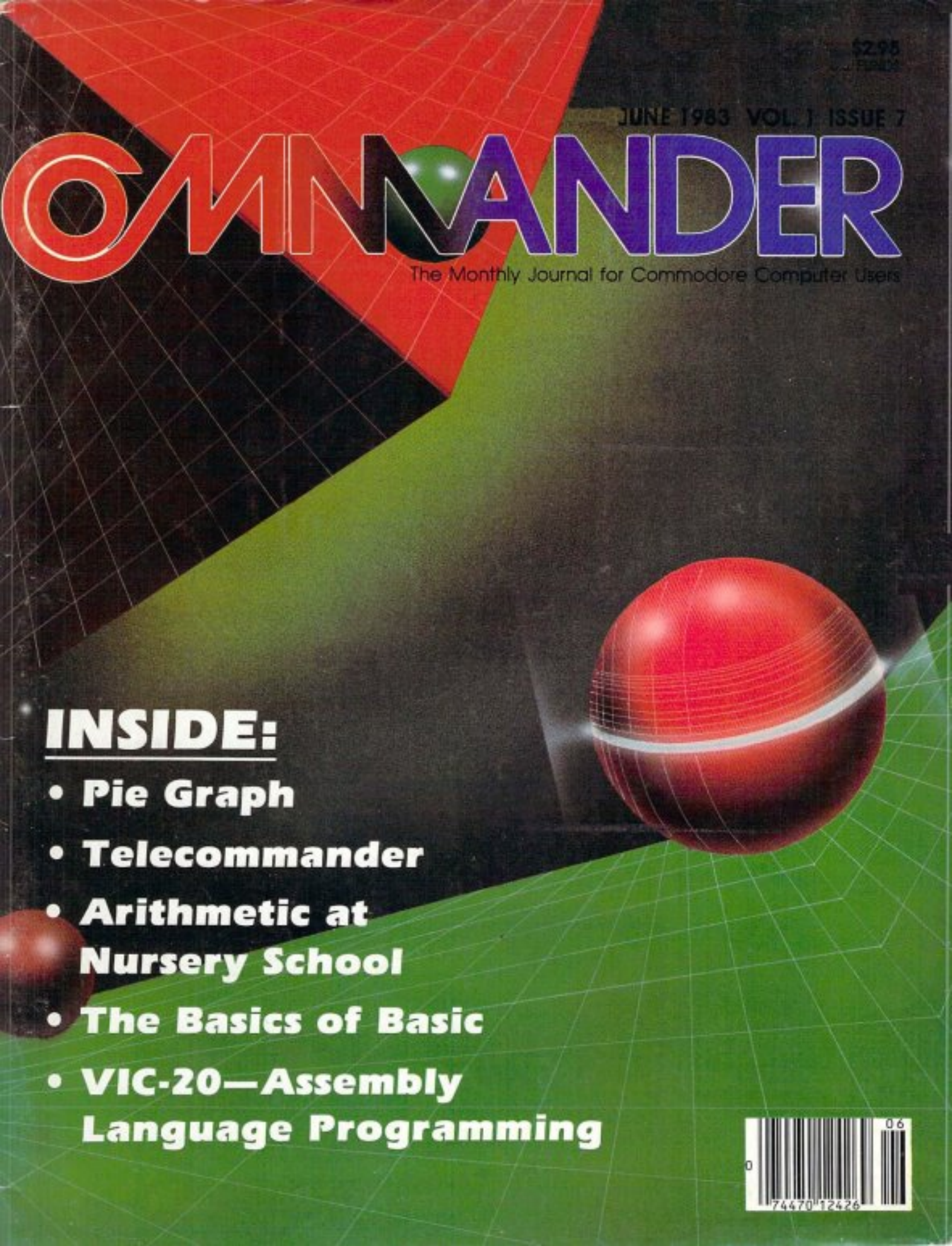
$2.95
JUNE 1983 VOL. 1 ISSUE 7

COMMANDER

The Monthly Journal for Commodore Computer Users



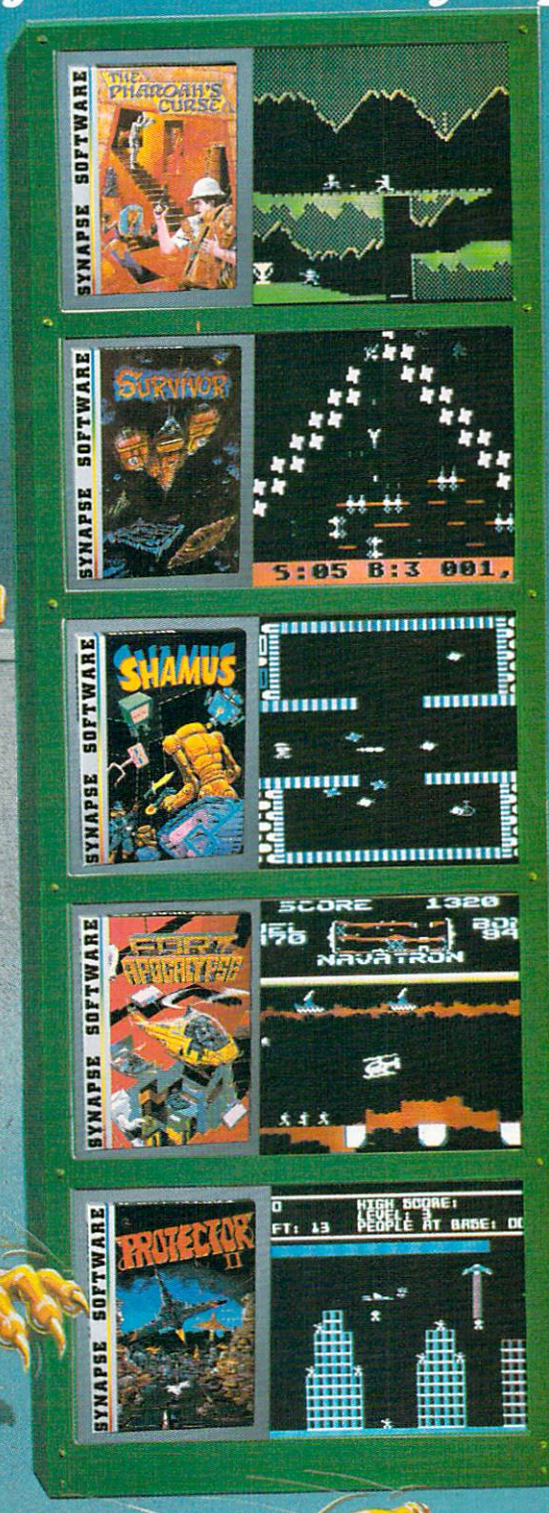
INSIDE:

- Pie Graph
- Telecommander
- Arithmetic at Nursery School
- The Basics of Basic
- VIC-20—Assembly Language Programming

0 74470 12426 06

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?

