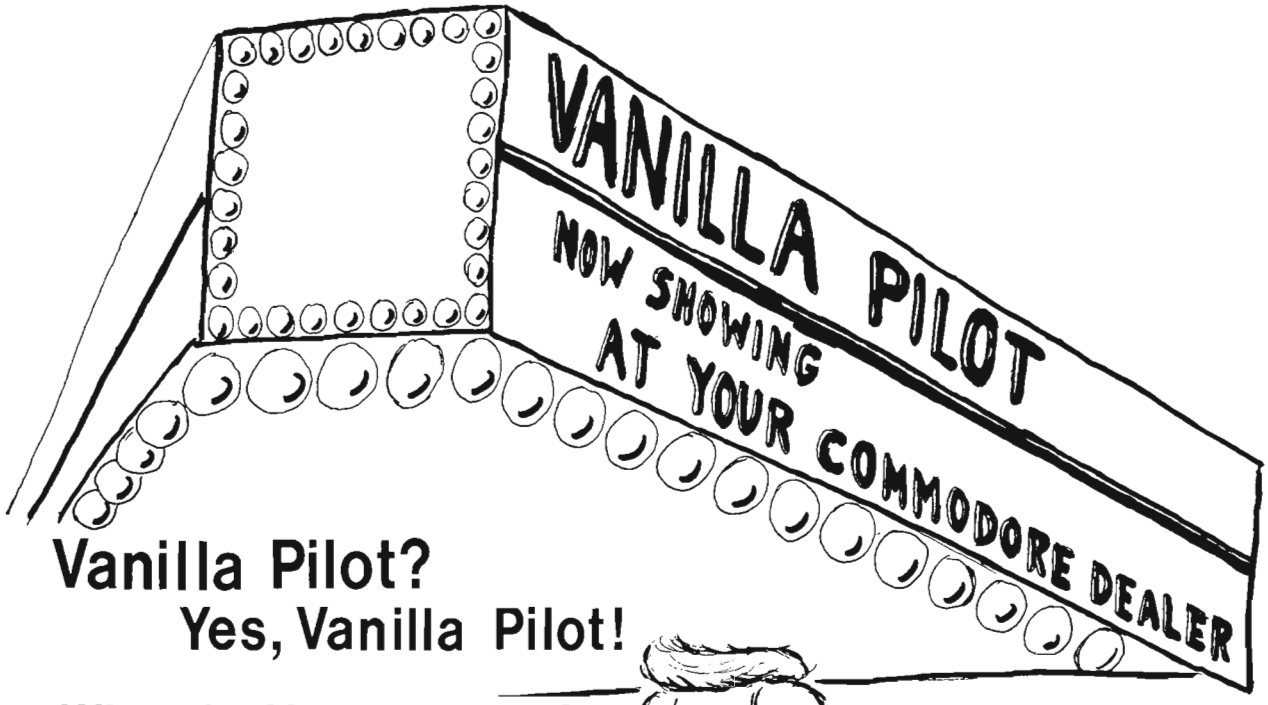
The game is a clever way to learn how to read a ruler. The student can count the points or as he gets more proficient he can glance at the screen and increase his speed.

Backfire starts with displaying an instrument panel on the screen. The object of the game is to identify all the divisors in a given number. There are three levels of difficulty. Each correct answer is a hit. Each incorrect answer is a burn. The score is given as the number of hits, burns and the percentage of accuracy. This game gives the player practice in factoring.

Sector Five is another space game. Upon hearing the warp sounds you are suddenly cast into space as an observer. The colony is being attacked by the Kuminons. The number of invaders must be estimated so that defenses can be prepared. You must be 80% accurate or the colony will be lost. The Kuminons come and the ten second countdown begins. It is impossible to count that fast. The screen is blank. How many were attacking? You enter a number. Alas, you guessed too many and wasted some weapons. The accuracy percentile was not too great. Another attack is coming. This time you guess too low and miss some Kuminons. At the end of ten invasions you are given a percentile rating and the status of the colony.

As the player's skill in saving the colony improves so does his skill in quick visual estimation.

Scare City Motel opens with detective type music. The player is the proprietor of a motel. The assignment is to charge the highest room rate possible and fill all 100 rooms. The computer randomly picks an optimal rate and the player must find it to increase his score. After ten days a score is given as: the best rate, could have



Vanilla Pilot?
Yes, Vanilla Pilot!

What is Vanilla Pilot?

Vanilla Pilot is a full-featured pilot language interpreter including TURTLE GRAPHICS for the PET or CBM 4000, 8000, 9000 and CBM-64 series computers.

At last! A Pilot interpreter for the Commodore computers. This Pilot includes some powerful extensions to the screen editor of the computer. Things like FIND/CHANGE, TRACE and DUMP enhance the programming environment.

The TURTLE has a very powerful set of graphics commands. You can set the Turtle's DIRECTION and turn him LEFT or RIGHT. The pen he carries can be set to any of the 16 colors in the CBM-64. He can DRAW or ERASE a Line.

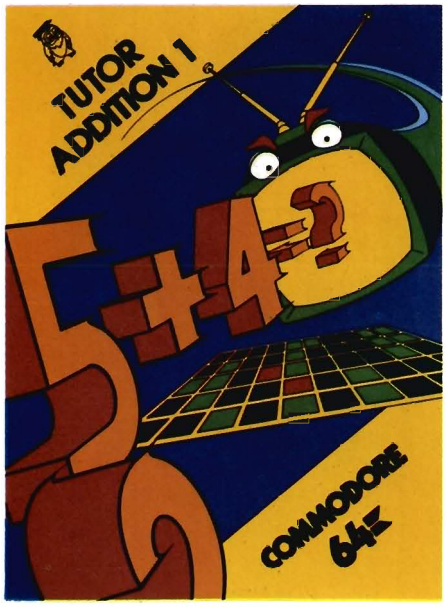
What else? Vanilla Pilot is all this and much, much more. In fact, we can't tell you about all of the features of the language in this small ad. So rush down to your local Commodore computer dealer and ask him to show you Vanilla Pilot in action.

Tamarack Software
Darby, MT. 59829

Circle No. 42

made, what was made, and percentage of how well the player did. This game is a simple simulation that requires players to use all of their math skills and reasoning ability.

With summer vacation approaching, Commodore's Math Improvement Six Pack would be a good choice for summer computer fun. These games would enable youngsters to retain and improve their math skills while really enjoying themselves. Who knows, they might surprise their teachers and themselves by returning to school in the fall with better math skills than they had in the spring.□



Addition Tutor I

COMM*DATA Computer House
VIC-5K Cassette
$16.95

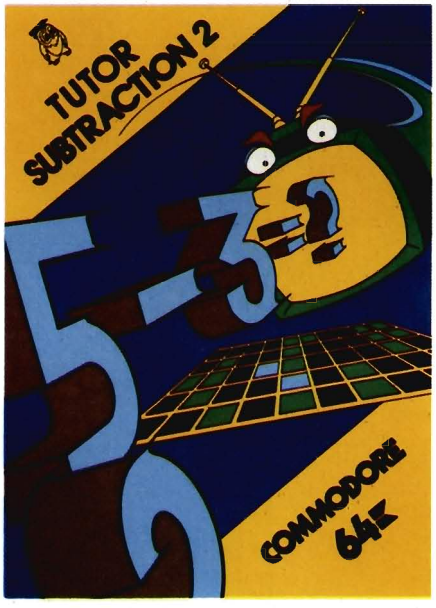
One of the applications of the computer in education is CAI (computer assisted instruction). COMM*DATA in their math tutor series offers an excellent example of this technique

Addition Tutor I is written for the young child who is just beginning to learn the concept of addition. A large red number then an equal number of red dots march across the screen complete with background sound. The red number is joined by a blue number with blue dots. The child counts the total number of dots and enters an answer. If correct, the screen flashes and "good" appears. If the answer is incorrect the computer prints "no" and gives the correct answer. The computer then flashes each dot individually so that the child can count along with the computer. After ten problems a score is given. There is no time limit involved.

The second level which could be used for drill presents vertical addition without the dots.

Addition Tutor I is designed around sound education principles. The large numerals make visual discrimination for a young child easy. The dots allow the child to see the mathematical operation being performed. The simple format is free of distractions that would confuse learning the basic concept.

I would recommend this program for young children first learning basic addition or for older children who are having trouble with basic addition facts.□



Subtraction Tutor II

COMM*DATA Computer House
VIC-5K Cassette
$16.95

Another program in the COMM*DATA series is Subtraction Tutor II. This program teaches two digit subtraction with regrouping.

Like Addition Tutor I there is a visual display of the numerals. The screen is divided into halves with the problem on one side and blocks very similar to Cunisart rods on the other side. When the subtrahend appears it is visually subtracted from the rods and then the student can count the number of tens and ones remaining.

If the correct difference is entered, "you did it" flashes on the screen. If an incorrect answer is entered "no" appears until the correct difference is entered. If the error was made in regrouping the computer patiently prints "remember you borrowed one ten." This one feature is worth the price of the tape. If you have ever tried teaching subtraction with regrouping and became frustrated when the students could not understand that they borrowed one ten, then you will really appreciate this program. The computer will demonstrate to the child visually and then remind the student that he borrowed one ten time after time without ever once raising its voice or losing its patience.

Subtraction Tutor II also randomly mixes simple subtraction and regrouping so that the student has to make a decision whether regrouping is needed. There is a second level without visual aids to be used for drill after the principles of the first level have been mastered. A summary is given after ten problems reporting the number right and wrong.

This program would be very useful for the introduction of regrouping or remedial work with students who haven't mastered the concept of regrouping in subtraction.□

Speed/Bingo Math

Commodore
Commodore 64
Cartridge
$29.95

Speed/Bingo Math is a cartridge containing two games for drilling basic arithmetic facts.

The Speed Math game presents an equation and the player must enter the correct response within 10 seconds. If an incorrect response or no response is entered the correct response flashes on the screen. At the completion of thirty problems a score is given. The game offers five categories: addition, subtraction, multiplication, division and

a mixture of all the operations.

Bingo Math is a one or two player game. Two bingo cards are displayed with a math equation at the bottom of the screen. The player has a choice of using the keyboard or joystick to move the flashing cursor to the correct number within five seconds. Five correct responses in a row wins the game.

Both of these games give drill practice for the basic facts. Because of the time limit involved and the manipulative dexterity needed for the bingo game I would not recommend this cartridge for below fourth grade. Fourth graders and beyond would enjoy the challenge of the games and the fast action involved. The Bingo Math is unusual in that it allows for two people to play and introduces competition as an added motivation.□

Kids and the VIC

Edward H. Carlson
DATAMOST
$19.95

There's an old cliché about not judging a book by its cover. In this case don't judge the book by its title. I am excited about Kids and the VIC. I wish that this book had been around three years ago when we first got a computer at our house. At that time I had no knowledge of the computer and after reading the first chapter of the user's manual I decided that I really didn't understand or care if the bits were byteing the RAMs or was it the ROMs? Therefore, I ignored the electronic marvel completely until my four year old son taught me how to turn it on and load a disk. At that point I reasoned either I would have to start making friends with the computer or all of my parental input would be a syntax error.

Someone at Datamount must have heard my frantic cries. Consider the following questions. Do you:

A. Find manuals with technical information confusing?

B. Need to have information repeated with different applications in several ways?

C. Know very little about VIC BASIC or have a spouse who is scared of the computer?

D. Have or know a child fourth grade or above who might be interested in computers?

If you circled one or more of the above, then check out Kids and the VIC.

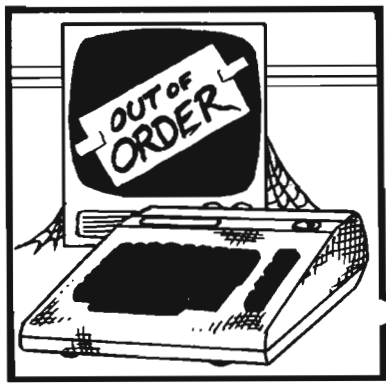
The author, Edward H. Carlson, rates Kids and the VIC for ages 10-14, but I would rate the book for ten years through adult.

The book is designed in a straightforward way. Each of the thirty-three lessons starts with notes to the instructor which summarize the lesson. Next, the lesson is followed by an assignment with possible answers to the assignment in the back of the book. Each lesson usually presents one or two concepts. The concept is stated then followed by a program utilizing the concept. There are also very clever cartoons that clarify the concepts.

The programs in the lessons are short and enable a person who doesn't have great typing skills to succeed in learning to program. There is enough variety to keep a preteen or teenager enthused with subjects which are interesting to them. For example, instead of drawing a square for the graphic introduction, the student draws a "cool, classy car." Both color and sound are introduced early in the book and each lesson reviews and builds upon previous lessons.

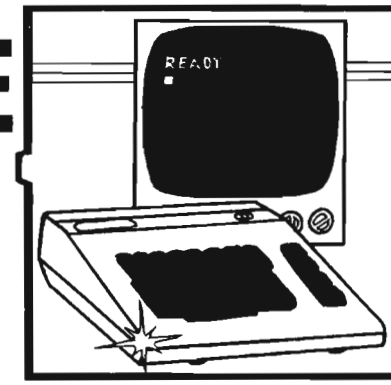
In reviewing Kids and the VIC, I found it relatively free of errors and very easy to understand. The only errors were due to the printer not accurately printing VIC graphics.

After completing this book a novice will have the background to pursue more technical books or write simple programs in basic and perhaps he will be motivated to go on to more difficult projects. Kids and the VIC would be a very pleasant introduction to computers for preteens or beyond and would really get them off to the right start in the fascinating world of computers.□



MAINTENANCE —What are Your Options?

by Tony Lamartina
Pittsburgh, PA



Whether you've already been down the road to obtaining repair service or not, sooner or later, your computer or peripheral (disk drive, cassette drive or printer/plotter) is going to need repair. In this three part series, we'll look at warranty service, non-warranty service and maintenance agreements. We'll also look at several preventive maintenance steps, you as the user, can perform to save money and increase the amount of time your equipment spends at home or the office and out of the repair shops.

The industry standard warranty is 90 days from the date of receipt for parts and labor. There may be variations from manufacturer to manufacturer, but the base warranty is fairly standard, 90 days.

Commodore follows this 90 day standard. For instance, on the 4000 PET, 8000 CBM and 9000 series computers and peripherals, the industry standard 90 day parts and labor warranty is in effect. This is also true of the VIC-20, MDL 64 and new P and B series computers.

If your computer or peripheral requires service within 90 days from date of purchase, you should return the defective unit to the dealer from whom it was purchased for repair. The dealer will repair or replace at his option, the defective components or subassemblies required to return the unit to operating condition.

If you are a VIC 20 or Model 64 owner, you may return your unit for repair to any of the merchandising outlets that sell the 20 or 64. Be prepared for a long wait, however. Unless any of these stores have an agreement with a local repair service, they will have to mail your unit back to Commodore for repair. You could be in for a wait of several weeks, unless the store you return it to is the store where it was purchased. In this case, they may replace your defective unit with a brand new unit.

While these mass market chains have low prices, they have very little post sale support to offer. They are able to offer low prices because they do not have authorized repair personnel on staff. They merely return your unit to Commodore for repair for you. This is not to say that none of these chain outlets offer service. Some may have contracted this service to a local authorized Commodore center or have some other faster service arrangement. You may want to ask before leaving your computer with them for repair.

Another option for warranty service is to check the yellow pages for any authorized Commodore dealer in your area. Give them a call, ask if they do their own service and whether they would be willing to do warranty service even though the unit was bought somewhere else. Most dealers, and this is why dealer prices may be higher than the discount chains, offer full post sale support. They maintain a staff of dedicated sales and service personnel. A local dealer may be interested in doing another outlet's warranty service because Commodore, in the interest of you, the end user, has set up a dealer plan that offers the dealer a labor rebate for any warranty work done and sold by other than that dealer. Be prepared, however, to produce a packing slip, sales receipt or some other proof of warranty.

You could also return the unit directly to Commodore for service in Santa Clara, California or King of Prussia, Pennsylvania or any of the service facilities throughout the U.S.

One note of caution here. If you send your equipment in for repair via UPS or the postal system, overpack them. A safe method of shipment would, of course, be in their original cartons with the original packing materials. Yes, these boxes take up space at home or the office, but should you ever need to send your computer or drive in for repair, you'll be glad you saved them. If you happen to send a 9000 series hard disc drive in for service, for instance, you would be required to send it in its original carton or run the risk of voiding the warranty. At the very least, Commodore will not honor any warranty repairs due to damage in transit. So, save those boxes!

If you hand carry your unit into the local repair facility, no problem, though it's still a good idea whenever you move your equipment, to transport

them in the original packing cartons. This measure almost assures you that your equipment will arrive at its destination in good condition. Even if you're just taking your computer or drive down the street or across town, a little safety measure now, could save you a big repair bill later.

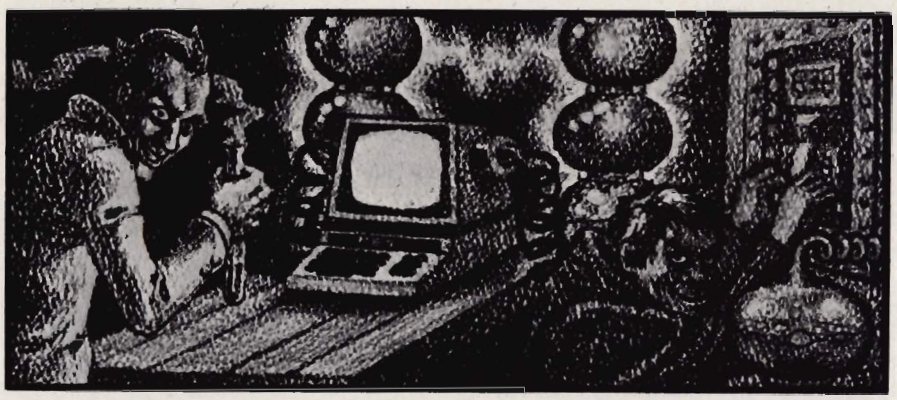
Well, that about covers warranty service. If you own a 4000, 8000, or 9000 series computer or peripheral, your dealers or any authorized servicing dealer can do the warranty repairs on your equipment. If you own a VIC 20 or Mdl. 64, again, any authorized servicing dealer can make the repairs, or if bought from a retail chain, they can obtain service for you, or give you a no charge replacement. For the safe transit of your equipment, always return them in their original cartons, and always, before leaving your computer or pheripheral for repair, ask questions. Is the service done here? Do they send your unit out for repair? Is your repair shop an authorized Commodore repair facility? How long will it be before I get my computer back? Remember, they work for you!

One last thought; whenever you return your equipment for repair, take all the cables, power cords, modulators, etc. in also. The problem may be power supply related, or cable related and not the actual computer or peripheral. The more information you can supply the servicing agent, the quicker the repair will be, and the quicker you'll get your unit back. If you return just the computer, for instance, and the computer operates normally under test, most often, the computer will be allowed to "burn-in" on the bench (operate continuously) over a long period of time to see if the problem might be heat related. This practice is a waste of time if the problem is caused by the cable that goes between the computer and the drive!

Next month we'll look at non-warranty repairs and the options you have available to you.□

Revive That Old "Commodore PET" Computer

by Robert E. Mergy
Santa Teresa, NM



There are a lot of old PETs out there, some are in use every day, some are in closets, and some are stuck back in a corner. Some of these PETs work and some don't. Some of their owners say, "I'll get back to that one of these days." With the economy the way it is today the computer can be your best friend and it can save you lots of money. With the computer programs that are available today you can use your computer to do your bookkeeping, balance your checkbook, speculate on the market or whatever turns you on. The point is—that old computer is not finished yet. It just needs to be revitalized and we will try to show you how to achieve that end. We can make the old PET run like new and at a cost that is less than 10% of a new one. The major cost is the ROMs from Commodore at $75.00 for the set.

The revitalization of your old PET requires a little bit of work and a little bit of knowledge. We will do our best in this article to give you the knowledge required to make this modification to your computer. You can purchase a **PC board** and instructions on how to install it from Better Solutions or you can wire wrap your own board. However, wire wrapping is a bit of a job and will take you a few hours. There are 148 pins to be connected and 120 of these pins will require 2 connections each.

I have owned a 2001 series PET for about three years. I have the large key board 32K model plus a dual disk drive and printer. In the three years that I have owned my PET I have never had one problem with it. Some of my friends have had a problem or two, mostly because they were fooling around inside the machine without any knowledge of what they were doing. There are some very simple rules to follow that will, for the most part, keep you out of trouble:

1. Always unplug your machine before you open it up.
2. Be certain that you understand the instructions before you start.
3. Check and double check your work.
4. Have a friend check your work. It is hard to find your own mistakes.
5. Make sure that you have the proper tools to do the job.
6. Always use an ohm meter to verify your connections and no shorts exist.
7. A good solder joint will have a shine when you finish.

A number of months ago I purchased an old PET at a very reasonable price because it did not work. I took it home and checked it out only to find that one of the ROMs was bad. I ran down to my friendly Commodore's dealer to purchase the ROM, but there are no ROMs available for the old PETs. To make matters even worse they told me that I would have to wait six months. It seems that Commodore only makes them to order and only a couple of times a year. Well, now what? Do I order a ROM from Commodore or do I toss the computer out? No, not that. Well that required a bit of thought; you just cannot rush into a decision like that. After a few days of thought and a good look at a set of schematics that I have for the old PET, I decided to convert the machine from a 28 pin style ROM to the 24 pin style ROM that is used in today's machine. It just so happens that I had on the shelf a set of ROMs that had come out of my PET when I upgraded it to the new BASIC 4.0. This called for a wire wrap job so first I drew myself a schematic of the modification. It took a few hours to do the job (148 pins) and when I was finished and had checked out the board I installed it in the old PET, turned it on and waited, "it works! it works!". My son and daughter were just as happy, but for different reasons. I had told them that they could have it when I fixed it.

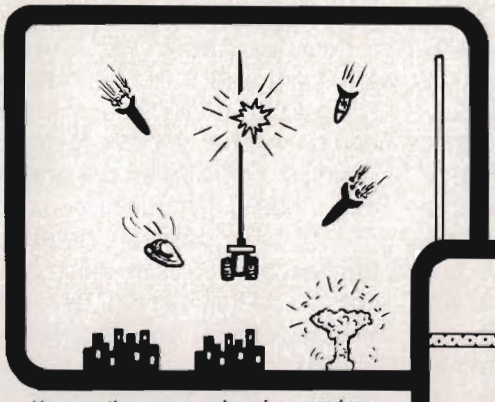
This story may sound as though it is a trival job to convert an exisiting machine to emulate another machine and it is if you know what to do and how to do it. I will try to show you and explain to you the what-to-do and how-to-do of upgrading your old PET to the new ROM's 4.0 BASIC. This will put your PET in a condition that will make it possible for you to continue to upgrade as new versions of PET BASIC becomes available. **"Old PETs Never Die.**

Get acquainted: If you know your way around the inside of your machine then jump over this part and go on to **getting started.** Let us start by getting acquainted with the insides of your PET and at the same time understand some of the basics of the hardware. Oh no! It's back to school again. Yes, I know, I don't much care for it either, but if something should go wrong with your computer how are you going to

D.E.S.-SOFT ™

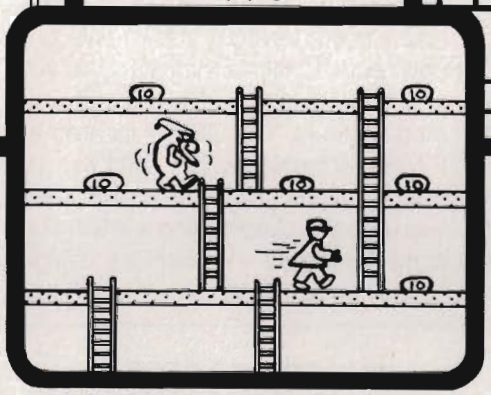
a division of
DES-Data Equipment Supply Corp.

LASER COMMAND by Bob Burnett



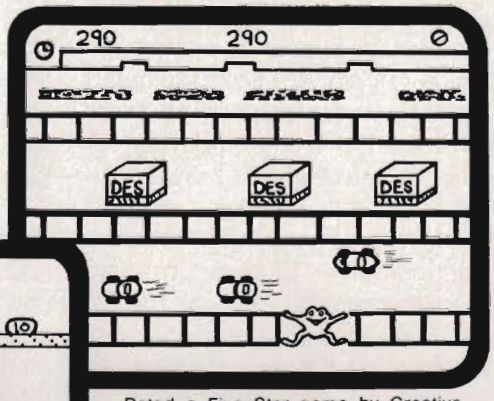
You are the commander of a squadron of laser ships. It is your duty to defend the cities of Earth against the alien onslaught. Spectacular graphics and machine code for super fast arcade fun. VIC-20 and cassette, joystick.
$20.00

BONZO (c)by Kavan



One of the most popular games in Europe. You control BONZO as he climbs the ladders and picks up point blocks. Watch out for the alien guards. Excellent graphics & sound. 100% machine code. VIC-20 w/8K expander, cassette. Joystick or keyboard.
$20.00

HOPPER by Thomas Kim

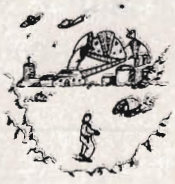


Rated a Five Star game by Creative Computing. Avoid the cars, buildings, logs and other obstacles to bring the frog home. Machine language. VIC-20, cassette and joystick.
$20.00

SHIFTY (c)by Kavan

Watch the maze change as you pass thru the revolving doors. This is a really cute one. Machine language. VIC-20 w/8K expander, cassette. Joystick and keyboard.
$20.00

SPACE TRADER
by Doug Caruthers



Search for the lost planet of Alantia. 1 to 4 players. Text adventure. VIC-20 w/16K expander and cassette.
$39.95

Many exciting titles to choose from.
New Software for the CBM B500/700 series available now !!

PAL 20 ™ — $10.00

- 95 pages of aids, worksheets & logs
- Something for every VIC-20 programmer
- Coming soon for the Commodore 64

NEW for the VIC-20 & 64
electronic ab VIC-20 products

VIC FORTH — **$59.95**
This is a highly capable language that operates from cartridge. It is based on Fig-Forth. Disk and cassette compatible. Vic Forth will work with any memory expansion. 3K of RAM is included in this cartridge.

VIC GRAPH — **$49.95**
The intention of this program is to serve as a mathematical and pedagogic aid for studying complicated equations and functions by their graphs. Plots graphics in high resolution within an x-axis range defined by you. You can also "blow up" parts of a graph in detail by a specified range.

VIC STAT — **$49.95**
Vic Stat is a cartridge which will simplify your work with statistics and graphic displays. It will add approximately 15 commands to BASIC. For example, bar chart, horizontal or vertical, plotting with 2024 points, printout of screen. Statistical commands for calculations of, for example, mean value, standard deviation, variance, etc.

VIC REL — **$59.95**
The purpose of this cartridge is to simplify control of, for example, burglar alarms, garage doors, door locks, heating elements, lamps, radios, remote controllers, valves, pumps, telephones, accumulators, irrigation systems, electrical tools, stop watches, ventilators, humidifiers, etc., etc. This cartridge contains 6 relay outputs and 2 inputs of type optocoupler. For the VIC-20 and Commodore 64.

SEE YOUR LOCAL DEALER TO SEE OUR FINE PRODUCTS

Dealer inquiries invited — ***Software Distribution Available*** — **Programs wanted**

(714) 778-5455 — **D E S** — **Data Equipment Supply Corp.**
8315 Firestone Blvd., Downey, CA 90241 — **(213) 923-9361**

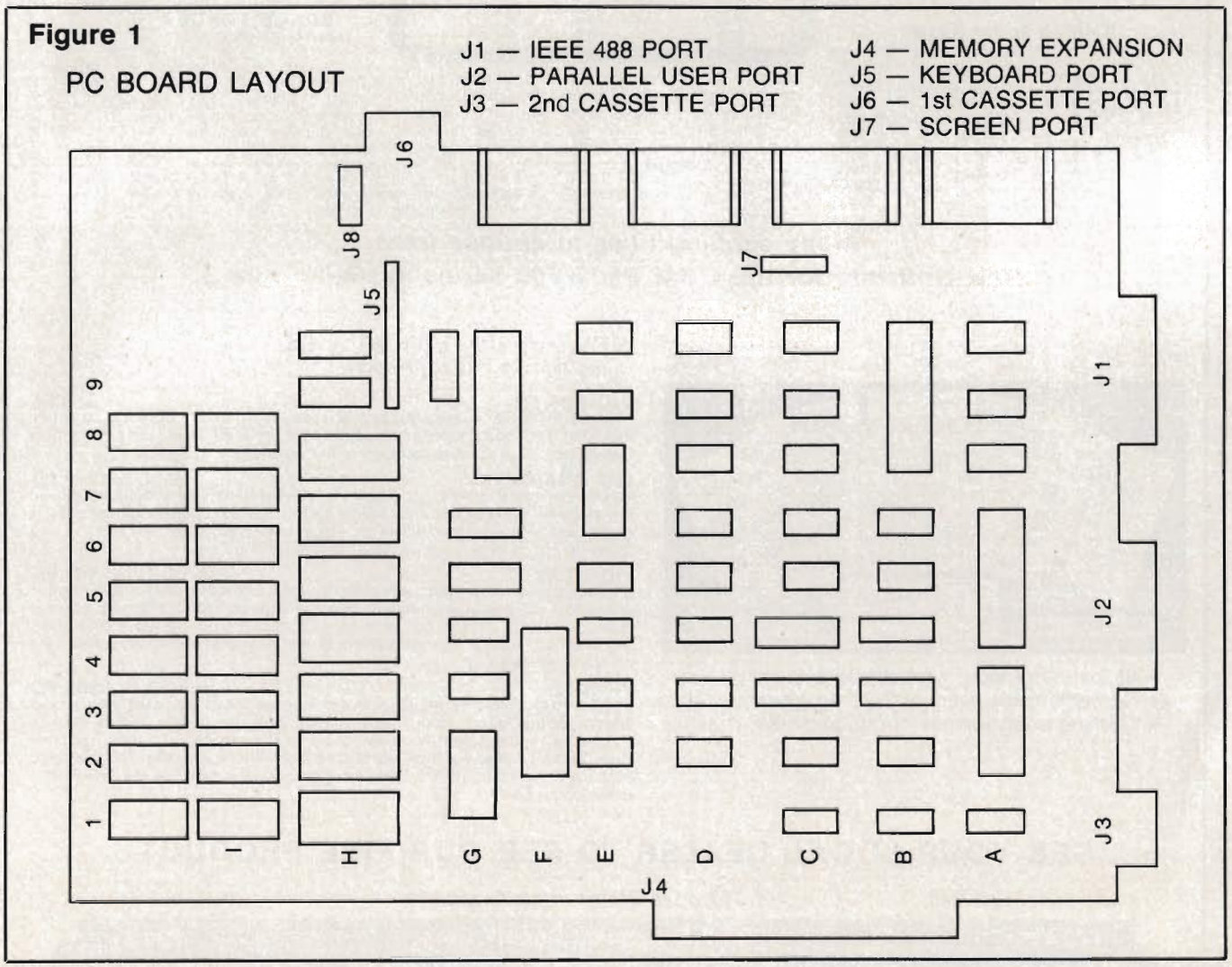
Circle No. 13

fix it? No, don't send it to me. I have all I can handle.

Open up your machine and take a look, isn't that a mess? No not really, but it is a bit complex when you look at it. The facts are that the printed circuit board you are looking at is no more than a piece of epoxy glass with some flat copper wires glued on it and a bunch of little black boxes soldered to them. Heck, that ain't nothing. There is a mess under the hood of your car that is much worse and you work on it. Now take a look at Fig. 1, this is a drawing that represents the PC board in your computer. Study the drawing and try to associate the two. The PC board is numbered across the front edge right to left as is the drawing. It is also lettered along the left side back to front as is the drawing. This is a matrix that we will use to locate the different IC's. You will also notice that there are wires plugged into your machine at different locations, these too are identified with a letter and a number (ie., J1, J2, J3, etc.). Take your time and try to form a good mental picture as to how the machine is assembled. This will help you to put it back together later. If you do not feel sure you have it, then take a little more time. I'll wait. At this point I must make an assumption and that is that you do understand the basics of Bits & Bytes, basic electronic theory and some basic electronic skills. Now get Fig. 2, this is a schematic of the decoding that is used in your machine. Try to locate and identify the IC's shown. We are now ready to take a look at Fig. 3, this is the same schematic but with my modifications. The CPU is located at F3, this is the 6502. The CPU has 16 address lines (A0-A15) and 8 data lines (D0-D7). Addressing is a function of the two different states of a line (high or low, on or off) which will give us 2 to the 16th power (65535) locations. The decoding scheme used is the 4K bytes per block. This scheme uses the top 4 address lines to select the block that you are going to address. The lower 12 address lines select the block that you are going to address. The lower 12 address lines select the individual byte that we want. Think of it as a room with 4 file cabinets each having 4 drawers all lined up in a row. Each drawer has 4000 file folders and in to each one we can put one byte. To decode our computers memory we put a 16 bit address on the address bus; the top four tells us which drawer to go to and the lower 12 tells us into which file folder to look. The read/write line tells us to get (read) or to put (write) something from the file folder. The data bus is our hand, it is what gets the contents of a file folder or puts into the file folder its contents. The IC that does this decoding is a 74154 located

Figure 1

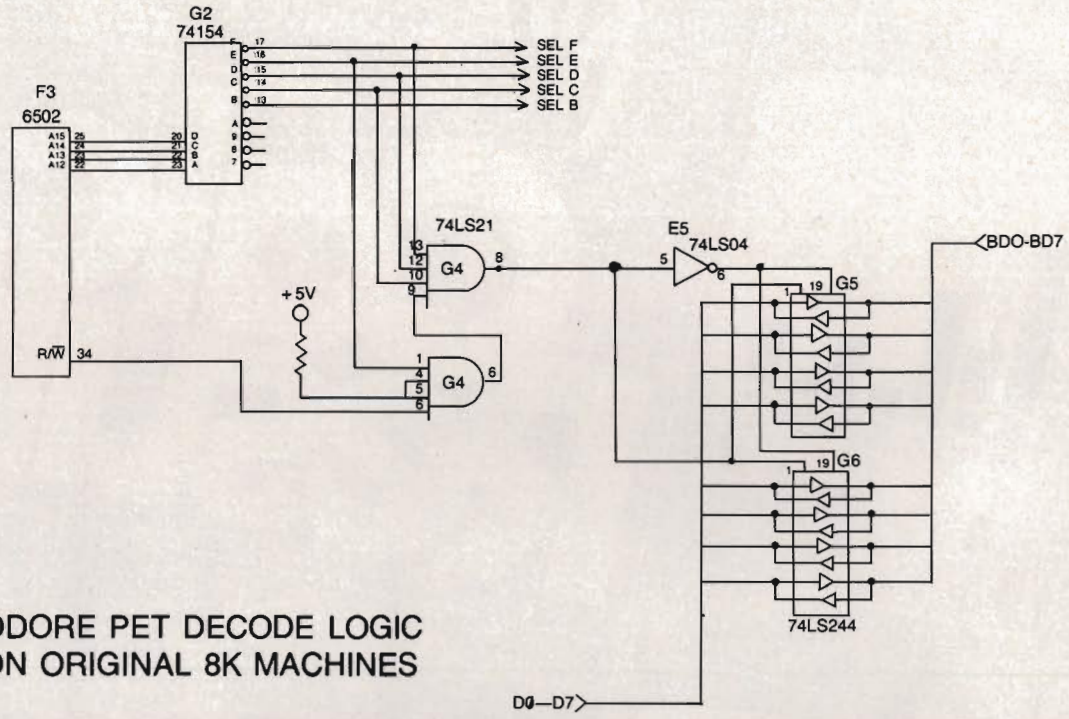
PC BOARD LAYOUT



at G2. Located at G4 is a 74LS21 which is a 4 input **AND** gate that is used to select internal/external memory. The only internal memory blocks used by PET are Ø, 1, 8, C, D, E and F. The balance is expected to be external memory to the computer selected via the memory expansion port J4. The top four address lines go to the 74154 decoder and the output of this chip is decoded by the 74LS21 for internal/external memory via the buffer ICs at G5 and G6 (74LS244).

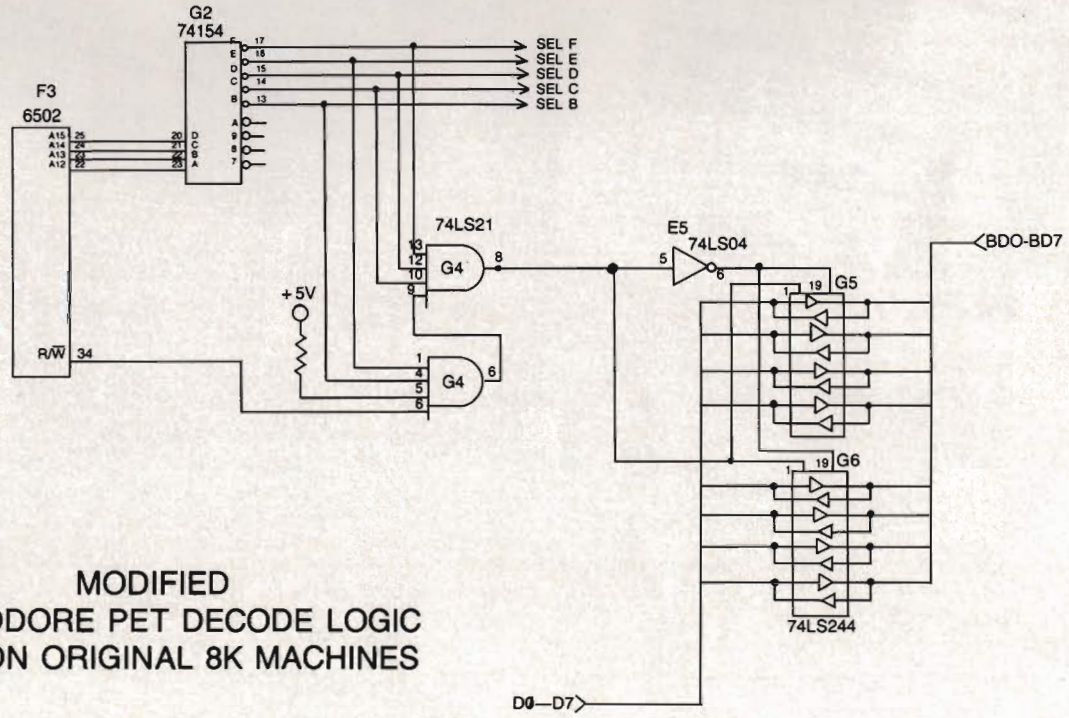
The **basic interpreter** resides in ROM (read only memory) and is located at H1 thru H7. These ROM chips are of a variety that is no longer available. They are 2K bytes by 8 bits (16K bits each) and are in a 28 pin package. Basic 4.Ø is available only in 4K bytes by 8 bits (32K bits each) and

Figure 2

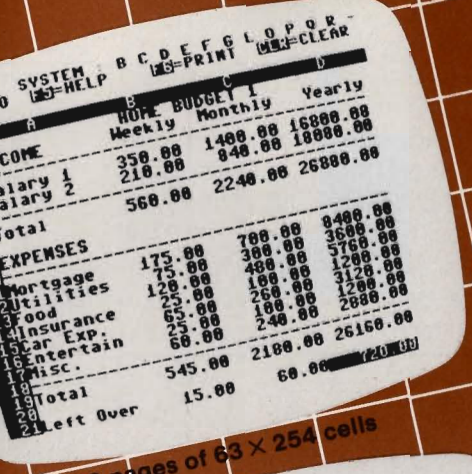


COMMODORE PET DECODE LOGIC
USED ON ORIGINAL 8K MACHINES

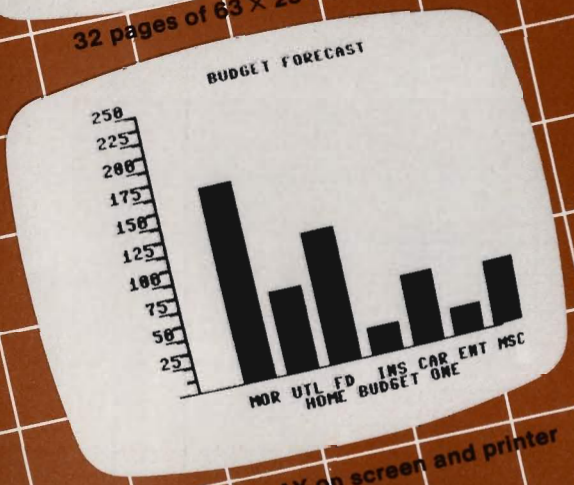
Figure 3



MODIFIED
COMMODORE PET DECODE LOGIC
USED ON ORIGINAL 8K MACHINES

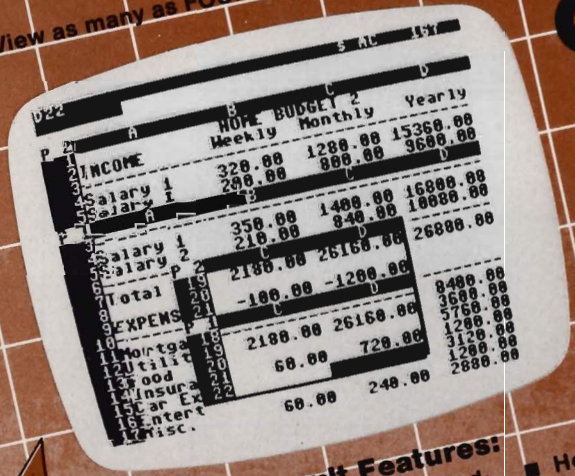


32 pages of 63 × 254 cells



GRAPHIC DISPLAY on screen and printer

View as many as FOUR pages at one time



Calc Result™

For the COMMODORE 64™ with disk drive

We started with the best that standard spreadsheet programs could offer.