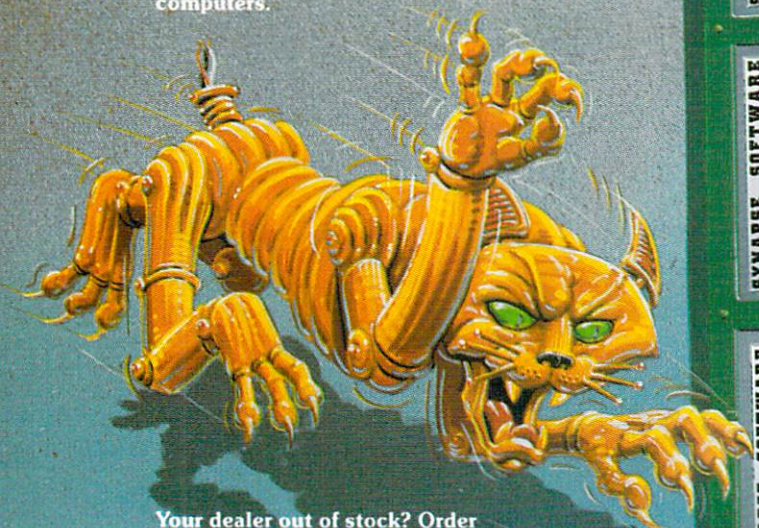
SHAMUS
The odor tells you the Shadow's there—in one of four levels of 32 rooms, each bristling with danger. You know it won't be a high school prom, but there's no turning back. Shamus—the sleuth adventure classic.

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 SOFTWARE

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

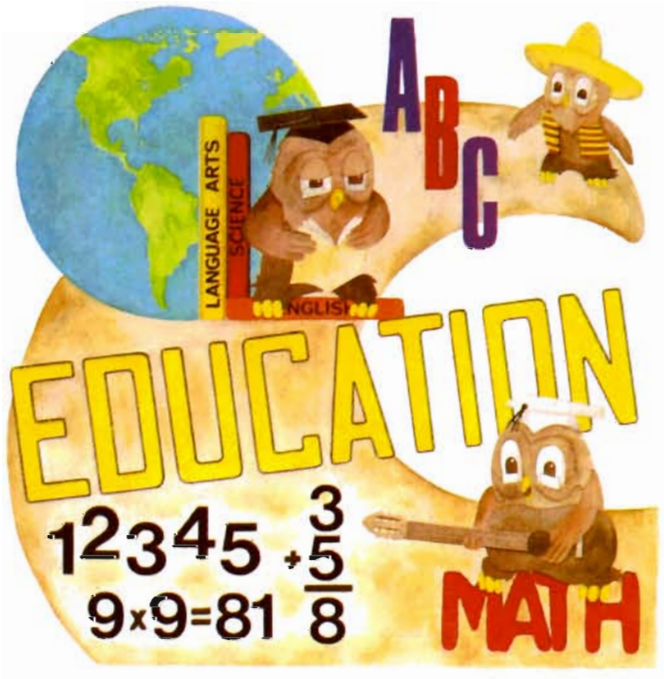
COMM*DATA SOFTWARE

GUIDES YOU AND YOUR VIC 20® DOWN ROADS OF ADVENTURE WITH:

- Maelstrom*
- Escape MCP*
- Gator Chase*
- Astro Command
- Caves of Annod
- Capture the Beast
- Whirlwind Rescue*
- Street Maze
- The Market
- Chivalry

THROUGH TRAILS OF CREATIVITY WITH:

- Sketch and Paint



ALONG THE PATH TO KNOWLEDGE WITH:

- Wordspot
- Math Tutor Series
- Alphabet Tutor
- Conversion
- Gotcha Math
- English Invaders
- Math Invaders Series

ASK FOR COMM*DATA COMPUTER HOUSE SOFTWARE AT YOUR LOCAL DEALER.

Or Send for FREE Catalog:

COMM*DATA COMPUTER HOUSE
320 Summit Avenue
Milford, Michigan 48042
(313) 685-0113

Dealer Inquiries Welcome.

Quality software also available
for Pet and Commodore 64 computers

*High Res Full Machine Code Arcade Style Games.

King of the mountain!

Workhorse solutions for tough questions.

When **Southern Solutions** acquired the exclusive marketing rights for the CMS Accounting System, the first (and the best) accounting system for the Commodore computer, we offered dealers who were dissatisfied with their current accounting software the opportunity to swap ... ours for anyone else's.

WOW! We were covered with the others ... MAS, BPI, EBS, etc ... all trading for CMS. We provide the only complete coverage of real software for Commodore computers:

THE PREMIER ... SYSTEM IV. Real accounting. More like a mini, yet priced for the Commodore. SuperMath™ gives precision to **$1 billion.** No one else comes close. General ledger, accounts receivable, accounts payable, payroll, inventory, mailing list. Plus important vertical products: oil accounting, pharmacy management, encumbrance accounting, church records and more.

THE STANDARD ... SYSTEM III. Similar to System IV but lower priced. G/L, A/R, A/P, P/R, mailing list.

Commodore 64*. Complete line of bookkeeping record keeping, personal and household management. Usually sells for under $100. Uses one or two drives, just about any printer.

Peripherals. Monitors, monitor cables, blank cassettes.

All software has FileGuard™. Never lose data files, EVEN IF YOU LOSE ELECTRICITY! Compatible with almost any computer, disk drive and printer combination. User-definable reports. Fast file access.

Sold only through professional computer dealers.

To become a **Southern Solutions** dealer, or for the name of your nearest retailer, call or write our General Manager, Bill Swingler.