Then we added features designed to create a planning tool more useful than ever before.

The Result?

CALC RESULT! The most powerful, understandable and economical spreadsheet program on the market today.

Thirty-two pages, graphics and the ability to view up to four pages (spreadsheets) at once make CALC RESULT the spreadsheet program that outfoxes them all!

Calc Result Features:

- A three dimensional spreadsheet with 32 pages of 63 × 254 cells, offering unrivaled flexibility
- Graphic display on screen and printer
- The ability to view as many as four pages at once through a window and split screen. This allows you to compare spreadsheets!
- Help function on-line to make Calc Result's features easy to use
- Color coordinated cells that speed calculations
- IF-THEN-ELSE with AND, OR and NOT-ELSE functions in each cell give you unlimited possibilities for decision making
- Timesaving full function editing
- Flexible column width printouts for formatting reports
- Utilization of memory only in cells that are active
- Replicate, copy and move commands that save time
- Consolidation of spreadsheets to get the "bottom line"
- Protection of cells containing formulas
- Ability to load VisiCalc™ files

Distributed by:

COMPUTER MARKETING SERVICES INC.

300 W. Marlton Pike • Cherry Hill, N.J. 08002 • 609-795-9480

"Our products outfox them all!"

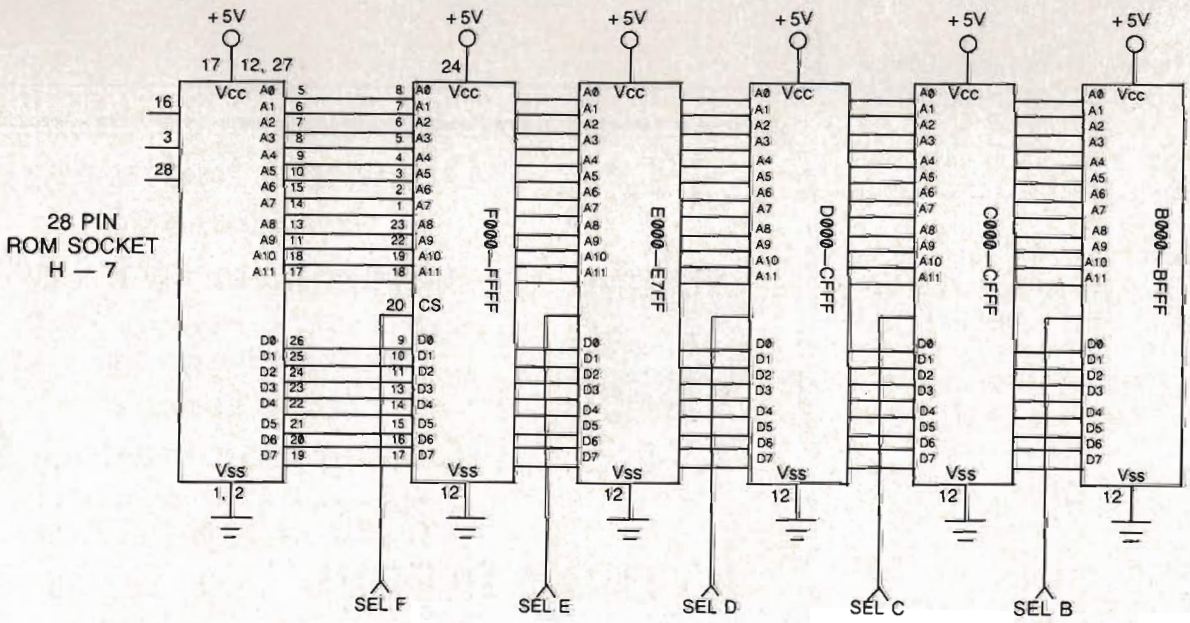
are in a 24 pin package. This 4K ROM chip will work just fine in our machine, but we will have to make a translator board to achieve the proper pin-to-pin connections. This board will need five 24 pin sockets (one for each of the 4.0 ROM set) and one 28 pin connector to be used to pick up each of the address lines (A0-A11), the data lines (D0-D7) plus power and ground. The block select lines will need to be picked up at the output of the 74154 decoder. The block select lines used to decode 4.0BASIC are B, C, D, E and F (pins 13, 14, 15, 16 and 17).

Getting started: Take a felt tip laundry marker and put a mark on one side of each connector and then put a corresponding mark on the PC board beside the mark that you just put on the connector. This will help you to reconnect the wires later. Now disconnect the wires, remove the three screws that retains the PC board, and with a pair of needle nose pliers depress the catch on the plastic retainers and at the same time lift up gently on the PC board. Remove the PC board from the computer and lay it on a towel on a flat surface. That was easy. Now look at the PC board and position it with the numbers to the front and the letters to the left. Now refer to Fig. 2, locate the following IC's F3, G2, G4, G5 and E5. These are the IC's that decide if the memory location is on the PC board or off the PC board (via memory expansion port J4). Now turn to Fig. 3, notice that pin #4 of G4 has been separated from pin #5 and connected to pin #13 of G2. Turn the PC board over and lay it flat on the table with the heat sinks hanging over the side of the table. Now relocate the G4 IC pin #4 & #5, notice that there is a connection between them. Take an exacto knife and cut this connection away. Take a small solid wire and solder one end to pin #4 of G4 and the other end solder to pin #13 of G2. Now then from pin #13 of G2 follow the copper trace to the memory expansion port pin #25 of J4. Make a small cut in this copper trace close to the port J4 separating the trace about 1/16". This is to protect your machine from bus contention by having more than one device addressing the same location. This could save you some repairs.

Construction: We are now ready to construct our ROM board. Turn to Fig. 5, this is a layout of the ROM board. This will take some time as there are 148 pins to be connected and most require two connections. Turn to Fig. 4, this is a schematic drawing of the circuit. Be sure that you follow this drawing to the letter. Now that you have finished the WW board proceed to install it in your computer by removing the ROM IC's at locations H1 thru H7. Note the location of pin #1 of these IC's. Now install your new ROM set 4.0 BASIC into your WW board. Be sure that you follow the layout in Fig. 5.

The next task is a bit tricky but you can do it. You have come this far and you are not going to quit now. Take an eraser and scrub pins 13 thru 17 of G2 including the solder and the PC board, make them clean and bright. An ink eraser works best. Take the 5 pin male molex header and glue it to the top of G2 with the pins hanging over the side of the IC and touching G2's pins 13, 14, 15, 16 and 17. Now using a 25 watt soldering iron solder the molex pins to the IC pins. Just be careful not to over heat the IC. Turn to Fig. 6, this is a circuit that you should build and put on your bench as you will need it to check for wiring errors. When the test probes are put together it will buzz.

Figure 4



1 28 PIN 3 LEVEL WIRE WRAP SOCKET
5 24 PIN 2 LEVEL WIRE WRAP SOCKET
1 22-27-2051 MOLEX HEADER

1 22-01-2057 MOLEX PLUG
5 08-50-0114 CRIMP TERMINAL
WIRE WRAPPING TOOL

WIRE (FOR WIRE WRAP)
1/2"x2"x2" STYROFOAM
5 CONDUCTOR RIBBON CABLE

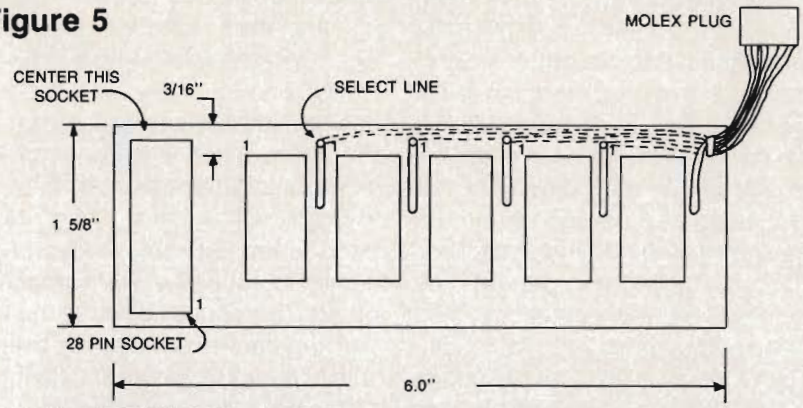
You can also use an ohm meter. You are now ready to plug in your handy work and go for it. Install your board and turn on your machine. You should have the 4.0 BASIC sign on message.

A printed circuit version of the above is offered as a kit from Better Solutions, PO Box 581, Santa Teresa, NM 88008, for $49.95 plus $1.50 shipping. This kit comes with very good installation instructions. The ROMs containing 4.0 BASIC are available from Commodore Business Machines Inc. or one of their dealers for about $75.00 for the set (5 ROM's).

Check List:

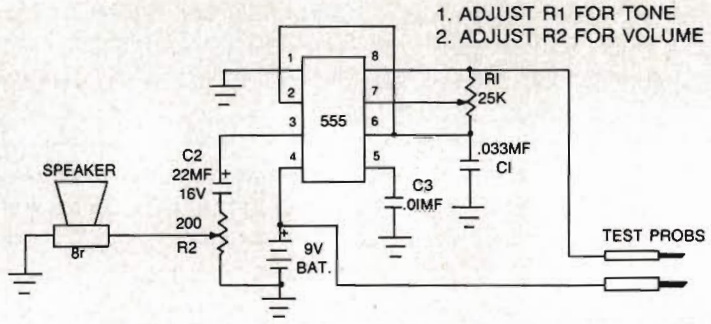
1. Separate pins 4 & 5 of IC G4.
2. Solder jumper wire from pin 4 of IC G4 to pin 13 of IC G2.
3. Cut trace that goes to memory expansion port J4.
4. Molex plug soldered to IC G2 pins 13, 14, 15, 16 and 17.
5. Double check that you do not have any solder bridges.
6. Check that you do have continuity from IC G2 pin 13 to pin 20 of ROM "B".
7. Check that you do have continuity from IC G2 pin 14 to pin 20 of ROM C.
8. Check that you do have continuity from IC G2 pin 15 to pin 20 of ROM D.
9. Check that you do have continuity from IC G2 pin 16 to pin 20 of ROM E.
10. Check that you do have continuity from IC G2 pin 17 to pin 20 of ROM F.
11. Inspect your work again to be sure that you do not have any shorts.
12. Do all of your solder joints have a shine on them? If not, please re-do.
13. Do you have all of the cables properly installed in their correct positions?
14. Do you have the ROM chips properly installed? Pin #1 in proper location.
15. With an ohm meter read from pin 24 to pin 12 of any of the ROMs, you should get a reading of about 20 ohms. If you get a very low reading, then look for a short. If you get a very high reading, then look for an open. □

Figure 5



LEAVE RIBBON CABLE TO MOLEX PLUG 2 1/2" FROM BOARD. SLOT THE BOARD AND RUN RIBBON CABLE THRU AND UNDER THE BOARD. BRING EACH SELECT LINE THRU A HOLE ALONG SIDE OF THE SOCKET THEN BACK THRU THE BOARD TO PIN 20. THIS KEEPS THE WIRE DRESSED CLOSE TO THE BOARD. HOLD THE SIZE AND SPACING SHOWN TO PREVENT INTERFEANANCE WITH OTHER COMPONENTS AND TO ALLOW FOR FUTURE MODIFICATIONS.

Figure 6



SJB DISTRIBUTORS. THE MOST COMPETITIVE PRICES ON COMMODORE.

commodore

NEW COMMODORE PRODUCTS

CBM 128-40	$ 695
CBM C128-80	795
CBM B700	2990
CBM 1520 Plotter	169
CBM 1701 Color Monitor	269
B Series Software	Call
CBM 4023 Printer	399

SOFTWARE FOR CBM 64

Word Processing (WordPro 3+)	$ 69
M File Database (merge with Word Pro)	89
Quick Brown Fox	56
Writer's Assistant (easy and flexible)	99
File Assistant (database with merge)	99
Spreadsheet Assistant	99
Pers. Finance Assist.(great reports)	45
Busicalc (Spreadsheet)	55
Coco II (build your own games easily)	45
Calc Result	125
General Ledger, A/R, A/P (with check writing)	ea.139
CBM EasyFinance	50
CBM EasyScript	80
CBM EasyFile	80
Data Manager	70
Stock(investment analysis)	80
Pet Emulator (emulates 4.0 basic)	30
Sprite-Master	29
Assembler Package (cassette or disk, compiled, includes editor, loader, disassembler)	39
Spacebelt	20
Retroball	34
ZWARK	25

INTERFACES & ACCESSORIES

80 Column Expander	$159
VIC 1600 Modem	95
VIC 1650 (auto answer, auto dial)	150
VIC 1525 Graphic Printer	329
VIC 1530 Datasette Recorder	65
VIC 1541 Disk Drive	329
VIC Switch (connect 8 64's or Vics to printer, dd)	149
PET-IEEE cable	33
IEEE-IEEE cable (2m)	43
Parallel Interface (Epson, Okidata, IDS, NEC)	80
RS-232 Printer Interface (Okidata, Diablo, etc.)	60
Programmers Reference Guide	18
Verbatim Diskettes (10 per box)	26
Victree (Programmers Utility)	75

VIC PRODUCTS & ACCESSORIES

8K RAM Memory Expansion Cartridge	$ 40
16K RAM	70
24K RAM	105
VIC 3 Slot Expander	27
VIC 6 Slot Expander	70
RS-232 Printer Interface	65
Cassette Interface	30
Gorf (64 also)	30
Omega Race	30
Arcade Joystick - Heavy duty w/2 firing buttons! Great for the VIC or 64	25

MONITORS - GREAT RESOLUTION (64 OR VIC)

Amdek Color I	$ 299
Amdek II or III	call
Panasonic CT160	279
Comrex 6500 - 13" Color	299
Transtar 20 (High Resolution Green Phosphor)	129
Video/Audio Cable	15

PRINTERS - LETTER QUALITY

CBM 8300, 40 cps	$1450
Diablo 620, 25 cps	949
ComRiter, 17 cps	819
Transtar 130, 16 cps (auto load, wp features!)	769
NEC 7700 series	2350
NEC 3500 series	1600

PRINTERS - DOT MATRIX

CBM 8023, 150 cps/graphics	589
Epson FX Printer, 160 cps	549
Epson MX-80FT	459
Epson MX-80 w/Graftrax	349
CBM Graphics for Epson	65
Okidata 82A, 120 cps (serial and parallel)	429
NEC 8023A (parallel)	429
Okidata 92	559
Star Gemini, 10	329
Star Gemini, 15	499

COMMODORE BUSINESS SERIES

SuperPet (5 languages, 2 processors)	$1409
CBM 8032 Computer, 80 Column	1029
CBM Memory Expansion, 64K	359
CBM 8050, 1 mg. Dual Drive	1259
CBM 8250, 2 mg. Dual Drive	1395
CBM D9060, 5 mg. Hard Disk	1995
CBM D9090, 7.5 mg. Hard Disk	2295
CBM 2031, 170K Single Drive (New)	489
DC Hayes Smart Modem	220

BUSINESS SOFTWARE

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

MasterCard, Visa, Money Order, Bank Check

COD (add $5) accepted.
Add 3% surcharge for credit cards.
In stock items shipped within 48 hours, F.O.B, Dallas, Texas
All products shipped with manufacturer's warranty.

Prices are subject to change without notice.

TO ORDER
CALL TOLL FREE
800-527-4893
800-442-1048
(Within Texas)

Business Hours
Mon.- Fri. 8 to 6, Sat. 10-2

Write for free catalog.

GAME OF THE MONTH

Adventu-Writer (make your own adventure game)	39

PRODUCT OF THE MONTH

INTERPOD (intelligent IEEE RS232, serial interface for VIC or C64)	$ 179

SJB DISTRIBUTORS INC.
10520 Plano Road, Suite 206
Dallas, Texas 75238
(214) 343-1328

Auto-Clock

by Colin F. Thompson
Santa Monica, CA

One way to gauge the maturity of a computer system is to count the number of exotic peripherals made for it. The VIC and C-64 have just taken a giant step toward maturity with the release of their first clock card.

Clock card? As the name implies, it keeps time. But, you say, the VIC/64 has a built in clock. Why buy another? Good question. For those of you unfamiliar with clock cards, a description is in order.

The primary function of any clock card is to keep accurate time and make that time available to the operating system. Most clock cards allow the user access to a time/date/calendar and alarm on comparison. Secondary functions may include modem control or print buffering.

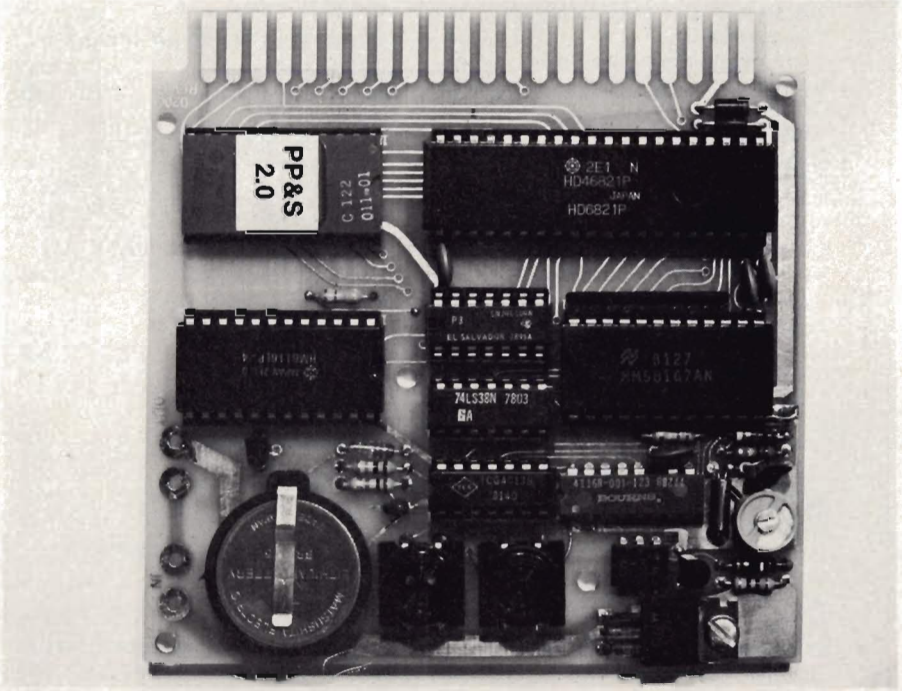
Auto-Clock

The Auto-Clock (AC), made by Progressive Peripherals & Software has taken the concept a step further. The card allows the programmer access to time/date/calendar and alarm functions from a menu, or from a BASIC or machine language program. An unusual (unique?) function is 2K of onboard CMOS RAM powered by a lithium battery. User programs may be loaded from and saved to this 2K block. Finally, the card can switch on and off any electrical device(s) using up to 300 watts of power.

AC is a cartridge which plugs into the VIC's expansion port. Construction of the card looks first rate. The pins are gold plated and the compact layout is uncluttered.

AC occupies Block 5 starting at $A000. When you turn on the power to the VIC, the Auto-Clock menu comes to the screen instead of the usual power-up message. The menu options are Set Time, Set Alarm and Exit. The Set Time option asks you to key in the time, date, month and year. AC will remember this data even after the VIC is turned off because the card is battery powered. The time is updated by a National Semiconductor MM58167A Microprocessor Real Time Clock chip. This clock chip is the heart of the AC.

The alarm is set from a similar menu. The alarm can be set to go off at any time in the future. It compares the real time to the scheduled alarm time and when the two match, the word ring is flashed on the screen and an audible buzzer sounds five times. While the alarm is going off, all current program operations are suspended. After the five buzzes sound, your program continues without knowing it had been interrupted. The alarm can also trigger many other functions we will cover later.



Power Switching

To function as a real-time controller, AC must be able to send signals to a remote device. It does this by switching the power line to the device. The device to be controlled can be any electrical device using 110VAC or low voltage (10V). These devices may draw up to 300 watts. If the load exceeds 300 watts or the load is inductive, (large motors, fans, etc.) a relay should be used.

The user's manual gives detailed instructions for the wiring and soldering needed to do this. It's not difficult. If you know which end of a hot soldering iron to hold onto, you can do it. When completed, you will have a power cord leading to the AC, and an extension cord coming out of AC. The devices to be switched can then be plugged into the extension cord. (See figure 1)

Plug the VIC's power supply into the extension cord and you are ready to test the connections. The first thing you will notice is the VIC's power switch does not work. To manually turn on the VIC, push in the **On Switch** mounted on AC. The VIC will power up and AC's menu will come on the screen.

Now that the wiring has been done, let's see what Auto-Clock will do under program control.

The Sky's the Limit

From your own BASIC program you can call 16 different subroutines built into the AC's ROM. The machine language programmer has another 19 routines available. (See the Table of Contents.) These routines will provide any clock data, alarm data or access to the 2K RAM block. A SYS command will call any of the routines for your use. I've built one subroutine into many of my programs. It displays the time on the screen. I can display the time in any color and in any place on the screen, even while writing a program. The time is updated once a second, and doesn't interfere with my program.

Blue Sky

My favorite feature is the 2K block of RAM. I treat it like a 2K ramdisk. You can save BASIC or M/L programs and load them back to main memory with a SYS call. If a program is stored in the 2K block, it will be loaded and run when power is applied, or the computer is reset. This is done automatically, unless you hold the Return Key down during power-up. My pet peeve with the VIC is the screen color. Blue on white is the pits. My eyes prefer yellow on black. AC will solve this little problem easily. My 2K block usually holds a short M/L wedge called KeyBeep which sounds a short beep when any key is struck. I've added some BASIC code to KeyBeep which pokes the screen black and prints the following on the screen (in yellow):

```
LOAD"TOTL.LABEL",8
LOAD"PRACTICALC",8
SYS24576: REM QBF
SYS64802: REM RESET
28159 BYTES FREE
READY
```

On power-up the KeyBeep loads to the cassette buffer, the screen turns black and three of my most used programs are on the screen, waiting to be called. Any M/L routine that can be put in the cassette buffer could also be stored in and called from the 2K block.

Chain Gang

Auto-Clock will allow you to chain a series of programs called from the disk, at any time. To accomplish this feat, you write a master program and store it in the 2K memory. This master has the names of all the programs that will be called in from disk. Each called program calls the next program. By using the dynamic keyboard technique, each program can be loaded and run, in any order you wish. You can use any other clock feature with this chain. The procedure is simple and completely explained in the manual. The entire BASIC master is listed there.

Program chaining, under clock control, makes AC a very powerful controller. I know of one person in the Midwest using AC to control the irrigation of fields. Your imagination is the only limit to the AC.

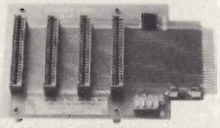
Manual Labor

I was impressed with the accuracy and thorough documentation found in the user's manual. Every subroutine is explained with a few lines of BASIC code, showing exactly how to use it. The M/L routines are accompanied by assembly code to show the proper usage. A complete memory map, table of contents, and a science fiction short story round out the manual.

Steve Luedders and Steve Spring have done a masterful job of designing the Auto-Clock. When I asked them if this is a precursor to the fabled "battery powered RAM card" they just smiled. I can't wait to see THAT!□

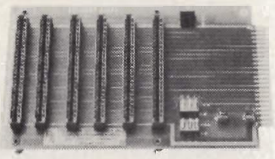
Table of Contents for Auto-Clock

VIC-20 and CBM 64
EXPANDER BOARDS

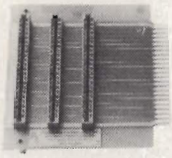


4 Slot for the 64. Toggle switches and reset switch.
P/N C64 **$69.95**

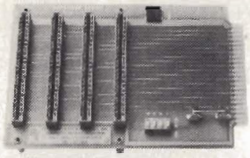
PTI offers the finest selection of expander boards available for the VIC-20 and CBM 64. The design features, quality construction, and competitive prices make any of them an exceptional value. New products are being added monthly, so write for complete catalog.



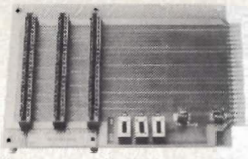
6 Slot for the VIC. Toggle switches and reset switch.
P/N V36 **$79.95**



Slot for the VIC. No switches, reset, or fuse.
P/N V13 **$49.95**



4 Slot for the VIC. Toggle switches and reset switch.
P/N V24 **$69.95**



3 Slot for the Vic. Slide switches, no reset switch.
P/N V23 **$59.95**

PRECISION TECHNOLOGY, INC.
COMPUTER PRODUCTS DIVISION
P.O. BOX 15454
SALT LAKE CITY, UTAH 84115
(801) 487-6266

See your dealer, or place your order direct

VISA-M/C-CHECK-COD

Circle No. 33

C64-FORTH
for the Commodore 64

FORTH SOFTWARE FOR THE COMMODORE 64

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

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

(Commodore 64 is a trademark of Commodore)

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

PERFORMANCE MICRO PRODUCTS

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

Circle No. 32

FOR THE VIC 20™

JOIN THE COMPUTER REVOLUTION WITH A MASTERY OF THE KEYBOARD!

In the age of the computer, *everyone* from the school child to the Chairman of the Board should be at home at the computer keyboard. Soon there will be a computer terminal on every desk and in every home. Learn how to use it right ...and have some fun at the same time!

Rated THE BEST educational program for the VIC 20™ by Creative Computing Magazine

TYPING TUTOR PLUS WORD INVADERS — $21.95
(2 programs on one cassette tape for the unexpanded VIC 20™)

Typing Tutor plus Word Invaders makes learning the keyboard easy and fun! Typing Tutor teaches the keyboard in easy steps. Word Invaders makes typing practice an entertaining game. Highly praised by customers:

"Typing Tutor is great!", "Fantastic", "Excellent", High quality", "A source of great joy and learning for our children", "Even my little sister likes it", "Word Invaders is sensational!"

Customer comment says it all . . .

". . . and it was everything you advertised it would be. In three weeks, my 13 year old son, who had never typed before, was typing 35 w.p.m. I had improved my typing speed 15 w.p.m. and my husband was able to keep up with his college typing class by practicing at home."

FOR THE COMMODORE 64™

SPECIAL VERSION OF TYPING TUTOR PLUS WORD INVADERS NOW AVAILABLE FOR THE COMMODORE 64™...$21.95 *(Tape)*

All of the features of the VIC 20™ Version and more

SPECIAL!

SPRITE DESIGNER
by Dr. Lee T. Hill

$16.95 *(Tape)* $21.95 *(Disk)*

Create and then transform sprites automatically. We have the other sprite making programs, but this is the one *we use* to make sprites. The automatic transformations are great!

Shipping and handling $1.00 per order. California residents add 6% sales tax. VISA and Mastercard orders must include full name as shown on card, card number, and expiration date. Free catalog sent with order and on request.

ACADEMY SOFTWARE

P.O. Box 9403, San Rafael, CA 94912 (415) 499-0850

Programmers: Write to our New Program Manager concerning any exceptional VIC 20™ or Commodore 64™ game or other program you have developed.

Circle No. 2

AARDVARK — THE ADVENTURE PLACE

TRS-80 COLOR COMMODORE 24 VIC-20 SINCLAIR/TIMEX TI99

WE CARRY MORE THAN ADVENTURES!!

MAXI-PROS WORD PROCESSING NEW

The easiest to use word processor that I know of. Has all the features of a major word processor (right and left margin justification, page numbering, global and line editing, single, double, triple spacing, text centering, etc.) at a very cheap price because we wrote it in BASIC. Includes 40 page manual and learning guide. Easily modified to handle almost any printer combination. Available on disk or tape for VIC20, COMMODORE64, and TRS-80 COLOR computer. Requires 13k RAM on Vic, 16k EXTENDED on TRS-80 COLOR. $19.95 on tape $24.95 on disk. NEW

GENERAL LEDGER — Complete bookkeeping for a small business. Disk required. For Vic20 (13k), Commodore64, TRS-80 COLOR (16k EXTENDED). $69.95 (Send $1.00 for manual before ordering.)

LABYRINTH — 16K EXTENDED COLOR BASIC — With amazing 3D graphics, you fight your way through a maze facing real time monsters. The graphics are real enough to cause claustrophobia.

Similar game for Timex/Sinclair 16k - hunting treasure instead of monsters $14.95.

ADVENTURE WRITING/DEATHSHIP by Rodger Olsen — This is a data sheet showing how we do it. It is about 14 pages of detailed instructions how to write your own adventures. It contains the entire text of Deathship. Data sheet - $3.95. NOTE: Owners of TI99, TRS-80, TRS-80 Color, and Vic 20 computers can also get Deathship on tape for an additional $5.00.

Dealers—We have the best deal going for you. Good discounts, exchange programs, and factory support. Send for Dealer Information.

Authors—Aardvark pays the highest commissions in the industry and gives programs the widest possible advertising coverage. Send a Self Addressed Stamped Envelope for our Authors Information Package.

ADVENTURES — Adventures are a unique form of computer game. They let you spend 30 to 70 hours exploring and conquering a world you have never seen before. There is little or no luck in Adventuring. The rewards are for creative thinking, courage, and wise gambling — not fast reflexes.

In Adventuring, the computer speaks and listens to plain English. No prior knowledge of computers, special controls, or games is required so everyone enjoys them—even people who do not like computers.

Except for Quest, itself unique among Adventure games, Adventures are non-graphic. Adventures are more like a novel than a comic book or arcade game. It is like reading a particular exciting book where you are the main character.

All of the Adventures in this ad are in Basic. They are full featured, fully plotted adventures that will take a minimum of thirty hours (in several sittings) to play.

Adventuring requires 16k on Sinclair, TRS-80, and TRS-80 Color. They require 8k on OSI and 13k on VIC-20. Sinclair requires extended BASIC. Now available for TI99.

TREK ADVENTURE by Bob Retelle — This one takes place aboard a familiar starship and is a must for trekkies. The problem is a familiar one — The ship is in a "decaying orbit" (the Captain never could learn to park!) and the engines are out (You would think that in all those years, they would have learned to build some that didn't die once a week). Your options are to start the engine, save the ship, get off the ship, or die. Good Luck.

Authors note to players — I wrote this one with a concordance in hand. It is very accurate — and a lot of fun. It was nice to wander around the ship instead of watching it on T.V.

DERELICT by Rodger Olsen and Bob Anderson — For Wealth and Glory, you have to ransack a thousand year old space ship. You'll have to learn to speak their language and operate the machinery they left behind. The hardest problem of all is to live through it.

Authors note to players — This adventure is the new winner in the "Toughest Adventure at Aardvark Sweepstakes". Our most difficult problem in writing the adventure was to keep it logical and realistic. There are no irrational traps and sudden senseless deaths in Derelict. This ship was designed to be perfectly safe for its' builders. It just happens to be deadly to alien invaders like you.

Dungeons of Death — Just for the 16k TRS-80 COLOR, this is the first D&D type game good enough to qualify at Aardvark. This is serious D&D that allows 1 to 6 players to go on a Dragon Hunting, Monster Killing, Dungeon Exploring Quest. Played on an on-screen map, you get a choice of race and character (Human, Dwarf, Soldier, Wizard, etc.), a chance to grow from game to game, and a 15 page manual. At the normal price for an Adventure ($14.95 tape, $19.95 disk), this is a giveaway.

PYRAMID by Rodger Olsen — This is one of our toughest Adventures. Average time through the Pyramid is 50 to 70 hours. The old boys who built this Pyramid did not mean for it to be ransacked by people like you.

Authors note to players — This is a very entertaining and very tough adventure. I left clues everywhere but came up with some ingenous problems. This one has captivated people so much that I get calls daily from as far away as New Zealand and France from bleary eyed people who are stuck in the Pyramid and desperate for more clues.

MARS by Rodger Olsen — Your ship crashed on the Red Planet and you have to get home. You will have to explore a Martian city, repair your ship and deal with possibly hostile aliens to get home again.

Authors note to players — This is highly recommended as a first adventure. It is in no way simple—playing time normally runs from 30 to 50 hours — but it is constructed in a more "open" manner to let you try out adventuring and get used to the game before you hit the really tough problems.

QUEST by Bob Retelle and Rodger Olsen — THIS IS DIFFERENT FROM ALL THE OTHER GAMES OF ADVENTURE!!!! It is played on a computer generated map of Alesia. You lead a small band of adventurers on a mission to conquer the Citadel of Moorlock. You have to build an army and then arm and feed them by combat, bargaining, exploration of ruins and temples, and outright banditry. The game takes 2 to 5 hours to play and is different each time. The TRS-80 Color version has nice visual effects and sound. Not available on OSI. This is the most popular game we have ever published.

32K TRS 80 COLOR Version $24.95.

Adds a second level with dungeons and more Questing.

PRICE AND AVAILABILITY:

All adventures are $14.95 on tape. Disk versions are available on VIC/COMMODORE and TRS-80 Color for $2.00 additional. $2.00 shipping charge on each order.

Please specify system on all orders

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

AARDVARK

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

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

TRS-80 COLOR TIMEX/SINCLAIR COMMODORE 64 VIC-20

$2.00 shipping on each order

Circle No. 1

Peek & Poke

On the Buss

by George R. Gaukel

This article will mainly apply to those people who have the MSD Model CIE Transparent Interface for the C64. I have included two listings of the same program. One is in BASIC and the other is an assembler listing, as some assemblers will not run properly until the IEEE primitive vectors and traps are installed.

This program will allow the running of applications that use the SERIAL primitives in the KERNAL jump table. The program, as written, makes several important assumptions:

(1) That the RAM under the KERNAL ROM is used by the application.

(2) That the Vectors starting at $0314. which point to IN/OUT functions, are not changed by the application.

(3) That the MSD coding at $9C00 to $9FFF is not used by the application.

To install the serial compatible primitives, the program does three things:

(1) Copies the BASIC and KERNAL ROMs to RAM. The coding we want to change is in the KERNAL ROM. In order to get the KERNAL into RAM, the BASIC also has to be copied.

(2) Alters the MSD coding so the TALK and LISTEN calls match the SERIAL syntax.

(3) Re-writes the vectors in the KERNAL jump table and installs vector traps for several non-standard KERNAL calls.

I have run the following software, with little or no modification, on the IEEE buss:

TOTL: TOTL TEXT2.6 with CHICKSPEED (A Text Processor).

Eastern House Software: MAE (Macro Assembler, Editor, Simpified Text Processor and other Utilities).

Commodore: The Commodore 64 Macro Assembler Development System.

The people at MSD say they are in the process of revising their coding to be more compatible with the serial

Program 1

MSD.DATA

```
100 REM 'MSD.DATA'
110 FOR AD=8192TO8466:READ DA
120 POKE AD,DA:NEXT
130 SYS8192 : NEW
140 :
150 DATA  32, 190, 223, 169,   0, 133
160 DATA 251, 168, 169, 224, 133, 252
170 DATA 177, 251, 145, 251, 230, 251
180 DATA 208, 248, 230, 252, 208, 244
190 DATA 169,   0, 133, 251, 169, 160
200 DATA 133, 252, 177, 251, 145, 251
210 DATA 230, 251, 208, 248, 230, 252
220 DATA 165, 252, 201, 192, 208, 240
230 DATA 162,   5, 189, 214,  32, 157
240 DATA 147, 255, 202,  16, 247, 162
250 DATA  17, 189, 220,  32, 157, 165
260 DATA 255, 202,  16, 247, 162,  35
270 DATA 160,  23, 185, 250,  32, 133
280 DATA 253, 136, 185, 250,  32, 133
290 DATA 252, 136, 132, 251, 160,   2
300 DATA 189, 214,  32, 145, 252, 202
310 DATA 136,  16, 247, 164, 251,  16
320 DATA 227, 169,   0, 133, 148, 169
330 DATA   9, 141, 135, 156, 141, 138
340 DATA 156, 169, 234, 141, 166, 156
350 DATA 141, 167, 156, 162, 192, 160
360 DATA 159, 142, 234, 157, 140, 235
370 DATA 157, 142, 194, 158, 140, 195
380 DATA 158, 142, 109, 159, 140, 110
390 DATA 159, 162, 197, 160, 159, 142
400 DATA  96, 159, 140,  97, 159, 162
410 DATA  10, 189, 204,  32, 157, 192
420 DATA 159, 202,  16, 247, 169, 148
430 DATA 141,  69, 156, 141, 145, 156
440 DATA 141, 159, 156, 141,  29, 157
450 DATA 141,  33, 157, 169, 149, 141
460 DATA 169, 156, 141, 193, 156, 141
470 DATA 247, 156, 141,  42, 157, 165
480 DATA   1,  41, 253, 133,   1,  96
490 DATA 165, 186,  76, 138, 156, 165
500 DATA 186,  76, 135, 156,  76, 246
510 DATA 156,  76, 237, 156,  76,  58
520 DATA 157,  76,  28, 157,  76,  45
530 DATA 157,  76,  51, 157,  76, 138
540 DATA 156,  76, 135, 156,  76, 221
```

primitives. They also have available information on relocating their code.

Not all SERIAL applications will operate with this fix. Some software writers believe that everyone is using a serial disk and the KERNAL RAM is free for their use. **BUYER BEWARE!!!** Make sure you can get your money back if the package is not KERNAL standard for alternate interfaces. Some packages may specifically be designed for the various interfaces. Check with the software producer for compatible versions. The point is that a well designed serial package which uses only the standard jump table and allows the drive number to be entered with the file name, (Dn:Filename) will usually operate just fine with alternate disk system installed.

Another problem that may pop up is where the application uses the BASIC RAM. This should not cause any problems, unless the application clears the LORAM bit without checking to see if the HIRAM bit is cleared. The result of this little bungle is the loss of all the ports and latches. Also as wearisome is the application that resets HIRAM while clearing the LORAM bit.

The market place will determine the final outcome of the RAM under KERNAL ROM range wars. It is understandable that the software producer will try to use all the available RAM to increase the effectiveness and speed of his application. However, if speed is a prime consideration, the compatibility to the IEEE-488 interfaces could be a strong selling point. Also, there is the fact that Commodore installed the standard jump table to accomodate ROM revisions, alternate IN/OUT configurations. They have also stated that this table will be propagated into future computers for software compatibility.