Dealer Hotline: 1-800-527-4548

*Commodore 64 is a registered trademark of Commodore

Southern Solutions

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

SUPER DISK

FLOPPY DISK DRIVE FOR VIC-20 AND COMMODORE 64

GROW WITH US!

EXPANSION PRODUCTS FOR YOUR MICROCOMPUTER

Super Disk is a Commodore compatible disk drive designed to interface to the various Commodore computers such as the PET, VIC-20, and the Commodore 64. The disk drive is compatible to the model 4040, 2031, 1540, and 1541 disk drives and recognizes programs generated on any of these disk drives. The capacities are comparable to those found on the Commodore drives, and Super Disk recognizes the full instruction set of the Commodore drives. Super Disk offers 8K RAM area within the disk unit, and a serial and an IEEE bus interface. The following technical specifications apply to Super Disk.

Disk Parameters: 5.25 inch single-sided, double density diskette media; number of tracks — 35; number of sectors per track — 17 to 21 (variable); sector size — 256 bytes; number of sectors per disk — 683.

Capacities: User sectors per disk — 667; disk capacity — 174,848; data capacity — 168,656; relative data capacity — 167,132; program capacity — 168,656; directory capacity — 144 Filenames.

Electrical Interface: Power requirements — input voltage — 110/220 volts +/— 10%; frequency — 50 or 60 Hertz; power — 30 watts. Computer interface — Commodore type SERIAL bus or Commodore type IEEE bus.

System Features: 16K ROM based operating system; 8K RAM area; 6511Q Microprocessor; Power on self test diagnostics.

Win a FREE trip to Hawaii Join our "Grow With Us" Club The more you buy the better are your chances. Details at your Dealer

Ask for us by name MSD

MSD also makes Port Expanders, RAM cartridges, Games, Audio Cassette Interface, Monitor Cables and Terminal Emulation Software.

Dealer Inquiries: 1-800-527-5285

MICRO SYSTEMS DEVELOPMENT, INC.

(214) 241-3743

11105 Shady Trail • Suite 104 • Dallas, Texas 75229

Call for Clubs and Newsletters Directory

To be included in the future issues of the Commander Clubs and Newsletters Director, your club or publication must supply the following information:

1. *name of organization or publication*
2. *mailing address*
3. *contact person and telephone number*
4. *name of newsletter or publication*
5. *special interests*

Send your information to Clubs and Newsletters Directory, Commander, PO Box 98827, Tacoma, WA 98498.

Commander—The Monthly Journal for Commodore Computer Users is published monthly by Micro Systems Specialties, P.O. Box 98827, Tacoma, WA 98498. 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 Computer Users; P.O. Box 98827, Tacoma, WA 98498.**

Table of Contents

STAFF

Publisher
THOMAS L. ROSENBAUM

Editor
ALICIA A. LINDEN

Editorial Assistant
TERILYN M. AICHLMAYR

Dealer/Distributor Manager
DEBBIE A. GABBARD

Sales/Subscription Manager
ELIZABETH K. STEAN

Advertising/Article Manager
LINDA L. LINDEN

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

COVER BY:
Randy "Tarkas" Hoar

DEPARTMENTS

by Donald L. Stoner
Mercer Island, WA

Recently I was asked to explain the difference between an "information utility" and a "data base." Most readers have heard the term "data base" but may not be familiar with an "information utility." I define a data base as a very narrow collection of specific information, while an information utility is a broad collection of general information.

Virtually everyone reading this column has used a data base at one time or another. For example, when you call the telephone company information operator, you are accessing an enormous data base of telephone numbers. The phone company operators have millions of numbers right at their finger tips (their computer keyboard). The operator can extract the correct phone number in seconds but he or she cannot tell you the weather in Houston nor who won the ball game last night.

The public library fits my definition of an information utility. The little drawers, containing the 5x7 cards, are the data base for all the books contained in the information utility. If you know where to look (or the 5x7 cards can tell you), information on virtually any subject is available.

There are literally thousands of data bases available to computer owners with telecommunications capability (those having a modem and communications software). An extensive listing (but, by no means complete) is contained in a publication called "Directory of On-Line Data Bases."

There are a dozen or so services that fit my definition of an "information utility." Most are intended for businesses and are horribly expensive. Costs in the order of $100 per hour are not uncommon! There are only three that I feel are within the budget of the average person. One is called "Dow Jones News Service" and provides extensive information on stocks, bonds, commodities, etc. One might argue that this makes DJNS a data base. However, Dow Jones also provides news and other information of general interest (albeit with a business and financial slant).

The cost of the information varies, depending on which of their services is being accessed. My 13 year old, Dan, is learning to play the market (on paper) and has racked up a fair-sized bill with Dow Jones! If DJNS is of interest, check with the folks at your local Radio Shack store. They sell a Universal Sign-Up Kit, which provides one free hour on DJNS (see details below).

A second popular information utility is called The Source, and is a subsidiary of Readers Digest, Inc. The Source can provide an enormous collection of information to telecommunicators. It is so extensive, in fact, that I plan to make this information utility the subject of a separate column. It costs $100 to join The Source and approximately $5 per hour for access time. See your local Computerland store to sign up.

This month, I plan to devote the column to one of the nation's largest information utilities, CompuServe Information Service (CIS). You can purchase a subscription to CompuServe at your local Radio Shack store. It is called their Universal Sign-Up Kit (part number 26-2224) and costs only $19.95. In addition to the free hour of access time on Dow Jones (mentioned earlier), you will receive a free "get acquainted" hour on CIS.

CompuServe Information Service (CIS) is located in Columbus, Ohio and operates a large number of PDP computers made by Digital Equipment Corp. As a matter of fact, when you are connected into this utility, your VIC-20 or C-64 is actually running these powerful 32 bit "mainframe" devices . . . making the discs whir and accessing megabytes of memory.

The reader may wonder "who can afford long distance calls to Columbus, Ohio?" Fortunately, it is not necessary to rack up a lot on long distance charges to "talk" to CompuServe. They have established their own telephone network and have local telephone numbers in all the major cities of the country. These are called I/O ports in and out of their network. For example, the telephone number for their Seattle port is 634-1713. The number answers automatically and sends the answer tone (see last month's column).