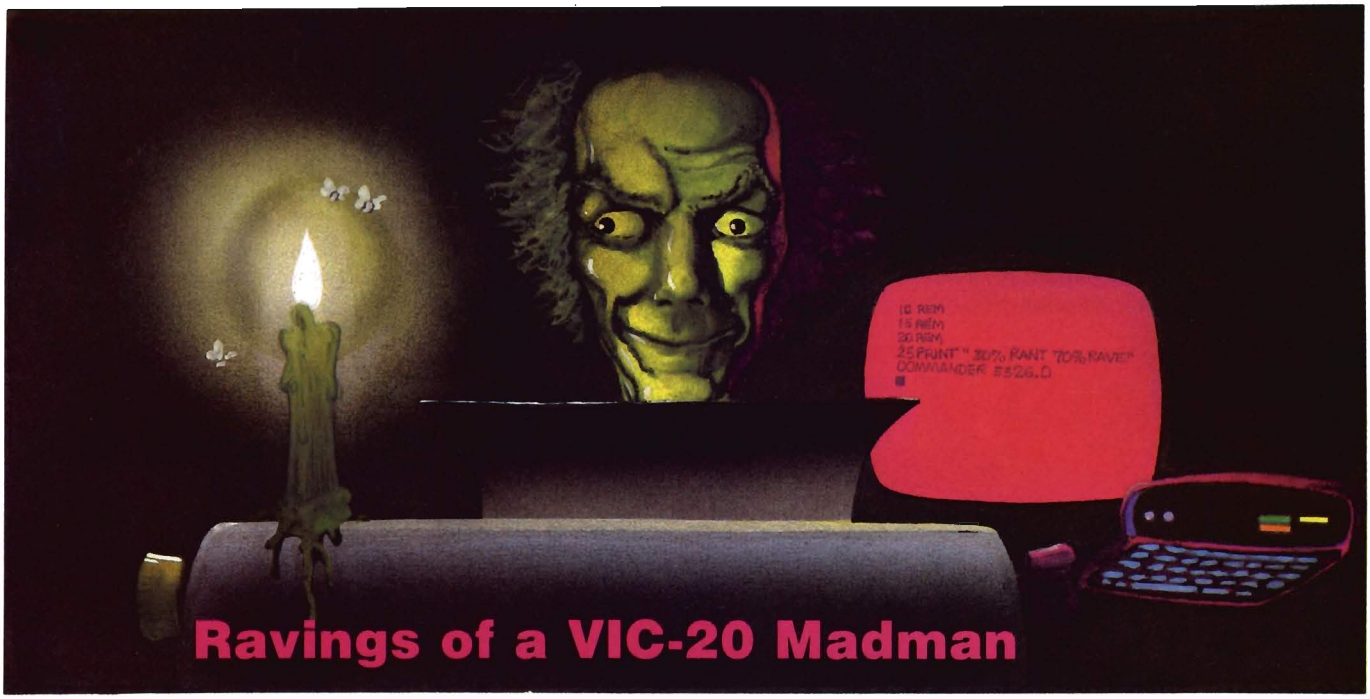
In summary, there are some pitfalls to be aware of if you buy a NON-BASIC application to be used with the 488 buss. In this article I have demonstrated how easy it is to patch a well designed serial program to a 488 interface, which was not designed to run low-level applications, and still get usable results. Some applications will just be impossible to patch.□

```
550 DATA 157,  76, 189, 158,  76, 180
560 DATA 156,  76, 251, 156, 185, 237
570 DATA 199, 237,  19, 238, 221, 237
580 DATA 239, 237, 254, 237,  12, 237
590 DATA   9, 237, 213, 243,  66, 246
600 DATA  54, 237, 190, 237, 234
```

Program 2

```
0010  .LS
0020 ;  'MSD.FIX.ASM'
0030
0040 ;  COPYRIGHT 1983 G.R.GAUKEL
0050 ;  PATCHES IN IEEE PRIMITIVES
0060 ;  FOR THE MSD INC. MODEL CIE
0070 ;  IEEE TRANSPARENT INTERFACE
0080
0090  .BA $2000
0100  .OC
0110  .CE
0120
0130
0140 ;DATA REGISTER
0150 R6510   .DE $01
0160 ;OUTPUT FLAG C64
0170 KC3PO   .DE $94
0180 ;OUTPUT CHAR C64
0190 KSOUR   .DE $95
0200 ;CURRENT DEV#
0210 FA      .DE $BA
0220 ;OUTPUT FLAG MSD
0230 EC3PO   .DE $FB
0240 ;OUTPUT CHAR MSD
0250 ESOUR   .DE $FC
0260
0270
0280 ; KERNAL JUMPS
0290
0300 KSECND .DE $FF93
0310 KTKSA  .DE $FF96
0320
0330 KACPTR .DE $FFA5
0340 KCIOUT .DE $FFA8
0350 KUNTLK .DE $FFAB
0360 KUNLSN .DE $FFAE
0370 KLISTN .DE $FFB1
0380 KTALK  .DE $FFB4
0390
0400
0410 ; KERNAL SERIAL INTERNALS
0420
0430 SSECND .DE $EDB9
0440 STKSA  .DE $EDC7
0450
0460 SACPTR .DE $EE13
0470 SCIOUT .DE $EDDD
```

Continued on page 107



by Tim Parker
Ontario, Canada

There has recently been quite a bit of activity on the VIC-20 front. Many new (and some quite interesting) bits of hardware and software have cropped up. Employing these can make the VIC-20 into a powerful machine.

At least two companies (including Commodore) have introduced cases for the VIC that have the VIC-20 or VIC-64 slip into a molded steel or plastic enclosure with expansion slots built into the rear compartments. Holes are cut for the side mounted ports and on/off switches. After playing with two versions, I can definitely say that they simplify the clutter problem most computer owners have. The VIC-20 sits in the case with its memory and a few utility ROMs mounted on the expansion board, and the monitor I use (color, or course) sits on top. All in all, it looks like an Apple with monitor on the case. However, it will never be confused as such.

A while ago I mentioned a few new hardware expansion options that allow Atari VCS games to be played on the VIC. After conducting extensive and intensive research (I dedicated my firing thumb to the cause of science) I can report they work quite well. As noted in the previous column, they do not task the VIC's capabilities at all, and some of the graphics look downright ridiculous as block figures when compared to VIC versions, there are some games that are still not available in VIC cartridge form. Also, as many of the video games for the Atari VCS get more sophisticated, the games become more of a pleasure to play. As these gadgets cost less than one hundred bucks, they seem to be an easy way to get into video games in a major way.

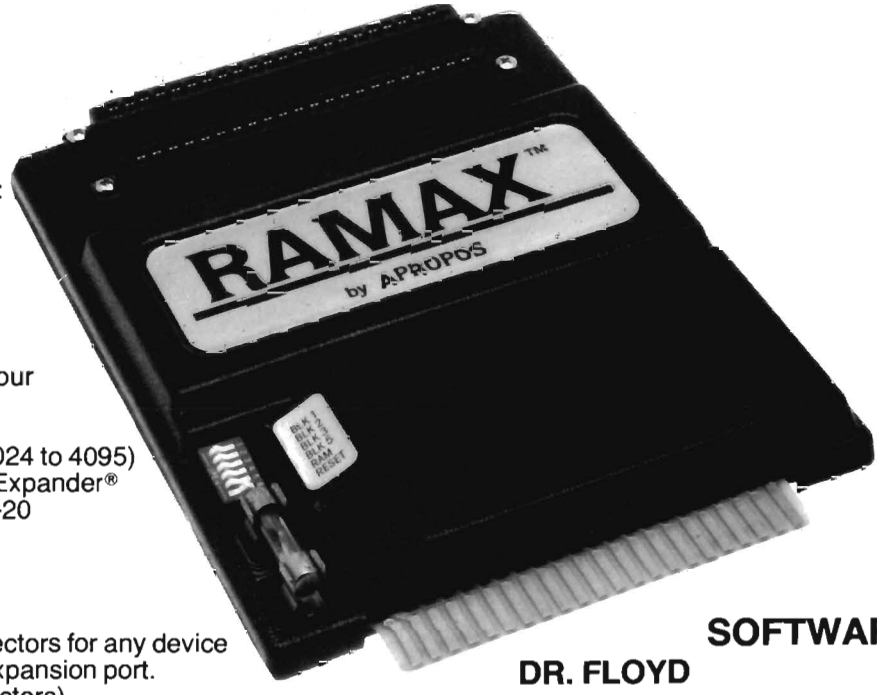
After visiting my local Commodore dealer, I was somewhat astonished to see the staggering array of new games available for the VIC-20. The shelves that used to house only Commodore, UMI and a couple of other company products now has blossomed to overstocking with almost every type of game conceivable. Although I don't play games that much these days, I felt it was my duty as a correspondent with Commander to give a few the old workout. Although the majority were only mediocre (i.e. a variation on a previously published theme), there were a few suprisingly good entries. I will be reviewing some of these in future columns for this erudite journal. There were also a few atrocities. These will not be reviewed, as the best things about them was the cover art.

Diversion time again: it is very difficult to get a magazine of any type, computer or otherwise, to publish totally negative reviews. Although most reviewers come across turkeys now and again, and although we feel that the buying public should be made aware of some of the wastes of money, it does not look good to have a review in a magazine that decimates a product. As a rule, most reviewers, if they don't like the item in question, will try and find something good to say, simply so the review is at least partially optimistic. No such luck with some upcoming reviews! I have taken it upon myself to compile a list of "Turkeys" for the VIC-20 that should never have seen the dealer's shelf. That will be upcoming some year. End of diversion.

Of all the games examined of late, at least sixty percent are "shoot-em-ups." This perennial type of game also seems to have the shortest attention span for most people (unless, of course, the game is very good, or an obsession sets in . . . usually the latter). The treatments are getting more clever and more varied, but it all comes down to the same thing: hit the fire button as many times as you possibly can without killing the thumb, finger, or foot. (Sideline: one game I play frequently uses the space bar for firing and supports the full repeat feature,

RAMAX™

by APROPOS
The **ONLY** MEMORY your **VIC-20®** will need



FEATURES

- A full 27k bytes of RAM (added to VICs 5k equals 32k.)
- Fully switchable in sections:
  - BLK 1 switches 8k (Adr. 8192 to 16383)
  - BLK 2 switches 8k (Adr. 16384 to 24575)
  - BLK 3 switches 8k (Adr. 24576 to 32767)
  - BLK 5 allows/disallows your 8k ROM (games) (Adr. 40960 to 49152)
  - RAM switches 3k (Adr. 1024 to 4095)
  - May be used with Super Expander® games or **ANY** other VIC-20 compatible cartridge.
- Built in RESET switch.
- Fuse protected.
- Totally self-contained.
- 2 duplicate extension connectors for any device normally plugged into the expansion port. (BLK 5 is switched to connectors)
- Very low power usage. (.150 amp max.)
- High reliability gold plated connectors.
- 6 month parts and labor warranty.
- Factory service. - Extended service always available.

THIS SUPERB PLUG-IN GIVES YOUR VIC-20 REAL POWER AND EXPANDABILITY

FOR ONLY $149.00 Shipping included

10 DAY SATISFACTION OR YOUR MONEY BACK GUARANTEE

Already own an 8k Expander? The new 19k **RAMAX JR.™** allows you to use your 8k Expander as BLK 3 to get the full compliment of memory. Complete instructions included. Only **$129**, Includes shipping.

WE SERVICE WHAT WE SELL

TO ORDER:

Send Check or Money Order For the Total
Calif. residents add 6% tax.

Phone orders: CALL **(805) 482-3604 24 HRS.**
For credit card orders, include all information on card.
or **contact your local dealer.**

VISA Foreign orders, add $15.00. MasterCard

All items shipped from stock.
DEALER INQUIRIES WELCOME

SOFTWARE

DR. FLOYD

Psychoanalysis by computer? — well, not quite, but Dr. Floyd will carry on a conversation with you using psychoanalytic techniques giving the appearance of artificial intelligence. Requires 16k RAM or more.
$14.95 shipping included.

WORD PLAY

"WORDPLAY" is a collection of programs which allow the user to make original stories, write a form of Japanese poetry, play the fun game of Animal (children love this one), and create jargon. A bonus secret message (cypher) program is also included. In a word, "WORDPLAY" is a bargain.
Requires 16k RAM or more.
$14.95 shipping included.

TYPE FOR YOUR LIFE

With more challenge than an arcade game, learn to type up to 75+ words/min. (User selectable, but no FOOLING AROUND allowed). TEXT IS WIDELY VARIED SINCE IT COMES FROM THE PROGRAM TAPE. Action color graphics with sound fix your eyes to the screen (away from your fingers - clever!) Your man rows your boat up stream as fast as you can type. Maintain speed and destroy the Sea Monster; slow down and he will get you. Runs on the unexpanded VIC.
$14.95 shipping included.

All software is on high quality cassettes and is replacement guaranteed.

350 N. Lantana Ave., Suite 821
Camarillo, CA 93010

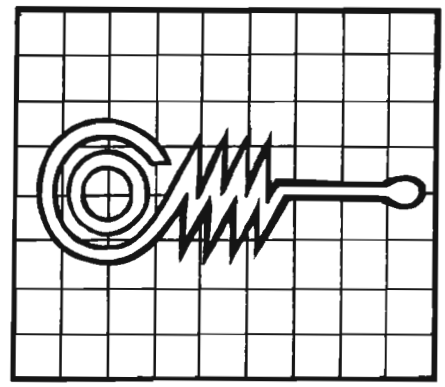
APROPOS TECHNOLOGY

while the joystick fire button doesn't repeat. For maximum score, my hands work the joystick to move my image on the screen, and my foot holds down the space bar for automatic repeat fire. It's crude, yes. It may even be unhygenic. The scores I get are terrific!)

A growing percentage of games are the thinking types, though, which always signifies that the computer it is intended for has gone beyond the video game genre. That can only be to the attraction of the VIC-20. A few are chess programs, and the backgammon, checkers, go, reversal, etc. types. These as I have stated before in this column are good for the thinking types who like to spend two hours playing chess. (One of them plays an excellent game: a review should follow in a month or two.) The change is that there are those games that are neither shoot-em-ups or chess genre. Games that require strategy other than placing the ship in the right position to blast the creepy bugs, and still require reflexes and ability. Most are modeled on existing video games in the arcades, and some are done with surprising integrity. What it all comes down to in the final assessment is that there is now a computer game for everyone. (By the way, adventure-type games have multiplied too, but I have the feeling that after playing a couple, they all are the same! And on top of that, I really detest the silly games that require totally illogical actions to accomplish a goal, such as tossing rocks into a barrel while standing on your head in order to open a door. They may amuse some: to me they are a waste of time.)

Also proliferating are the more serious sides of programming. Word processors have been showing up in increasing numbers, and a large number of companies offer packages that handle spreadsheets, accounting procedures, maintenance schedules, mailing lists, etc. Useful stuff for business and home alike. There is also the expanding number of packages for specific purposes such as insurance schedules, or tax aids.

Teaching or educational software has likewise increased. PILOT and other languages have been increasing in number, and all will help the newcomer and youngster to the computer field. The more people who can understand computers and not feel oppressed by them, the better. In an increasingly computerized world, it is useful to have a knowledge of what is going on. That wraps up the philosophy class. Time for a drink.□



CREATIVE SOFTWARE

– the #1* independent VIC-20 full-line software publisher in the U.S. – is proud to announce 4 new Game Cartridges & 5 Home Applications for the COMMODORE 64.™

CREATIVE SOFTWARE PRESENTS

A GAME ON CARTRIDGE FOR THE COMMODORE 64™

MOONDUST™

Joystick controller required.

CREATIVE SOFTWARE PRESENTS

A GAME ON CARTRIDGE FOR THE COMMODORE 64™

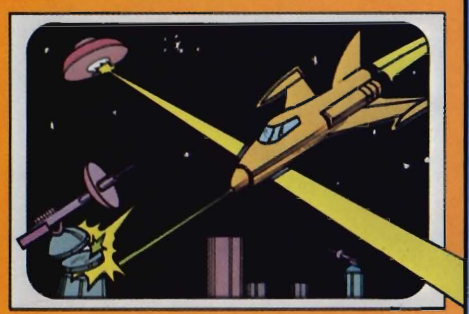
SAVE NEW YORK

Joystick controller required.

CREATIVE SOFTWARE PRESENTS

ASTROBLITZ

A GAME ON CARTRIDGE FOR THE COMMODORE 64™



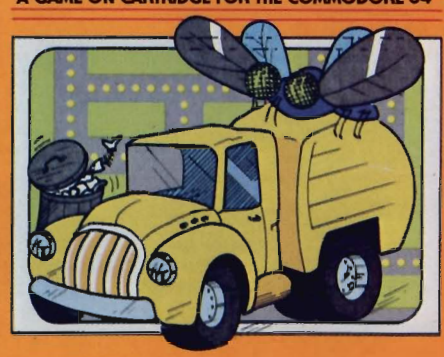
Joystick controller required.

THIS GAME IS ALSO AVAILABLE FOR THE VIC-20.

CREATIVE SOFTWARE PRESENTS

TRASHMAN

A GAME ON CARTRIDGE FOR THE COMMODORE 64™

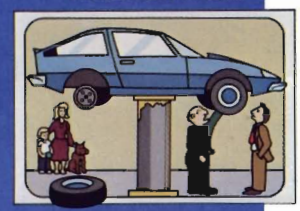


Joystick controller required.

THIS GAME IS ALSO AVAILABLE FOR THE VIC-20.

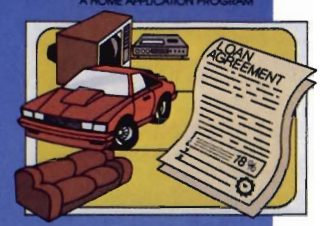
CAR COSTS

A HOME APPLICATION PROGRAM



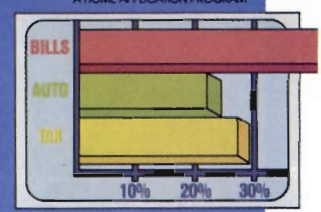
LOAN ANALYZER

A HOME APPLICATION PROGRAM



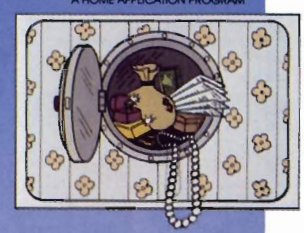
HOUSEHOLD FINANCE

A HOME APPLICATION PROGRAM



HOME INVENTORY

A HOME APPLICATION PROGRAM



DECISION MAKER

A HOME APPLICATION PROGRAM

These Home Application Programs are also available for the VIC-20.

CREATIVE SOFTWARE
A Division of ASCI, Inc.
230 East Caribbean Drive
Sunnyvale, CA 94086

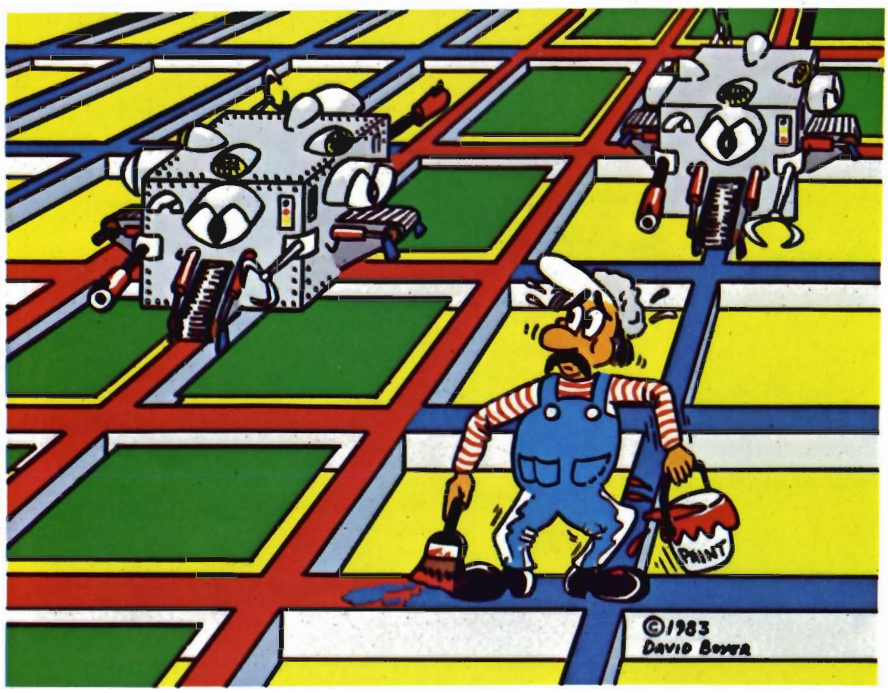
*Based on survey of distributors and retailers.