As soon as I am connected to the local port, the information I send and

receive is handled (along with many other users) on high speed telephone lines that CIS leases from the telephone company. In a split second, my keystokes arrive in Columbus, Ohio and tell the CIS computer what I want it to do. I am charged approximately $5 per hour (you can spend that playing Pac-Man!) for the use of their equipment and their transcontinental telephone network. If you have ever been charged for a long distance voice telephone call, you know this is quite a bargain price.

What if your city does not have a CompuServe port (a local telephone number)? In some of the smaller cities, CIS sub-leases the lines of what are called "value added carriers" (VAC), such as TymeNet and TeleNet. The VAC equipment and setup is very similar to the proprietary network operated by CompuServe. However, access is slightly more complicated since you have to log in twice, once with the value added carrier (VAC) and once with CompuServe. How do you know if CompuServe has a local phone number or the location of the nearest VAC port? The easiest way is to check with your local Radio Shack dealer before you purchase your access package. If you have to use one of the value added carriers to access CompuServe (or most other information utilities), it will cost you slightly more per hour of on-line time. By the way, you don't have to stay on-line for a full hour. The billing equipment usually logs charges in six minute increments.

It is somewhat difficult to explain an "information utility" to someone who has not yet connected his computer to the telephone network. The preceding definition (a collection of information) seems quite inadequate.

Rather than describe it, why don't you log onto CompuServe with me and see what is available on the hundreds of menus sent to us over the telephone line?

I made the accompanying printout of a recent session on CompuServe. The first step is to dial the nearest port. You will hear the phone ring a couple of times, then answer. After a couple of seconds, you should hear a pierc-

```
+C

User ID: 71625,1620
Password:
CompuServe Information Service
8:12 PST  Saturday  19-Mar-83

WHAT'S NEW

Software Exchange Now Has 120
  Microcomputer Programs
Aunt Nettie Moves to Fridays
New Weather Features

For details, see What's New
Enter: GO NEW at the ! prompt
on any page.

Key <ENTER> for next page:

CompuServe                Page CIS-1

CompuServe Information Service

1 Home Services
2 Business & Financial
3 Personal Computing
4 Services for Professionals

5 User Information
6 Index

Enter your selection number,
or H for more information.

!1

CompuServe                Page HOM-1

HOME SERVICES

1 News/Weather/Sports
2 Reference Library
3 Communications
4 Home Shopping/Banking
5 Groups and Clubs
6 Games and Entertainment
7 Education
8 Home Management
9 Travel

CompuServe                Page FIN-1

BUSINESS AND FINANCIAL SERVICES
```

ing whistle. When you have this, unplug the telephone handset and plug in your VIC Modem. You will not see anything on your screen until you send CompuServe a Control C. This is done (for most VIC and C64 terminal software) by depressing the F1 key. The control character is echoed back to you and this is the first thing you see on your screen (and the accompanying listing). This is followed by a request for your user identification number. As you can see, from the listing, my number is 71625,1620. The next step is to enter your secret password. Note that this is not echoed back to you, in case someone is looking over your shoulder. You would not want them to know your security number any more than you would for your bank card.

At this point, you are logged into a CIS. A short preamble tells you what new information and features are available. This is followed by what is called the "top menu" (CIS-1 in the listing). You might think of the menu structure as a tree, with menu CIS-1 at the very tip-top. Each one of the selections shows branches off to other menus (which sometimes lead to other menus!)

To show you a broad cross section of the information available, I requested each of the menus for the items on the top menu. At any point, where you see the exclamation mark prompt (!), you can either type in the selection number, the letter "m" for the previous menu, or the letter "t" to set to the top menu and start all over.

After each of the menus shown (HOM-1, FIN-1, PCS-1, SFP-1, CIS-4), I entered an "m" to take me back to the top menu. However, I edited out this repetitive menu so the listing required as little space as possible in the magazine.

Note that every information page has a designator (for example, CIS-1, PCS-1, etc.). If you know exactly where you want to go, you can jump to the page directly. This is what I did to get to the Commodore SIG (special interest group). If you have been following the listing, you will note that I entered "go PCS-116" at the prompt. CompuServe provides (both on-line

```
 1 News/Reports
 2 Reference Databases
 3 Communications
 4 Brokerage Services
 5 Banking Services
 6 Discussion Forum
 7 Travel Services

 8 Personal Finance

CompuServe                Page PCS-1

PERSONAL COMPUTING SERVICES

 1 News
 2 Reference
 3 Communications
 4 Shop at Home
 5 Groups and Clubs
 6 Programmer's Area

CompuServe                Page SFP-1

Services for Professionals

   1 Agribusiness
   2 Aviation
   3 Engineering/Technical
   4 Environmental
   5 Legal
   6 Medical

CompuServe                Page CIS-4

USER INFORMATION
  1 What's New
  2 Command Summary & Usage Tips
  3 FEEDBACK, Manuals, Products
  4 Changing Terminal Defaults
  5 Changing Your Password
  6 Reviewing Your Charges
  7 Changing Credit Card Info
  8 Telephone Access Numbers
  9 Current Rates
 10 CompuServe Viewpoint
 11 Electronic Bounce Back

!go pcs-116

CompuServe                Page PCS-116
Request Recorded,
One Moment, Please
Thank You for Waiting

Welcome to Commodore Computers
```

and by mail) a listing of each page. From previous experience, I knew the Commodore bulletin board entry page was PCS-116.

There is a slight pause (One Moment Please) while CIS finds which host computer that the information is on. After a few seconds, my keyboard is connected to the Commodore data base. Their program knows who I am from my ID number, advises me the last time I was connected and what the high message number was during that connection. It also tells me that I am the 60275th person to use the SIG and what message numbers are presently being stored (all old messages "drop off" the bottom of the pile like leaves on a tree).

By entering an RM, I can retrieve any messages marked to my attention. Since there were none, I entered a RR for reverse retrieval of messages. I listed out message 19002 from a fellow named Neil to Commodore (the SYSOP, or system operator). At the end of the message, I can either Continue, Reply or go to the Top. Selecting "T" takes me back to the function prompt.

Here again, I can go directly to the entry page of another SIG. The program tells me the exit date and time and the number of the highest message I retrieved. Following this, I was transferred directly to page CEM-450 the SIG for Computers and Electronics Magazine.

Again, I was transferred to another host computer. This time, I found there was a message waiting for me. After reading the message, I made an immediate reply (RE). There is a whole "raft" of editing commands available if I make a mistake entering the message or if I decide to change the way I say something. By entering the blank line (depressing the RETURN key, without having typed any characters), the program knows I am done and gives me the option of editing, listing or saving the message. As you can see, I saved it, then logged off of CompuServe (not shown).

Once again, we are out of space, but I will be looking forward to seeing you here again next month. Until then, keep on telecommunicating.□

```
Name:       Don Stoner 71625,1620
Last on:    12-Mar-83  07:05:05
High mss#:  18055

You are user number 60275
System contains messages
18548 to  19003

Function: rm
No marked messages present

Function: rr
System contains messages
18548 to  19003
Starting message number:9002

#: 19002       Sec. 2 - Vendors
Sb: VIC/C64 COMPAT?
    19-Mar-83  07:54:03
Fm: NEIL MCANALLY 73225,263
To: COMMODORE BUS MACH.

I'M CONSIDERING REPLACING MY VIC
WITH A NEW 64. I NEED TO KNOW
WHETHER I CAN RUN VIC SOFTWARE ON
THE 64, AND WHETHER MY VIC
PERIPHRALS WILL FUNCTION ON THE
NEW UNIT. CAN SOMEONE AT THE
COMPANY STRAIGHTEN ME OUT ON THESE
QUESTIONS? THANKS FOR YOUR TIME.
 SIGNED: NEIL MCANALLY  73225,263.

(C RE T):t

Function: go cem-450

Exiting at 19-Mar-83  08:23:41
Last message on system: 19004
High message retrieved: 19004
Thank you for visiting Commodore
!
CompuServe              Page CEM-450

Request Recorded,
One Moment, Please
Thank You for Waiting

Welcome to CEMSIG, V. 1A(46)

Name:       DON STONER 71625,1620
Last on:    19-Mar-83  08:21:21
High mss#:  9357

You are user number 21533
```

```
System contains messages
8950 to  9418

You have a message waiting:

 #: 9382       Sec. 2 - TI
Sb: VIC CASSETTE
    17-Mar-83  22:22:04
Fm: EDWARD COLE 74575,1466

These messages have been marked
for retrieval with the RM command

Function: rm

 #: 9382       Sec. 2 - TI
Sb: VIC CASSETTE
    17-Mar-83  22:22:04
Fm: EDWARD COLE 74575,1466
To: DON STONER 71625,1620

PLEASE GO INTO DETAIL ON THE VIC
PROGRAM THAT UPLOADS AND DOWNLOADS
TO CASSETTE. IS THIS THE VICTERM-40?
I WOULD APPRECIATE YOUR HELP IN THIS
MATTER...BEST WISHES

(C RE T) (D=delete): re
1:
Hi Edward.. no, I think VicTerm 40 is
just a dumb terminal.
2:
However, this program acatually "hand
shakes" CompuServe
3:
while it dumps the downloaded data to
the Datasette.  It
4:
will be announced by Bytesize Micro
Technology next month
5:
for about $19.95.  Works with 3.8K
of memory.  Will keep
6:
you posted.  Regards dls

7:

Leave option: s

Message # 9419 Stored
```

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

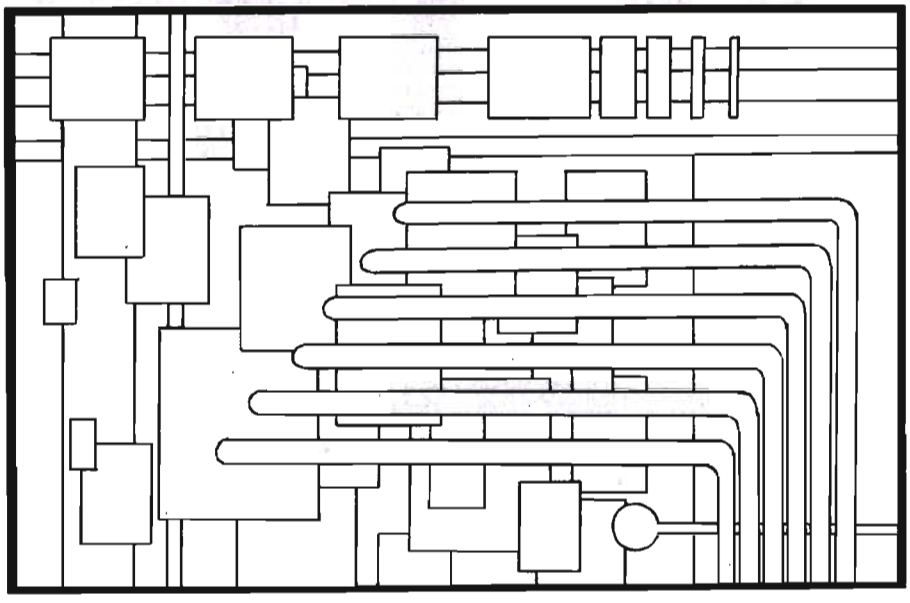
With more challenge than an arcade game, learn to type up to 75+ words/min. (User selectable, but no FOOLING AROUND allowed). TEXT IS WIDELY VARIED SINCE IT COMES FROM THE PROGRAM TAPE. Action color graphics with sound fix your eyes to the screen (away from your fingers - clever!) Your man rows your 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

Data Base Files



by Claud E. Cleeton
Bellevue, WA

A program is described for creating a file of dated entries of stock market data that is often used in market analysis. Techniques for assembling the data in strings, saving on tape files for future use, and recovery of specific items are discussed. The program also provides for adding to the data list or modifying it. The program as listed will run on any PET with 8K RAM. Techniques useful in other programs are pointed out.

There are many computer applications utilizing an extensive data base such as a time series for analysis of trends. This program is designed to generate such a data base, update it, modify or select portions of the data, and store the data in a tape file for later use. A number of programming techniques are illustrated by this program and will be specifically described as an aid to other programming applications. Programming is in BASIC and can be followed by the beginning programmer.

The example given here is taken from my stock market analysis programs where a time series of stock market data is built up consisting of date, price (P), number of stocks advancing (AD) and number of stocks declining (DE). The items may be expanded if desired. The listing given will run on a PET with only 8K RAM, but the amount of data which can be stored is limited. With more memory available, the data handling capability can be extended simply by increasing the Dimensions in line 40. With 16K memory, daily data for over a year can be stored along with additional programming for analysis.