PEEK AND POKE—continued from page 101

```
                    0480 SUNTLK .DE $EDEF
                    0490 SUNLSN .DE $EDFE
                    0500 SLISTN .DE $EDOC
                    0510 STALK  .DE $ED09
                    0520
                    0530 SOPEN  .DE $F3D5
                    0540 SCLOSE .DE $F642
                    0550 SDCIOU .DE $ED36
                    0560 SCATTN .DE $EDBE
                    0570
                    0580
                    0590 ; MSD IEEE INTERNALS
                    0600
                    0610 ESECND .DE $9CF6
                    0620 ETKSA  .DE $9CED
                    0630
                    0640 EACPTR .DE $9D3A
                    0650 ECIOUT .DE $9D1C
                    0660 EUNTLK .DE $9D2D
                    0670 EUNLSN .DE $9D33
                    0680 ELISTN .DE $9C8A
                    0690 ETALK  .DE $9C87
                    0700
                    0710 EOPEN  .DE $9DDD
                    0720 ECLOSE .DE $9EBD
                    0730 EDCIOU .DE $9CB4
                    0740 ECATTN .DE $9CFB
                    0750
                    0760
                    0770 ;GET GOOD COPY MSD
                    0780 START
2000- 20 BE DF      0790  JSR 57278
                    0800
                    0810 ;COPY ROMS TO RAM
                    0820 COPY
                    0830 ;KERNAL ROM
2003- A9 00         0840  LDA #L,$E000
                    0850 ;TEMP POINTER
2005- 85 FB         0860  STA *EC3PO
2007- A8            0870  TAY
2008- A9 E0         0880  LDA #H,$E000
200A- 85 FC         0890  STA *EC3PO+1
                    0900 KERRAM
200C- B1 FB         0910  LDA (EC3PO),Y
200E- 91 FB         0920  STA (EC3PO),Y
2010- E6 FB         0930  INC *EC3PO
2012- D0 F8         0940  BNE KERRAM
2014- E6 FC         0950  INC *EC3PO+1
2016- D0 F4         0960  BNE KERRAM
                    0970
                    0980 ;BASIC ROM
2018- A9 00         0990  LDA #L,$A000
201A- 85 FB         1000  STA *EC3PO
201C- A9 A0         1010  LDA #H,$A000
201E- 85 FC         1020  STA *EC3PO+1
```

continued on page 110

Game—CONTEST



Terms for Game Contest

First prize will be awarded to the person with the highest score. The winning entry must contain a photograph of the highest score of the game along with a GRIDDER package front and proof of purchase.

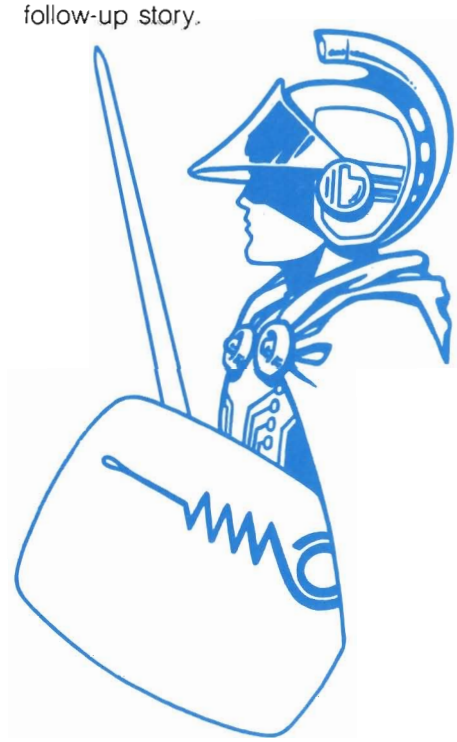
Entries must be mailed to Commander, GRIDDER Contest, PO Box 98827, Tacoma, WA 98498. All entries must be mailed, as postmarks are required to determine the earliest winning entry. In the event of a tie, duplicate prizes will be awarded. Employees of MICRODIGITAL and their families may not participate. First prize will be $500, second prize $200, third prize $100 in merchandise.

The contest will run until September 31. MICRODIGITAL will notify Commander Magazine of the winner(s) and provide copy and photographs for a follow-up story.

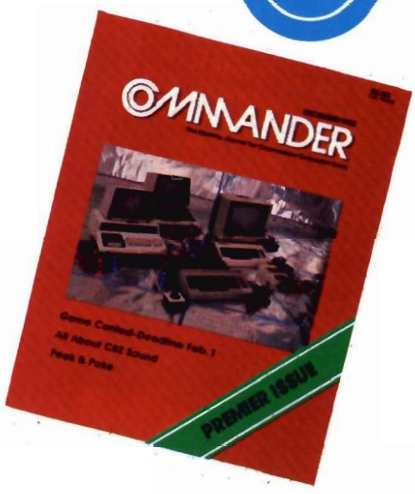
The Game Contest is a continuing feature of Commander Magazine aimed at providing entertainment for and promoting competition among our readers. MICRODIGITAL has graciously provided us with this Game Contest.

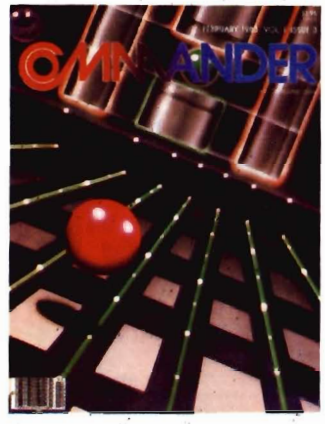
DEADLINE FOR ENTRIES
SEPTEMBER 1, 1983

GRIDDER may be purchased from
MICRODIGITAL.
752 John Glenn Blvd.
Webster, NY 14580
1-716-872-0518 inside New York
1-800-833-7384 toll free outside of New York



COMMANDER





Only COMMANDER will do all this

GUIDE Reports that give you hands-on, user oriented information on the entire range of Commodore Computers • More comparison charts, diagrams, instructional illustrations, and how-to photos than any other publication in the field • Articles written by the top working experts in the field.

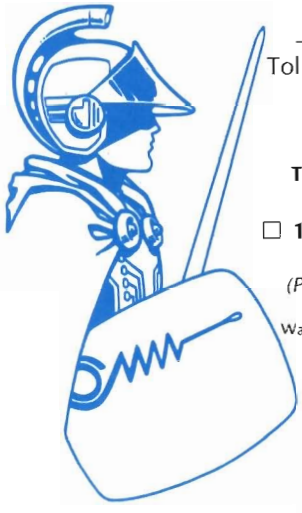
INSTRUCT Innovative ways to use your computer equipment—from self-improvement to home security • Valuable tricks and practical advice on how to get the most from your investment in the high tech field of computers • In depth and unbiased analyses of the latest in educational software.

ENTERTAIN Fascinating glimpses into the wonders of the computer future • Profiles of the industry's most creative people • Provocative discussions on the impact computers will have on you and your family.

SIMPLIFY Critical comparisons of the Commodore Computers • Exactly how your computer equipment works • What those computer "buzz words" mean and expensive "special features" do • Useful trouble-shooting & maintenance advice.

INFORM An expanded directory of the latest programs on ROM-PAK, tape and disc • New product previews of the latest and best equipment on the market • Exciting technologies in development for tomorrow • Important applications of computers for health, education, business, and the public good • New directions in computers • Much More . . .

Subscribe now and take advantage of our Limited Offer of $4 off the one year subscriptions



—Subscription Orders Only—
Toll Free Number: 1-800-426-1830
(except WA, HI, AK)

COMMANDER
PO BOX 98827
TACOMA, WASHINGTON 98498
(206) 584-6757

☐ **1 YR. $22** ☐ **2 YR. $40**
☐ **3 YR. $58**

(Prices do not include $4 discount)
Prices in U.S. Funds
Washington residents please add 7.8% sales tax
Allow 5-6 weeks for delivery.

DON'T MISS OUT
on the most informative magazine dedicated to the COMMODORE COMPUTERS
HAVE YOU GOT WHAT IT TAKES TO BE A

COMMANDER

PEEK AND POKE—continued from page 107

```
                        1030 BASRAM
2020- B1 FB             1040  LDA (EC3PO),Y
2022- 91 FB             1050  STA (EC3PO),Y
2024- E6 FB             1060  INC *EC3PO
2026- D0 F8             1070  BNE BASRAM
2028- E6 FC             1080  INC *EC3PO+1
202A- A5 FC             1090  LDA *EC3PO+1
202C- C9 C0             1100  CMP #$C0
202E- D0 F0             1110  BNE BASRAM
                        1120
                        1130
                        1140 ; FIRST SET THE KERNAL
                        1150 ; JUMP TABLES - SO THERE
                        1160 ; IS NO SPEED DELAY FOR
                        1170 ; CORRECT CALLS.
                        1180
                        1190
2030- A2 05             1200  LDX #TAB2-TAB1-1
                        1210 KERN1
2032- BD D6 20          1220  LDA TAB1,X
2035- 9D 93 FF          1230  STA KSECND,X
2038- CA                1240  DEX
2039- 10 F7             1250  BPL KERN1
                        1260
203B- A2 11             1270  LDX #TAB3-TAB2-1
                        1280 KERN2
203D- BD DC 20          1290  LDA TAB2,X
2040- 9D A5 FF          1300  STA KACPTR,X
2043- CA                1310  DEX
2044- 10 F7             1320  BPL KERN2
                        1330
                        1340
                        1350 ; NONSTANDARD CALLS
                        1360
                        1370
2046- A2 23             1380  LDX #TAB4-TAB1-1
2048- A0 17             1390  LDY #TAB5-TAB4-1
                        1400 CATCH1
                        1410 ;ADDRESS HI
204A- B9 FA 20          1420  LDA TAB4,Y
204D- 85 FD             1430  STA *ESOUR+1
204F- 88                1440  DEY
                        1450 ;ADDRESS LO
2050- B9 FA 20          1460  LDA TAB4,Y
2053- 85 FC             1470  STA *ESOUR
2055- 88                1480  DEY
                        1490 ;SAVE (Y)
2056- 84 FB             1500  STY *ESOUR-1
                        1510 ;COPY 3 BYTES
2058- A0 02             1520  LDY #$02
                        1530 CATCH2
205A- BD D6 20          1540  LDA TAB1,X
205D- 91 FC             1550  STA (ESOUR),Y
205F- CA                1560  DEX
2060- 88                1570  DEY
2061- 10 F7             1580  BPL CATCH2
```

```
                        1590 ;GET (Y)
2063- A4 FB             1600  LDY *ESOUR-1
2065- 10 E3             1610  BPL CATCH1
2067- A9 00             1620  LDA #$00
                        1630 ;FLAG EMPTY
2069- 85 94             1640  STA *KC3PO
                        1650
                        1660
                        1670 ; FIX MSD LISTEN & TALK
                        1680
                        1690 ;ORA OPC
206B- A9 09             1700  LDA #$09
206D- 8D 87 9C          1710  STA ETALK
2070- 8D 8A 9C          1720  STA ELISTN
                        1730 ;NOP OPC
2073- A9 EA             1740  LDA #$EA
2075- 8D A6 9C          1750  STA $9CA6
2078- 8D A7 9C          1760  STA $9CA7
                        1770
                        1780
                        1790 ; PATCH MSD CALLS
                        1800
                        1810 ;LISTEN PATCH
207B- A2 C0             1820  LDX #L,$9FC0
207D- A0 9F             1830  LDY #H,$9FC0
207F- 8E EA 9D          1840  STX $9DEA
2082- 8C EB 9D          1850  STY $9DEB
2085- 8E C2 9E          1860  STX $9EC2
2088- 8C C3 9E          1870  STY $9EC3
208B- 8E 6D 9F          1880  STX $9F6D
208E- 8C 6E 9F          1890  STY $9F6E
                        1900
                        1910 ;TALK PATCH
2091- A2 C5             1920  LDX #L,$9FC5
2093- A0 9F             1930  LDY #H,$9FC5
2095- 8E 60 9F          1940  STX $9F60
2098- 8C 61 9F          1950  STY $9F61
                        1960
                        1970 ;COPY THE PATCH
209B- A2 0A             1980  LDX #TAB1-CODEP
                        1990 CODEL
209D- BD CC 20          2000  LDA CODEP,X
20A0- 9D C0 9F          2010  STA $9FC0,X
20A3- CA                2020  DEX
20A4- 10 F7             2030  BPL CODEL
                        2040
                        2050 ;FIX MSD C3PO
20A6- A9 94             2060  LDA #KC3PO
20A8- 8D 45 9C          2070  STA $9C45
20AB- 8D 91 9C          2080  STA $9C91
20AE- 8D 9F 9C          2090  STA $9C9F
20B1- 8D 1D 9D          2100  STA $9D1D
20B4- 8D 21 9D          2110  STA $9D21
                        2120
                        2130 ;FIX MSD BSOUR
20B7- A9 95             2140  LDA #KSOUR
20B9- 8D A9 9C          2150  STA $9CA9
```

continued on page 112

PEEK AND POKE—continued from page 111

```
20BC- 8D C1 9C   2160  STA $9CC1
20BF- 8D F7 9C   2170  STA $9CF7
20C2- 8D 2A 9D   2180  STA $9D2A
                 2190
                 2200
                 2210
                 2220 ;SWITCH ROMS OUT
                 2230
                 2240 IEEE
20C5- A5 01      2250  LDA *R6510
                 2260 ;HIROM CLR
20C7- 29 FD      2270  AND #%11111101
20C9- 85 01      2280  STA *R6510
                 2290
                 2300 ;FINISHED
20CB- 60         2310  RTS
                 2320
                 2330
                 2340 CODEP
20CC- A5 BA      2350  LDA *FA
20CE- 4C 8A 9C   2360  JMP ELISTN
20D1- A5 BA      2370  LDA *FA
20D3- 4C 87 9C   2380  JMP ETALK
                 2390
                 2400 ; *** FIXUP TABLES ***
                 2410
                 2420 TAB1
20D6- 4C F6 9C   2430  JMP ESECND
20D9- 4C ED 9C   2440  JMP ETKSA
                 2450 TAB2
20DC- 4C 3A 9D   2460  JMP EACPTR
20DF- 4C 1C 9D   2470  JMP ECIOUT
20E2- 4C 2D 9D   2480  JMP EUNTLK
20E5- 4C 33 9D   2490  JMP EUNLSN
20E8- 4C 8A 9C   2500  JMP ELISTN
20EB- 4C 87 9C   2510  JMP ETALK
                 2520 TAB3
20EE- 4C DD 9D   2530  JMP EOPEN
20F1- 4C BD 9E   2540  JMP ECLOSE
20F4- 4C B4 9C   2550  JMP EDCIOU
20F7- 4C FB 9C   2560  JMP ECATTN
                 2570 TAB4
20FA- B9 ED      2580  .SE SSECND
20FC- C7 ED      2590  .SE STKSA
20FE- 13 EE      2600  .SE SACPTR
2100- DD ED      2610  .SE SCIOUT
2102- EF ED      2620  .SE SUNTLK
2104- FE ED      2630  .SE SUNLSN
2106- 0C ED      2640  .SE SLISTN
2108- 09 ED      2650  .SE STALK
210A- D5 F3      2660  .SE SOPEN
210C- 42 F6      2670  .SE SCLOSE
210E- 36 ED      2680  .SE SDCIOU
2110- BE ED      2690  .SE SCATTN
2112- EA         2700 TAB5 NOP
                 2710                EN
```

MICROCOMPUTERS AND EDUCATION

by Authur J. Dudley
Bremerton, WA

As Microcomputers become increasingly popular, more and more arcade games are flooding the market. Not until recently has the interest shifted towards the educational aspects. This is not to say that software publishers were unaware of the importance of microcomputers in education, but instead felt there was a greater demand for games. I believe this emphasis is beginning to shift. People are not buying microcomputers solely for their entertainment value, but for their practical value as well. The two major uses of microcomputers in the practical sense are home management/ business applications and education.

The microcomputer, coupled with good software, can be invaluable as an instructional device for teachers and parents. It has unlimited patience, is interesting, and never tires. Two big advantages microcomputers have over textbooks are their ability to animate and randomize. The first advantage maintains the student's interest and reinforces learning; the second provides him/her with an unlimited number of combinations. However, for microcomputers to be effective, educational software packages must be carefully planned, well organized, and hold the child's interest. Many promising educational techniques and aids failed because they lacked these very qualities.

If you are planning to buy or develop educational software packages, the below characteristics will be of interest to you.

Subject Matter

Subject matter, believe it or not, is the most important part of the package. Fantastic graphics, smooth animation, and marvelous sound effects are worthless from an educational viewpoint if the program lacks quality subject matter. The topic must be concise, accurate, and have sufficient detail and quantity so in fact the child has something to learn. Be wary of products with vague catalog descriptions.

Organization

Good educational packages will take the student through a process step by step—first explaining the subject matter and then allowing the student to practice. Problem sets completed by students should be evaluated and briefly analyzed by the student to advanced levels or again through present levels of difficulty. It is extremely important that students master a section before proceeding.

Reinforcement

Positive reinforcement is essential. Students should be praised and encouraged throughout the course of the program. It is beneficial to use graphics and sound as rewards for correct answers. However, programs should not get carried away with this. Positive reinforcement should be geared to the student's age level. Also, programs that provide graphics and sound effects for wrong answers may unintentionally encourage students to answer incorrectly.

Graphics/Sound

Remember this is a supplement to enhance the student's interest and reinforce learning, not to totally dominate the program. If graphics and sound effects are to be used heavily, they must be directly related to the subject matter (i.e. using figures to teach children how to count, use of various shapes in geometry, use of sound to teach children musical notes, etc). Using graphics and sound extensively as rewards or as non-essential elements of the program take up valuable memory space that can be a powerful part of any educational program if correctly used.

Difficulty

The topic must be geared to the proper age level. (This should be identified in catalog descriptions). As a further note, the student must be able to understand the operating instructions of the program. One does not design a program having a set of operating instructions comparable to VISICALC to teach four year-olds to count. Young

COMPU SENSE

C-64	**Compute**	$399.00
VIC-20®	**Personal Computer**	147.00
VIC-1515	**Printer**	334.95
VIC-1530	**Datasette**	67.50
VIC-1541	**Disk Drive**	347.00
VIC-1010	**Expansion Module**	139.95
VIC-1311	**Joystick**	9.95
VIC-1312	**Game Paddles**	19.95
	Telephone Modem	99.95
VIC-1210	**VIC 3K Memory Expander Cart.**	34.95

Plugs directly into the VIC's expansion port. Expands to 8K RAM total.

VIC-1110 **VIC 8K Memory Expander Cart.** 52.50
8K RAM expansion cartridge plugs directly into the VIC.

CM102 **24K Memory Expander Cart.** 119.95

VIC-1011A **RS232C Terminal Interface** 39.95
Provides interface between the VIC-20 and RS232 telecommunications modems. Connects to VIC's user port.

PETSPEED - Basic Compiler for Commodore 140.00
Compile any Pet Basic program. The only optimizing compiler. Programs compiled with Petspeed run up to 40 times faster. Petspeed code is unlistable and compiled programs cannot be tampered with. No security device required for compiled programs. Available NOW for the Commodore 64.

Star Gemini 10 Printer 360.00
Star Gemini 15 Printer 450.00
SND Monitor 347.00

CS1 **QUICK BROWN FOX** $55.00
The Word Processor of this decade! For the VIC-20 and C-64.

COMMODORE SOFTWARE

VIC-1211A **VIC-20 Super Expander** $55.00
Everything Commodore could pack into one cartridge - 3K RAM memory expansion, high resolution graphics plotting, color, paint and sound commands. Graphic, text, multicolor and music modes. 1024x1024 dot screen plotting. All commands may be typed as new BASIC commands or accessed by hitting one of the VIC's special function keys. Includes tutorial instruction book. Excellent for all programming levels.

VIC-1212 **Programmer's Aid Cartridge** $45.99
More than 20 new BASIC commands help new and experienced programmers renumber, trace and edit BASIC programs. Trace any program line-by-line as it executes, pause to edit. Special KEY command lets programmers redefine function keys as BASIC commands, subroutines or new commands.

VIC-1213 **VICMON Machine Language Monitor** $48.99
Helps machine code programmers write fast, efficient 6502 assembly language programs. Includes one line assembler/disassembler.

CARDCO

Atari Adapter - play your 2600 games on the VIC-20 $79.95

CARDBOARD 6 $87.50
An expansion interface for the VIC-20. Allows expansion to 40K or accepts up to six games. May be daisy-chained for more versatility.

CARDBOARD 3 $35.95
Economy expansion interface for the Vic-20.

CARD "?" CARD/PRINT $76.00
Universal Centronics Parallel Printer Interface for the VIC-20 or CBM-64. Use an Epson MX-80 or OKIDATA or TANDY or just about any other.

CARDETTE $30.95
Use any standard cassette player/recorder with your VIC-20 or CBM-64.

LIGHT PEN $29.95
A light pen with six good programs to use with your VIC-20 or CBM-64.

16K Memory Expander $50.50

All CARDCO Products have a lifetime warranty.

BUSINESS USES FOR YOUR VIC-20®

SS	**Accounts Payable & Receivable**	$29.95
CW-107A	**Home Calculation Program Pack**	48.95
CPV-31	**Data Files** - your storage is unlimited	14.95
CPV-96	**Household Finance Package** - to keep records of all your household expenses	30.95
CPV-208	**Bar-Chart** - display your numerical data	8.95
CH	**Turtle Graphics** - learn programming	34.95
CH	**VIC Forth** - a powerful language for BASIC programming	49.95
CH	**HES MON** - a 6502 machine language monitor with a mini-assembler	34.95
CH	**HES Writer** - time-saving word process tool	34.95
CH	**Encoder** - keep your personal records away from prying eyes	34.95
CT-21	**Statistics Sadistics** - statistical analysis	14.95
CT-121	**Total Time Manager 2.0** - creates personal or business schedules	15.95
CT-124	**Totl Label** - a mailing list & label program	13.95
CT-125	**Totl Text BASIC**	15.95
CT-126	**Research Assistant** - keep track of data	17.50
CT-140	**Totl Text Enhanced**	29.95
CM-152	**Grafix Designer** - design graphic characters	12.95
CQ-5	**Minimon** - allows you to program, load, save, or execute machine language programs	13.95
CS	**Home Inventory** - lists your belongings	17.95
CS	**Check Minder** - keep your checkbook right	14.95
CS	**General Ledger** - a complete ledger	19.95

GAMES FOR YOUR VIC-20®

CH-G203	**Tank Wars**	$15.95
CH-G205	**Pinball**	13.45
CH-G206	**Simon** - It gets tougher as you get better. Great for kids of all ages.	13.45
CH-G207	**Fuel Pirates**	13.45
CH-G209	**Laser Blitz**	15.95
CH-G210	**Tank Trap**	15.95
CH-G211	**Concentration**	13.45
CH-G212	**Dam Bomber** - pilot the plane, avoid enemy	13.45
CH-C307	**Shamus** - search room after room for the shadow-eluding androids; 2 levels of intense arcade action	34.95
CH-C308	**Protector**	36.95
CPU-79	**Breakout**	7.95
CPU-85	**Hangman** - unbelievable graphics & sound	9.95
CPU-87	**Memory** - VIC challenges your memory	9.95
CPU-88	**Match** - hand & eye coordination	7.95
CPU-89	**Monks** - a devilish game of logic	7.95
CPU-108	**Bomber** - you must decide who you want to fly for, then pick a target & your experience level	9.95
CPU-109	**Amok** - the halls of Amok are populated by robots that obey one command - get the intruder!	20.95
CPU-153	**Tank vs. UFO** - the tank is moving back & forth along the base; shoot the UFO before it shoots you	9.95
CPU-194	**Snakman** - Pacman for the VIC	14.95
Defender on Tri - you're the pilot of an experimental ship		17.95
3-D Man - the popular arcade game, requires 3K		17.95
Exterminator - a game full of bugs		20.95

We have over 400 programs for your VIC-20 and over 250 programs for your C-64!

Shipping & Handling Charges:
First two (2) items - $2.00 per item.
Three (3) or more items - $1.00 per item.
For orders over $100 total, surface shipping will be paid by CompuSense. Blue Label or special handling will be paid by the customer.
Additional $2.00 C.O.D. fee on all C.O.D. orders.
MasterCard and Visa accepted. Give card number and expiration date on order form.
Allow three (3) weeks for personal checks.

TO ORDER:
P.O. Box 18765 **C**
Wichita, KS 67218
(316) 263-1095

AMERICAN EXPRESS® MasterCard VISA

Prices subject to change.

VIC-20® is a registered trademark of Commodore

Write for FREE Catalog!

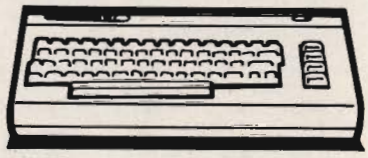
Circle No. 9

children may need their parent's help at the beginning, but should be able to take over quickly.

Now I grant you, I realize many good educational programs lack some of these qualities. The attributes of an educational program will vary from application to application. One should be aware of these attributes whether he/she is planning to develop or purchase educational software and determine which ones apply. Two characteristics that apply in all cases are quality subject matter and good organization.

The Commodore VIC 20 can give your children an added advantage in school. Its graphic capabilities and user friendliness will make learning fun and easy. Good educational software is available from many sources, and many of these software packages are reasonably priced. If you are planning to do your own programming, a good investment would be to purchase the Super Expander. This cartridge will add a new dimension to your VIC 20. Not only do you get an extra 3K of memory but also additional commands devoted to graphics and music. I plan to write a series of educational programs and articles in this publication designed for the VIC 20, some of which will require the Super Expander cartridge. If you have any suggestions on what you would like to see, please send your correspondence to me at: 2408 Snyder Ave., Bremerton, WA 98312.

I truly believe microcomputers and education go hand-in-hand, and that the Commodore 64 and VIC 20 are truly state-of-the-art in educational applications as well as many others. There is no doubt that the computer is the key to the future. A future more interesting and rewarding than we can possibly imagine.□



Commander Dealers

Given here, in zip code order, is a list of the Dealers who will be carrying the COMMANDER. We will provide updates for this list in following issues as a service to provide our readers with a local source at which they will find the COMMANDER Magazine. We sincerely appreciate the service provided by the Dealers listed below.

Our newest Dealers are on this page. Previously listed Dealers run consecutively on the following pages.

New Jersey
Bits, Bytes & Pieces
190 Buckelew Avenue
Jamesburg, NJ 08831
Manager-Owner: Erik & Margie Roll

New York
Computer Shoppe
283 Medford Avenue
Patchogue, NY 11772
(516) 758-6558
Manager-Owner: Ann Smith

Pennsylvania
York News Agency
1141 South Edgar Street
York PA 17405
Manager-Owner: Mark Lando

Georgia
Integrated Systems, Inc.
3300 Buckeye Road, #178
Atlanta, GA 30341

Florida
Computer Plus
1471 Timberlane Road, Suite 161
Tallahassee, FL 32312
(904) 893-7340
Manager-Owner: Dan Evans

Hillsboro News Co.
2102 North Sterling
Tampa, FL 33607
Manager-Owner: MaryAnn Michel

Computer Systems plus
1721 SE 47th Terrace
Cape Coral, FL 33904
Manager-Owner: Don Montgomery

Ohio
Big Bytes
1309 Boardman-Poland Road
Poland, OH 44514
(216) 758-0009
Manager-Owner: Gary Lunger

Central News Co.
2115 George Street
Sandusky, OH 44870
(419) 626-6962
Manager-Owner: Jim Justice

Computer Plus of Ohio
123 East Main Cross Street
Findlay, OH 45840
(419) 423-6700
Manager-Owner: Mike Deitelbach

Michigan
Videoasis
2237 18 Mile Road
Sterling Heights, MI 48078
(313) 979-5555
Manager-Owner: Ian Swanwick

Computers & More
1878-28th Street SW
Wyoming, MI 49509
Manager-Owner: Bill Slaughter

Illinois
Kroch's & Brentano's
29 South Wabash Avenue
Chicago, IL 60603
(312) 332-7500
Manager-Owner: Harlon Smith

Chas Levy Circulation Company
1200 North Branch Street
Chicago, IL 60622
(312) 440-4422
Manager-Owner: Harvey Wassermar

Cambridge Business Systems
3345 North Halsted
Chicago, IL 60657
(312) 525-3900
Manager-Owner: Larry Brault

Videos To Go
2517 West Reservoir
Peoria, IL 61615
(309) 682-6510
Manager-Owner: Patsy Wade

Nebraska
Computers & Components
14207 Pacific
Omaha, NE 61854
(402) 691-4238
Manager-Owner: Tim Pearson

Kansas
Kansas Micro Computer
1601 West 23rd
Lawrence, KS 66044
(913) 841-6348
Manager-Owner: John Ellena

Compu-sense
1001 South Washington
Wichita, KS 67211
(316) 263-1095
Manager-Owner: Skip Kendrick

Go Sub Int'l
501 East Pawnee, Suite 430
Wichita, KS 67211
(316) 265-9992

Texas
Rave Stores, Inc.
3308 Towerwood
Dallas, TX 75234
(214) 484-9581
Manager-Owner: Erroll Jacobson

East Texas Periodicals
7171 Grand Blvd.
Houston, TX 77225
Manager-Owner: Bill Melcher

Colorado
Denver News Company
3601 East 46th Avenue
Denver, CO 80216
(303) 321-1111
Manager-Owner: Jim Littlepage

California
Computer Barn
319 Main Street, #2
Salinas, CA 93901
(408) 757-0788
Manager-Owner: Natalie Vanderbelt

Software Emporium
4500 El Camino
Los Altos, CA 94022
(415) 941-8788

Software Depo
2652 Town Center Lane
Sunnyvale, CA 94086
(408) 730-9494
Manager-Owner: Roger Spangler

North Area News
2500 Marconi Avenue
Sacramento, CA 95821

Integral Biomedical Eng.
2947 Fulton Avenue
Sacramento, CA 95821
972-7577
Manager-Owner: Mike Kouri

Washington
Acme TV & Computers
1727 East Sprague
Spokane, WA 99202
(509) 535-1111
Manager-Owner: Jim Hanley

Computer Shop of Spokane
12218 East Sprague
Spokane, WA 99206
(509) 928-9610
Manager-Owner: Austin Hook

Dealers, continued

U.S.A.

Puerto Rico
The Micro Computer Store
1408 Avenue Jesus T. Pinero
Rio Piedras, PR 00921
(809) 781-0350
Manager-Owner: Julio C. Martinez

Massachusetts
Tycom Associates
68 Velma Avenue
Pittsfield, MA 01201
Manager-Owner: Dave Tyburski

Northshore News Co.
150 Blossom Street
Lynn, MA 01902
(617) 592-1300
Manager-Owner: Tom Mulken, Jr.

Computech Ltd.
214 Derby Street
Salem, MA 01970
(617) 741-1724
Manager-Owner: Tim Bush

Omicron Corporation
1416 Providence Highway
Norwood, MA 02062
(617) 769-6867
Manager-Owner: Steve Gavrilles

Rhode Island
International Computer Services
165 Dyerville Avenue
Johnston, RI 02919
(401) 273-1001
Manager-Owner: Steve Lablanc

New Hampshire
New England Periodical Service
6th South St.
Milford, NH 03055
Manager/Owner: Jim Nolen

Compu-Craft, Inc.
17 Dunbar Street
Keene, NH 03431
(603) 357-3901
Manager-Owner: Richard Bishop

Echo Consulting Services
PO Box 1199
Conway, NH 03818
(603) 447-5455
Manager-Owner: George Epotien

Maine
Maine Micro Systems, Inc.
55 Center Street
Auburn, ME 04210
(207) 786-0696
Manager: Nancy Lecompte

Vermont
Computeam
205 Dorset Street
South Burlington, VT 05401
(802) 862-2802
Manager-Owner: Mark Robinson

Market Place
1 Main Street
Winoski, VT 05404
Manager-Owner: Bob Howe

Connecticut
Multi-Business Computers Inc.
28 Marlborough Street
Portland, CT 06480
(203) 342-2747
Manager-Owner: Bob Stasko

New Jersey
Micro Computer Services
61 Mountain Blvd.
Warren, NJ 07060
(201) 561-3111
Manager-Owner: Jerry Prevete

Computer Workshop
1200 Haddenfield Road
Cherry Hill, NJ 07013
(609) 665-4404
Manager-Owner: Charles Kolbe

Software Land
99 Broadway
Elmwood, NJ 07407
Manager/Owner: La Lit Modi

NUBS Computer Center Inc.
6 Ames Avenue
Rutherford, NJ 07070
Manager-Owner: Robert Weigel

Software City
85 Godwin Avenue
Midland Park, NJ 07432
Manager-Owner: Arlene Destosito

Computerability, Inc.
441 Route 23
Pompton Plains, NJ 07444
(201) 835-0688
Manager-Owner: Dennis Mull

Wayne Computer Store
1459 Route 23
Wayne, NJ 07470
(201) 628-7318
Manager-Owner: Rick Delti

Software City
147 North Kinderkamack Road
Montvale, NJ 07645
(201) 391-0931
Manager-Owner: C.M. Hatfield

Software City
161 Cedar Ln
Teaneck, NJ 07666
Manager/Owner: George Barnes

Computer Workshop
1200 Haddenfield Road
Cherry Hill, NJ 08034
(609) 665-4404
Manager-Owner: Chris Kolbe

Software-n-Such
Warren Plaza, Route 130
East Windsor, NJ 08520
(609) 443-8984
Manager-Owner: Paul Hammer

BB/The Computer Store
216 Scotch Road
Trenton, NJ 08628
(609)883-2050
Manager-Owner: Barry Brown

New York
Computer Center
31 East 31st
New York, NY 10016
(212) 889-8130
Manager-Owner: Elliot Rabinowitz

Leigh's Computer
212 East 85th Street
New York, NY 10028
Manager-Owner: Leigh Goldstein

CompTek
90 John Street
New York, NY 10038
(212) 962-6131
Manager-Owner: Frances Banks

Computer Store and More
90 John St.
New York, NY 10038
Manager/Owner Francis Banks

Computer Emporium
37 North Street
Middletown, NY 10940
Manager-Owner: Kate Honders

Compu-Tech
511 Hempstead Avenue
West Hempstead, NY 11552
Manager-Owner: Jim Brewington

Software Emporium
151 Minola Avenue
Rosyln Heights, NY 11577
Manager-Owner: Sheldon Ostroy

B.C. Communications, Inc.
World Wide Electronics Dist.
207 Depot Road
Huntington Sta., NY 11746
(516) 549-8833

Computer Headquarters
1245 Middle Country Road
Selden, NY 11784
(516) 698-9373
Manager-Owner: James and Kathy Lyons

Ray's Supply
190 Route 9
Cohoes, NY 12047
Manager-Owner: Bob Howe

Upstate Computer Shop
1823 Western Avenue
Albany, NY 12203
(518) 456-3019
Manager-Owner: William J. Smith

Ray's Supply
350 Cornila Street
Plattsburgh, NY 12901
Manager-Owner: Bob Howe

Ray's Software
106 East Main Street
Malone, NY 12953
Manager-Owner: Bob Howe

Future Distribution
Trimex Bldg.—Route 11
Mooers, NY 12958
(514) 861-2831
Manager-Owner: Phillippe Faure

Ray's Software
412 East Jenesee Street
Fayetteville, NY 13066
Manager-Owner: Bob Howe

Durmac Cash Registers
1628 Erie Blvd. East
Syracuse, NY 13210
(315) 472-4531
Manager-Owner: William McCarthy

Upstate Computer Shop
99 Commercial Drive
Whitesboro, NY 13492
(315) 768-8151
Manager-Owner: Tony Violante

Personal Computers, Inc.
3251 Bailey Avenue
Buffalo, NY 14215
(716) 832-8800
Manager-Owner: Frank C. Smeirciak

Pennsylvania
One Stop Computer Shop
65 North 5th Street
Lemoyne, PA 17043
(717) 761-6754
Manager-Owner: Joanne Wright

Micro Age Computer Store
1352 Tilghman Street
Allentown, PA 18102
(215) 434-4301
Manager-Owner: Ed Eichenwald

Maryland
Professional Micro Service
100 West 22nd Street
Baltimore, MD 21218
(301) 366-0010
Manager-Owner: James A. Breen

Tri-State Computers
1504 South Salisbury Blvd.
Salisbury, MD 21801
(301) 742-2020
Manager-Owner: Tom Weiland

Virginia
Virginia Micro Systems
13646 Jeff Davis Highway
Woodbridge, VA 22191
(703) 491-6502
Manager-Owner: Suart Mitchell

Unidyne Corp.
536 Independence Blvd.
Virginia Beach, VA 23462
(804) 855-8037
Manager-Owner: Vicki Knick

CALPRO—The World of Computers
3119 Waterlick Road
Lynchburg, VA 24502
(804) 237-3825
Manager-Owner: Walter Leroy Ashley

West Virginia
Computer Associates, Inc.
113 Hale Street
Charleston, WV 25301
(304) 344-8801
Manager-Owner: Jeff Knapp

Alabama
Tricelin Corporation
Route 1, Box 128
Bankston, AL 35542
(205) 689-4999

Tennessee
American Computer Co
1004-8th Avenue South
Nashville, TN 37203
Manager-Owner: Jane Maggard

Dealers, continued

Metro Computer Ctr.
416 West Main Street
Chattanooga, TN 37402
(615) 875-6676
Manager-Owner: Wayne F. Wilson

Mississippi

Sunrise Persons Supplies
901 South John Street
Corinth, MS 38834
(601) 287-4721
Manager-Owner: Felex Gathings

Kentucky

All Business Computers
Suite C-2317 Versailles Road
Lexington, KY 40504
(606) 253-2545
Manager-Owner: Bud Walden

Stonehenge Computer, Inc.
2026-29th Street
Ashland, KY 41101
(606) 359-0545

Ohio

Office Mart, Inc.
1151 East Main Street
Lancaster, OH 43130
(614) 687-1707
Manager-Owner: Pat Blake

Computers Plus of Ohio
1346 West 4th Street
Mansfield, OH 43351
Manager-Owner: Tom Young

Computers Plus of Ohio
127 West Wiandot Avenue
Upper Sandusky, OH 43351
Manager-Owner: Roger Schoenberger

U-Compute
Parker Steele Bldg.
429 Monroe
Toledo, OH 43606
Manager-Owner: Paul Doldner

The Computer Store of Toledo, Inc
18 Hillwyck Drive
Toledo, OH 43615
(419) 535-1541
Manager-Owner: Al and Jackie Miller

Computer Connection
2851 Broadway
Lorain, OH 44055
Manager-Owner: Terry Rieger

Mentor TV Inc.
7516 Mentor Avenue
Mentor, OH 44060
Manager-Owner: William Tomkins

Computer Site
14763 Pearl Road
Strongsville, OH 44136
Manager-Owner: Bill Sero

Computer Corner Inc.
5104 Mayfield Road
Lyndhurst, OH 44124
(216) 423-5010
Manager-Owner: Ross Black

Computer Showcase
5855 Youngston-Warren Road SE
Niles, OH 44446
(216) 652-2571

Waltz Photo
438 Sixth Street
Canton, OH 44701
(216) 455-9421
Manager-Owner: Brad Zupp

Central News Co.
2115 George St
Sandusky, OH 44870
Manager/Owner: Jim Justice

Wards Computers, Inc.
868 Ohio Pike
Cincinnati, OH 45245
(513) 752-2882
Manager-Owner: Carl Ward

Computer Plus of Ohio
123 East Main Cross Street
Findlay, OH 45840
Manager-Owner: Mike Deitelbach