The program is menu driven, providing the following selections (see lines 2010-2075).

```
CREATE DATA TABLE
READ DATA TABLE
SAVE DATA ON TAPE
ADD TO DATA TABLE
SELECT DATA
CORRECT DATA ERROR
PRINT DATA TABLE
```

These items are largely self explanatory. The item SELECT DATA permits one to select a block of data from the total series. It is useful for discarding old data to shorten the series, but it can also select a block within the series. The selected block can be saved in a tape file as a part of a new series. When read back, it becomes the main table. The PRINT DATA TABLE selection is given merely to demonstrate recovery of the data components. After this recovery one would normally go to some analysis program rather than just printing a list.

In creating the table you are first asked to INPUT the month as three letters (subroutine 10000), then the year as 2 digits (line 2220). This is followed by a print of the table heading and a request for inputs under each item of DAY, PRICE, ADVANCES and DECLINES. Preceding each DAY is printed a number (K) starting with one (1) which keeps track of the number of line entries and is used as a line reference for the data for each date. This subprogram is contained in lines 2200-2360. To terminate the data entry you are instructed to enter DONE which returns the program to the menu. The three components of the data are entered as strings and these together with the numerical values for P, AD, and DE, converted to strings, are added (concatenated) in line 2320 as Y$(K). The symbols # and * are field separators. All the data for a given date is then contained in Y$(K) with K providing a serial number corresponding to the date. Line 2260 adds a zero ahead of each day number less than 10 so that each day is two characters long. The complete date is thus located in the first seven characters of Y$ and is recovered in line 2330 as DT$.

Collecting the various components in a single string simplifies the handling procedures. When the individual components are of known length, as in the date, each may be recovered by a LEFT$ or MID$ function. The remaining three components are of unknown length, hence the need for the field separators. The separation technique is illustrated in lines 3300-3410. FOR-NEXT loops are used to find the field separator and the component is then found by a MID$ function. As each component is found, the next FOR-NEXT loop may start from the previous separator. If this is done, the separator symbols may be repeated. Fixed length components may also be located at the end of the string and found by RIGHT$ and MID$ calls measured from the length of the string (LN).

Another useful technique is demon-

strated in creating the data file which saves inputs of repetitive values. The program may use daily or fairly frequent time intervals. Once the starting year and month are entered only the day is called for. Since the value of the day values increase during the month, when a day number less than the previous number is encountered, line 2270 senses this and the subroutine 1000 is called asking for the new month to be entered. When this occurs, line 2270 also notes when the new month is JAN and automatically adds one to the year number. This technique materially reduces the keystrokes needed for inputting a series of dates.

When the data series is terminated by entering DONE, the last data reference number is recorded as KM and the last date as LD$ (line 1300). When the data is saved on tape, lines 3000-3120, the number of items is first entered in the file for it will be needed when the file is read, lines 2400-2530. In this program there is a choice of files to be saved, either the complete file Y$ or a selected portion Z$. The number of items saved is MX which is either the number KM in the complete table or KS, the number in the selected table. Likewise, the last date saved is DT$ corresponding to either LD$ or LS$. This last date saved is DT$ corresponding to either LD$ or LS$. This last date is important when adding to a file to denote the starting point for the new data.

The files must first be opened (lines 2430 and 3040) and should be closed when the operation is completed. The reading program must correspond exactly to the saving program, that is, the order of the items must be the same. The designation of the variables may differ, but not the character, that is, strings must correspond to strings and numbers to numbers.