Indiana

Allan's Jewelry & Loan Co.
130 East 10th Street
Anderson, IN 46016
(317) 642-7978
Manager: Jerry Rubenstein

McCarels Computers
1204 Meridian Plaza
Anderson, IN 46016
(317) 643-2662

AVC Corporation
2702 Applegate
Indianapolis, IN 46203
Manager-Owner: Brent Enderle

Impair
342 Bosart
Indianapolis, IN 46201
(317) 353-9947
Manager-Owner: Fred Imhausen

A Computer Store
2140 North Mithoefor Road
Indianapolis, IN 46229
(317) 898-0331
Manager-Owner: Skip Robbins

Computer Plus
1501 Joliet Street
Dyer, IN 46311
Manager-Owner: Nancy L. Gray and Ronald Piercy

Computer People
900 Highway 212
Michigan City, IN 46360
(219) 879-8557
Manager-Owner: Harry Hopkins

General Micro Computers
52303 Emmons Road
South Bend, IN 46637
Manager-Owner: John Levy

Computer Corner
6722 East State Blvd.
Fort Wayne, IN 46815
(219) 749-8338
Manager-Owner: Tom Kutina

Custom Software
3197 South 3rd Place
Terre Haute, IN 47802
(812) 234-3242
Manager-Owner: Vicki McEntaffer

North Carolina

The Program Center
3400A West Wendover Avenue
Greensboro, NC 27407
(919) 855-8667
Manager-Owner: Rupert Fenequito

Piedmont Microsystems Inc
Route 3, Box 150 H
Frazier Professional Bldg.
Newton, NC 28658
(704) 465-3600
Manager-Owner: Lorne Michael

Bob West Computers
54 West Main Street
Brevard, NC 28712
(704)883-2595
Manager-Owner: Sylvia West

Georgia

Cardinal Computers
903 North Gleenwood
Dalton, GA 30720
(404) 226-0502

Integrated Systems, Inc.
3300 Buckeye Road NE. Suite 178
Atlanta, GA 30341
(404) 458-0713

Florida

COMPUTECH
1415 Timberlane Road
Tallahassee, FL 32312
(904) 893-1743
Manager-Owner: Dan Evans

Random Access Computers
296 Eglin Parkway
Fort Walton Beach, FL 32548
(904) 862-7763
Manager-Owner: Joanne Dodd

Florida Book Store
1614 West University Avenue
Gainesville, FL 32604
(904) 376-6066

Skippers Inc.
217 SE 1st Avenue
Ocala, FL 32671
(904) 732-3221
Manager-Owner: David Lee Skipper

Osceola Computer
1300 Dakota Avenue
St. Cloud, Fl 32769
(305) 892-1501
Manager-Owner: Raymond Barrieau

Sigma Systems of Orlando
590 North Semoran Blvd.
Orlando, FL 32807
(305) 273-2434
Manager-Owner: Tom Clance

Computer Specialties, Inc.
701 East Lincoln Avenue
PO Box 1718
Melbourne, FL 32901
(305) 725-6574
Manager-Owner: Otis P. Lutz

Software Centrum
2305 Ponce De Leon Blvd
Coral Gables, FL 33134
Manager/Owner: Chris Perez

Micro Byte, Inc
13710 SW 56 Street
Miami, FL 33175
(305) 385-2108
Manager-Owner: Ed Silverman and Lyman Conover

Focus Scientific
224 North Federal Highway
Fort Lauderdale, FL 33301
(305) 462-1010
Manager-Owner: M. Reinhardt

The Software Connection
5460 North State Road 7, Suite 108
Fort Lauderdale, FL 33319

Business Machines
2821 Pinewood Avenue
West Palm Beach, FL 33407
(305) 655-4730
Manager-Owner: Robert Frazier, Jr.

The Software and Computer Store
1506 Gulf-to-Bay
Clearwater, FL 33515
(813) 442-8803
Manager-Owner: Charles Kautz

User Clubs

Massachusetts
Masspet Commodore User Group
P.O. Box 307
East Taunton, MA 02718
Contact—David A. Rogers

New Hampshire
TBH VIC-NIC CLUB
PO Box 981
Salem, NH 03079
Contact—J. Newman
Publication—VIC-NIC NEWS
Interests—VIC-20 Exclusively

New York
National VIC Association
9 Crabapple Lane
Nanuet, NY 10954
Contact—Michael Kleinert
(914) 623-8929

North Carolina
Micro-Computer Users Club
P.O. Box 17142
Bethabara Station
Winston-Salem, N.C. 27116
Contact—Joel D. Brown
Interests—VIC-20 & CBM 64
Newsletter—The "VIC" Connection

Florida
Miami 2064
12911 SW 49th Street
Miami, FL 33175
Contact—Jim Luftman
(305) 226-1185

Kentucky
The Commodore Connection
1010 South Elm
Henderson, KY 42420
Contact—Jim Kemp
(502) 827-8153
Interests—VIC, CMB 64, PET

Ohio
Central Ohio PET User's Group
107 South Westmoor Avenue
Columbus, OH 43204
Contact—Philip H. Lynch
(614) 274-0304
Interests—Support of all Commodore Products

Commodore Youths of Ohio
9729 Lawndell
Navarre, Ohio 44662
Contact—Todd Archinal
(216) 767-3514
Interests—Commodore Users under 20

SW Ohio VIC Users Club
659 Carthage Avenue
Cincinnati, OH 45215
Contact—Tom E. Harris
761-7510

Public Domain
5025 South Rangeline Road
West Milton, OH
Contact: Bill Munch
(613) 698-5638

Indiana
The VIC Indy Club
PO Box 11543
Indianapolis, IN 46201
Contact—Linda Kropzer
(317) 878-3342

Michigan
Michigan's Commodore-64 Users Club
14342 Stephens
Warren, MI 48089
Contact—Doug Schwartz
(313) 776-5835 or
Chuck Ciesliga
(313) 773-6302
Newsletter—Sprite 64 (monthly)
Interests—All uses of Commodore 64

South Dakota
VIC-64 Users Club
203 East Sioux Avenue
Pierre, SD 57501
Contact—Larry J. Lundeen
(605) 224-4863

Illinois
The Fox Valley PET User's Club
833 Willow Street
Lake in the Hills, IL 60102
Contact—Art Dekneef
(312) 658-7321

Chicagoland C-64 Users Club
190 Oakwood Drive
Woodale, IL 60191
Contact—Russ Hurlbut
(312) 860-2015

Kansas
Commodore Users Group of Wichita
Route 1, Box 115
Viola, Kansas 67149

Texas
Commodore (Houston) Users Group
8738 Wildforest
Houston, TX 77088

California
SFVCUG (San Fernando Valley Commodore Users Group)
21208 Nashville
Chatsworth, CA 91311
Contact—Thomas Lynch (President)
(213) 889-2211 X2015 Days
(213) 709-4736 Nights
Newsletter—Monthly
Interests—All Commodore Products

Commodore Interest Association
c/o Computer Data
14660 La Paz Drive
Victorville, CA 92392

Amateurs and Artesian
PO Box 682
Cobb, CA 95426
Contact: B. Alexander KR6G

Washington
Cyborg Gazette
30023-118th Avenue SE
Auburn, Washington 98002
Contact—Ben Dunnington
(206) 939-0582
(206) 924-6992
Newsletter—Commodore 64 Magazine

Queen City Computer Club
PO Box 19597
Seattle, WA 98109
Contact—Dr. Ted Cooper
(206) 282-3271

A.T.S. VIC-20 Computer Club
7906-34th Avenue SW
Seattle, WA 98126
Contact—Ken Gazaway
(206) 935-2697
Publication—For VIC-20 only

Whidbey Island Commodore Computer Club
PO Box 1471
Oak Harbor, WA 98277
Contact—Michael Clark
(206) 675-4815
Donald Sims
(206) 675-0301
Newsletter: Chips 'N Bits

Central Washington Commodore User's Club
1222 South 1st Street
Yakima, WA 98902
Contact—Bob Wood or Tim McElroy

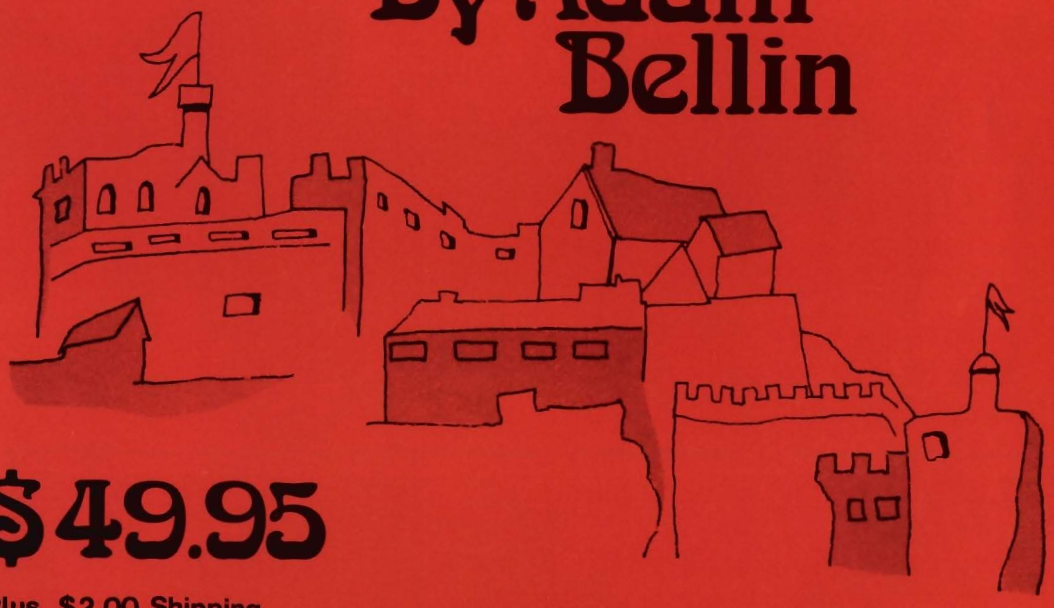
Commodore 64 Users
West 1918 Boone Avenue
Spokane, WA 99201
Contact—Terry or Sara Voss
(509) 327-7202

Canada
Winnipeg PET Users
9-300 Ennis Killeo
Winnipeg, Manitoba, Canada R2V 0H9
Contact: Larry Nevfeld

Advertising Index

Sorcerer's Apprentice

By Adam Bellin



$49.95

Plus $2.00 Shipping

TOTAL GRAPHICS FOR COMMODORE™ 64

- 16 COLORS
- HIGH RESOLUTION
- AUTO FILLS
- MEMORY MOVE
- MICROVIEW
- AUTO LINES
  - RECTANGLES
  - TRIANGLES
  - CIRCLES
  - ELLIPSES

EVENT HORIZON SOFTWARE

P.O. box 1327 New York, NY 10028 212 535-0697

Dealer inquiries invited

Circle No 16

LUNA SOFTWARE

If you own a VIC 20 or Commodore '64 get ready to explore the potential of these fine machines with LUNA Software. The software supplier for the 80's.

We have broken free of the pack with stunning games such as our sophisticated, three-screen, FINAL CONQUEST, the newest entry to our '64 line.

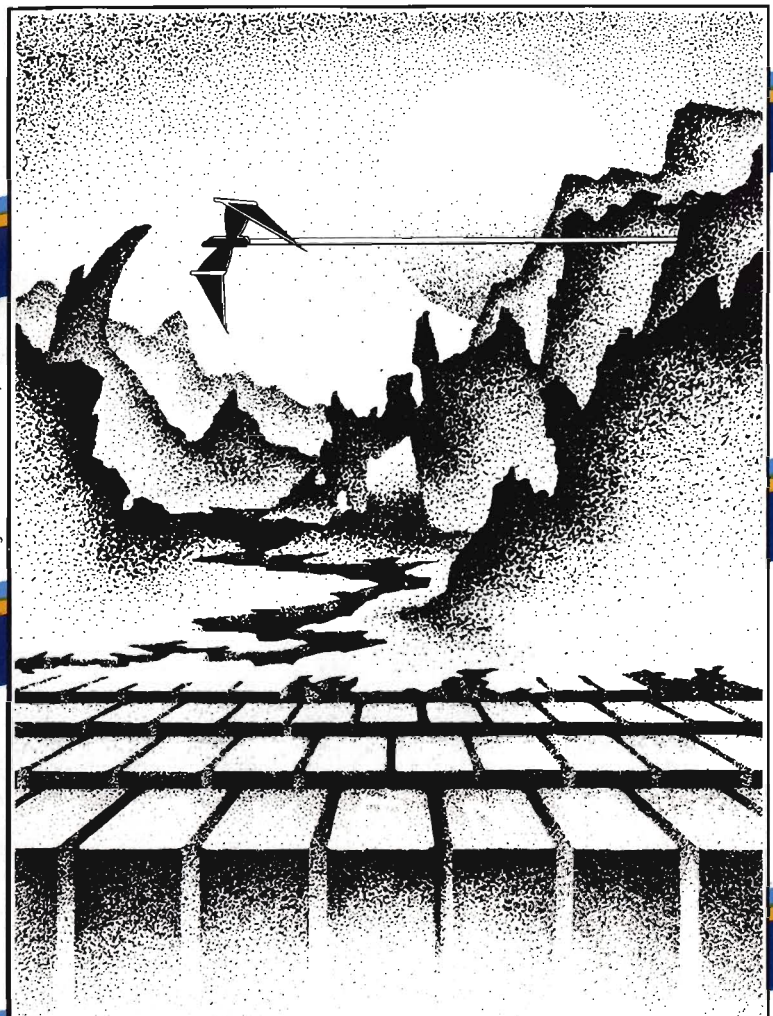
Experience heart-racing surges of adrenalin while trying to outrun lightning-footed coyotes on our HYPER-HEN grid. And, if you're the PEDESTRIAN, keep a sharp eye on the traffic behind you, and the muggers ahead.

LUNA's full line of arcade style games and user-friendly business software makes us the industry's most sought after new supplier.

LUNA Software: Committed to providing you with State-of-the-Art concepts and programming to bring out the best possible performance from your Commodore computers. Call LUNA today for the name of a dealer in your area.

LUNA SOFTWARE, P. O. Box 26922, San Jose, CA 95159-6922 (408) 378-7793

Circle No. 24



Design & Illustration: Paul Stephenson

ADVENTURE.
THE KEY IS YOUR COMPUTER

KONGO KONG

Climb ladders, avoid the barrels the crazy ape is rolling at you, and rescue the damsel. Fast machine code action.

Available for VIC-20 and COMMODORE 64
PLAYED WITH JOYSTICK OR KEYBOARD

$19.95

GRAVE ROBBERS

Introducing the first GRAPHIC ADVENTURE ever available on the VIC-20! With realistic audio-visual effects, you explore an old deserted graveyard and actually see the perils that lie beyond.

Available for VIC-20 and COMMODORE 64
PLAYED WITH KEYBOARD

$14.95

ANNIHILATOR

Defend your planet against the hostile aliens. All machine code makes this "Defender-like" program one of our best arcade games.

Available for VIC-20 and COMMODORE 64
PLAYED WITH JOYSTICK

$19.95

ADVENTURE PACK I

(3 programs)

Jack and the Beanstalk Defeat the Giant in your quest for his Golden Hen.

Computer Adventure Re-live the "excitement" of getting your computer. An adventure with a very different flavor.

Moon Base Alpha You must find a way to destroy the meteor that is racing towards your base, or else all moon colonies will be demolished!

Available for VIC-20 and COMMODORE 64
PLAYED WITH KEYBOARD

$19.95

ADVENTURE PACK II

(3 programs)

African Escape As the sole survivor of a plane crash, you must find your way out of the dark continent.

Hospital Adventure You are a spy whose mission is to complete the bungled assassination attempt on the evil dictator, who is recuperating in the hospital under heavy guard.

Bomb Threat Get back to town to warn the authorities of the bomb planted by the terrorists who left you prisoner at their hideout.

Available for VIC 20 and COMMODORE 64
PLAYED WITH KEYBOARD

$19.95

CHOMPER MAN

High speed machine action. Don't let the bullies catch you in a game packed full of machine code.

Available for COMMODORE 64
PLAYED WITH JOYSTICK OR KEYBOARD

$19.95

Programs for the **VIC-20** and the **COMMODORE 64**

All programs fit in standard memory. All programs are on cassette tape.

VICTORY SOFTWARE®

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

Check your local dealer or order directly.
ORDERING We accept personal checks, money orders, VISA, and MASTER-CHARGE. Charge orders please include number and expiration date. Overseas orders please use charge or have check payable through a U.S. bank. Add $1.50 postage and handling per order. PA residents please add 6% sales tax.

VISA

master charge

COMMANDER

☐ **1 YR. $22 (US)** ☐ **2 YR. $40 (US)** ☐ **3 YR. $58 (US)**

(PRICES DO NOT INCLUDE $4 DISCOUNT)

CHARTER SUBSCRIBERS **$4 DISCOUNT** ☐ **Check Here**

☐ Payment Enclosed Charge To:
☐ American Express ☐ Visa ☐ MasterCard ☐ Expiration Date _____

Account # _____

MasterCard Only—List 4 digits above your name _____ Signature _____

If you are using a credit card, you can enter your order by calling TOLL-FREE 1-800-426-1830 (except WA, AK, HI). (Washington residents add 7.8% tax)

Name _____
(Please type or print)

Street _____

City _____

State _____ Zip _____

Subscriptions to COMMANDER are $22 per year in the United States. Canadian and Mexican rates are $26 per year. Surface rates to other countries are $37, air mail to other countries is $54. All subscriptions begin with the current issue. *Please allow 5-6 weeks for first copy.* All monies in U.S. Funds.

OPINION POLL

Opinions of our Readers help the staff of Commander make a better magazine. The more we at Commander know about who you are and what your interests are, the better we can serve your needs. That is the reason for our new Reader's Opinion Poll. Please return this self-addressed post card to Commander.

Sex: ☐ Male ☐ Female

Age: ☐ Under 14 ☐ 15-17
☐ 18-19 ☐ 20-21 ☐ 25-29
☐ 30-34 ☐ 35-44 ☐ 45+

Zip Code: _____

Income:
☐ Under $5000
☐ $5000-$9999
☐ $15,000-$14,999
☐ $15,000-$19,999
☐ $20,000-$24,999
☐ 25,000+

What Computer do you presently own? ☐ VIC-20
☐ 64 ☐ PET ☐ PET/CBM
☐ SuperPET ☐ 128 Series

How many hours per week do you spend with your computer in the following areas:
_____ Entertainment
_____ Business
_____ Educational
_____ Programming
_____ Hobby

Please rate the following features from 1 (worst) to 10 (great).
_____ Telecommander
_____ Compukids
_____ Living with Practicalc
_____ An Introduction to Assembly Language on the VIC 20

1—8

Use only one card per person.

JULY 1983 **VOL. 1, ISSUE 8**

Name _____
(Please type or print)

Company Name (if applicable) _____

Address _____

City _____ State _____ Zip _____

1. **What is your present computer system?** A ☐ VIC-20 B ☐ Commodore 64 C ☐ PET D ☐ PET/CBM E ☐ SuperPET F ☐ 128 Series
2. **What do you expect to purchase in the next six months?** A ☐ More Memory B ☐ Printer Modem C ☐ Disk Drives D ☐ Business Software E ☐ Game Software F ☐ Educational Software ☐ Utility ☐ Word Processing
3. **What is the primary use of your computer?** A ☐ Business B ☐ Education Home C ☐ Classroom D ☐ Games E ☐ Graphics F ☐ Utility G ☐ Music H ☐ Electronics

Circle the number(s) on the card that corresponds to the numbers next to the Advertisment, New Product, or News Release for which you would like more information. The Reader Service numbers also appear next to the Advertisers' names in the Advertisers Index. Mail the card and the literature you have requested will be mailed to you, free of charge, directly from the manufacturer.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100

PLACE
STAMP
HERE

COMMANDER
PO BOX 98827
TACOMA, WASHINGTON 98498

PLACE
STAMP
HERE

COMMANDER
PO BOX 98827
TACOMA, WASHINGTON 98498

PLACE
STAMP
HERE

COMMANDER
PO BOX 98827
TACOMA, WASHINGTON 98498