The listing that follows can be keyed into most any computer using BASIC with only minor changes. The changes which you would probably need to make are the clear screen instruction which is CHR$(147) here and the cursor control shift up one CHR$(145) found in lines 2290-2310. Also, the file handling instructions must conform to the particular computer.□

```
10 REM DATA BASE FILES
20 CT$="ENTER CONT WHEN READY":ER$="ERROR"
40 DIM Y$(50),D(50),K(50),D$(50),Z$(50):GOTO 2000
60 PRINT"NO.  DATE  PRICE    ADVANCES  DECLINES":RETURN
70 PRINT"NUMBER OF DATA ENTRIES=";KM:RETURN
80 PRINT"LAST DATE =";LD$:RETURN
90 PRINT"INSERT DATA CASSETTE.":RETURN
100 D$(K)=LEFT$(Y$(K),2):D(K)=VAL(D$(K))
105 M$=MID$(Y$(K),3,3):YR$=MID$(Y$(K),6,2):RETURN
110 PRINT,"VALUE TOO LARGE":RETURN
200 IF A$=""GOTO 200
210 GOTO 2000
1000 PRINT:INPUT"MONTH <3 LETTERS>";M$:PRINT
1005 IF LEN(M$)<>3 THEN PRINT ER$:GOTO 1000
1010 RETURN
1020 PRINT TAB(10);:INPUT"YEAR <2 DIGITS>";YR$:PRINT
1025 IF LEN(YR$)<>2 PRINT ER$:GOTO 1020
1030 RETURN
1050 YR=VAL(YR$):YR=YR+1:YR$=STR$(YR):YR$=RIGHT$(YR$,2):RETURN
1100 PRINT CHR$(147):PRINT,"* CREATE DATA TABLE *":RETURN
1200 PRINT CHR$(147),"* TO CORRECT ERROR *":G=1
1210 PRINT"RETYPE DATA FOR THIS ITEM":INPUT"ITEM NUMBER";K
1220 GOSUB 100:GOTO 2240
1300 KM=K-1:D$=STR$(D(KM)):LD$=D$+M$+YR$:G=0:GOSUB 80
2000 PRINT CHR$(147),"** DATA FILE **":PRINT
2005 PRINT TAB(12);"SUBPROGRAMS":PRINT:G=0
2010 PRINT"TO SELECT SUBPROGRAM,",,"ENTER":PRINT
2020 PRINT,"CREATE DATA TABLE","1"
2030 PRINT,"READ DATA TAPE",,"2"
2040 PRINT,"SAVE DATA ON TAPE","3":PRINT
2050 PRINT,"ADD TO DATA TABLE","4"
2060 PRINT,"SELECT DATA",,"5"
2070 PRINT,"CORRECT DATA ERROR","6":PRINT
2075 PRINT,"PRINT DATA TABLE","7"
2080 INPUT I
2090 ON I GOTO 2200,2400,3000,2800,3200,1200,3300
```

```
2200 GOSUB 1100:M$="":K=1
2210 IF M$=""THEN GOSUB 1000
2220 INPUT"YEAR <2 DIGITS>";YR$
2225 IF LEN(YR$)<>2 THEN PRINT ER$:GOTO2220
2230 GOSUB 1100:PRINT"WHEN FINISHED, ENTER DONE FOR DAY"
2235 IF G=0 THEN GOSUB 80
2240 GOSUB 60:PRINT K;:INPUT"DAY";D$:IF D$="DONE"THEN 1300
2250 VL=VAL(D$):IF VL<0 OR VL>31 THEN PRINT ER$:GOTO 2240
2260 IF VL<10 THEN D$="0"+D$
2270 D(K)=VL:IF D(K)<D(K-1)THEN GOSUB 1000:IF M$="JAN"THEN
     GOSUB 1050
2290 PRINT TAB(11);CHR$(145);:INPUT P
2300 PRINT TAB(22);CHR$(145);:INPUT AD
2310 PRINT TAB(32);CHR$(145);:INPUT DE
2320 Y$(K)=D$+M$+YR$+STR$(P)+"#"+STR$(AD)+"*"+STR$(DE)
2330 DT$=LEFT$(Y$(K),7)
2340 PRINT CHR$(147):GOSUB 60:PRINT K;TAB(6);DT$;
2345 PRINT TAB(15);P;TAB(24);AD;TAB(34);DE
2350 IF G=1 GOTO 2000
2360 K=K+1:GOTO 2240
2400 PRINT CHR$(147),"* READ DATA FROM TAPE *"
2410 GOSUB 90
2420 INPUT"WHICH FILE NO.";F
2430 OPEN F,1,0
2470 INPUT#F,KM
2480 INPUT#F,LD$
2490 FOR J=1 TO KM
2500 INPUT#F,Y$(J)
2510 PRINT J;")  ";Y$(J)
2520 NEXT:CLOSE F
2530 GOTO 2000
2800 PRINT CHR$(147):K=KM:GOSUB 100:K=K+1:GOTO 2230
3000 PRINT CHR$(147),"* SAVE DATA ON TAPE *"
3010 PRINT"TO SAVE MAIN FILE, ENTER 1"
3020 PRINT"TO SAVE SELECTED FILE, ENTER 2"
3030 INPUT I:GOSUB 90
3035 POKE 243,122:POKE 244,2
3040 INPUT"FILE NO.";F:OPEN F,1,1
3050 IF I=1 THEN MX=KM:DT$=LD$:GOTO 3070
3060 MX=KS:DT$=LS$
3070 PRINT#F,MX
3075 PRINT#F,DT$
3080 FOR J=1 TO MX
3090 IF I=1 THEN PRINT#F,Y$(J):GOTO 3110
3100 PRINT#F,Z$(J)
3110 POKE 59411,53:NEXT:CLOSE F
3120 GOTO 2000
3200 PRINT CHR$(147),"* SELECT PORTION OF DATA *":GOSUB 70
3210 INPUT"STARTING NO. OF ENTRIES TO BE SAVED";S
3220 IF S>KM THEN GOSUB 110:GOTO 3210
3230 INPUT"LAST NO. OF ENTRIES TO BE SAVED";L
3240 IF L>KM THEN GOSUB 110:GOTO 3230
```

```
3250 FOR I=1 TO L-S+1
3260 Z$(I)=Y$(S+I-1):
 NEXT
3270 KS=I-1:LS$=LEFT$
 (Z$(I-1),7):GOTO 2000
3300 GOSUB 70:GOSUB
 80:GOSUB 60
3310 FOR J=1 TO KM
3320 DT$=LEFT$(Y$(J),
 7):LN=LEN(Y$(J))
3330 FOR K1=8 TO LN:
 IF MID$(Y$(J),K1,1)
 ="#"THEN 3350
3340 NEXT
3350 P=VAL(MID$(Y$(J)
 ,9,K1-9))
3360 FOR K2=K1 TO LN:
 IF MID$(Y$(J),K2,1)=
 "*"THEN 3380
3370 NEXT
3380 AD=VAL(MID$(Y$
 (J),K1+2,K2-K1-2))
3390 DE=VAL(MID$(Y$
 (J),K2+2,LN-K2-1))
3400 PRINT J;TAB(3);
 DT$;TAB(12);P;TAB
 (21);AD;TAB(31);DE
3410 NEXT J
3500 PRINT:PRINT
 "ENTER CONT":STOP:
 GOTO2000
```

An Introduction to Assembly Language Programming on the VIC-20

Part VI—More Branches

by Eric Giguere
Alberta, Canada

Last month my column concentrated on branches and comparisons in assembly language, as well as a bit on loops. Unfortunately, I ran out of space before I could explain all the remaining branch instructions and get into more detail on loops. So this month I won't get into another topic, but simply continue where I left off last month.

BCC and BCS

These two instructions cause the computer to branch if the carry bit in the status register is clear (Ø) or set (1), respectively. The carry bit is used mainly with the math instructions, and since I'm going to discuss those next month, I'll leave these for then.

BMI and BPL

These two instructions cause a branch if the N flag in the status register has the proper setting. The N flag determines if a byte is positive ($ØØ-7F) or negative ($8Ø-$FF), and relates directly to bit 7 of a byte. For example, loading the accumulator with $FE would set the N flag, because bit 7 is on:

$FE = % 1 1 1 1 1 1 1 Ø

Loading a register with a positive number (one without bit 7 set) would clear the N flag. BMI is the branch that works if the N flag is set, while BPL works if clear. It's easy to remember how these two work because BMI and BPL stand for Branch on MInus and Branch on PLus, respectively. Thus you know that a number with bit 7 set will branch on BMI because this means that it is negative. BPL works in a similar fashion, except it operates on positive numbers. You can use these instructions in a program to check if a value is within a certain range, as such:

```
    LDA $121C
    BMI ERROR
    •
    •
```

In this example the computer will branch to the routine labelled ERROR (not shown here) **only** if the value in $121C is greater than $7F, and is thus negative. If the BMI was changed to BPL, the branch would occur on a value less than $8Ø.

BVC and BVS

These two instructions are used to detect an **overflow** via the V flag in the status register. An overflow occurs when two binary numbers of the same sign are added and the sum has the opposite sign. This is not necessarily an error condition, but sometimes when you add signed numbers you don't want this (it doesn't make any difference if you don't care about the signs of the numbers). Also, an overflow will only happen if the two values together exceed the allowable range for signed numbers (−128 to +127). An example is adding −128 to −128:

```
sign
          1 Ø Ø Ø Ø Ø Ø Ø (−128)
plus      1 Ø Ø Ø Ø Ø Ø Ø (−128)
          ---------------
          1 Ø Ø Ø Ø Ø Ø Ø Ø
```

extra bit ignored, so value becomes %ØØØØØØØØ.

As you can see, the resultant from this addition is %ØØØØØØØØ, and causes the overflow flag to be set, because we added two negative numbers and the result is considered positive. Again, more on this in some future column.

Using Branches

Last month I briefly discussed using branches to create loops. These loops can be used for practical purposes within a program, as such:

```
for VIC/C64
  LOOP LDA $C5
       CMP #$4Ø
       BEQ LOOP
for PET/CBM
  LOOP LDA $97
       CMP #$FF
       BEQ LOOP
```

Location $C5 ($97 in PET/CBM) is a location in memory that holds the current value of the key currently being pressed. If no key is being pressed, it holds a value of $4Ø ($FF in PET/CBM). What the program does, then, is load the accumulator with the value in $C5 (or $97) and then use a CMP to check if a key is being pressed or not. If a key is being held, the computer will leave the loop and continue with the code after the example. Otherwise it will simply continue loading the accumulator and comparing until you do press a key. You can do the same thing from BASIC using the following:

```
1Ø A = PEEK(197):IFA = 64THEN1Ø:
  REM FOR VIC/C64
1Ø A = PEEK(151):IFA =
  255THEN1Ø:REM for PET/CBM
```

I could have coded it as 1Ø IFPEEK(197) = 64THEN1Ø (for VIC/C64), but I just wanted to make it resemble the machine language code as much as possible, so I "loaded" the variable A with the value and then did a comparison and branch. You might also find it easier to think of the registers as "variables." The accumulator would become A, the X- and Y-registers X and Y, and the status register S, subdivided into C,N,V and Z. If this helps you to understand what I'm talking about, then by all means use it.

Getting back to our little program, we can alter it very easily to wait for

you to press a certain key and ignore all others. The new program is as follows:

```
        VIC/C64
LOOP LDA $C5
        CMP #$xx
        BNE LOOP
        PET/CBM
LOOP LDA $97
        CMP #$xx
        BNE LOOP
```

All we did was change the CMP and the BEQ. Now the computer will wait until you press the key that has the value of xx (you put in the value you want). Anything else will be ignored. To find out what value to use, RUN the following short routine:

```
10 PRINTPEEK(197):GOTO10:
   REM FOR VIC/C64
10 PRINTPEEK(151):GOTO10:
   REM FOR PET/CBM
```

Your screen should fill with numbers. The program prints out the current value of the key being pressed by PEEKing location 197 (151 for PET/CBM). As long as you don't press a key, the value from that location (which I've named KEYMAT) should be 64 for the VIC/C64, or 255 for PET/CBM. Now press a key; the value changes. Find the key you wish to use, press it, and note its value. Then replace the variable xx in the machine language program with that value. For example, on the VIC/C64 the space bar's value if pressed is 32. If we place this in the program instead of xx, the CMP instruction will read CMP #$20 ($20 = 32). Now the program will loop until you press the spacebar. Wasn't that neat? (You can also use this in BASIC, instead of GET. Just use the statement X = PEEK(197) and a few IF..THEN statements to check for a proper value. The nice thing about it is that it repeats and ignores the shift key, so you don't have to worry about these.)

The monitor listing for this program is as follows:

```
VIC/C64
.:033c a5 c5 c9 xx d0
.:0341 fa 00 00 00 00
PET/CBM
.:033c a5 97 c9 xx d0 fa 00 00
```

(This program starts at $033c - 828 dec. - and will work on all machines, since it starts in the cassette buffer. Just remember to replace xx with the value you want to use.)

Printing a Message

Remember two columns back when I gave you monitor listings that placed a message onto the screen? It was a long listing and really amounted to nothing, since all it was doing was POKEing each letter in the message to the screen, as if you had said in BASIC: POKE 7680, 34: POKE 7681, 45 . . . etc. It's very inefficient but since we only knew how to load and store registers, it was all we could really do at the time (how would you have felt if I had thrown in a few branches without explaining what they were? . . .) But now we **have** learned about branches, so we can make a routine to print messages using these wonderful statements. Program 1 is that routine. It starts at $033C and outputs characters to the screen one at a time until it encounters a zero byte. You might find it useful. Following is a detailed explanation of how it works.

The message to be printed consists of the word "HELLO" and a carriage return (ASCII 13), which are placed into memory along with a zero byte, which indicates the end of the message. Lines 18 and 19 of the disassembly do this, at the same time giving the start of the message the label MESSGE (as if we were assigning a string variable in BASIC.) The X-register is used as a pointer to the next character to be loaded. This is achieved with the instruction LDA MESSGE,X, which will load the accumulator from the address pointed to by MESSGE **plus** the value of the X-register. A BEQ then tests to see if the value loaded was zero (we don't need to do a CMP #$00—remember?), and if so, exits the routine by branching to the RTS statement (RTS is like saying RETURN in BASIC—it ReTurnS you from a subroutine). Otherwise, a JSR (jump to subroutine, like GOSUB in BASIC) is made to a routine in ROM at $FFD2 which is present in all Commodore computers and prints the ASCII value of the accumulator to the screen. So if the accumulator has a value of $30, it will print a zero (the number zero) to the screen, since ASCII $30 (48) is the code for a zero. The program will then branch back via a BNE (the accumulator doesn't hold a zero ($00)) to increase the X-register and print the next character if not a zero. If you're not quite certain how it works, perhaps this BASIC equivalent will clarify it for you. It follows the machine language routine line for line except for the defining of the message, which is done in line 1 instead of lines 18 and 19:

```
 1 ME$ = "HELLO" + CHR$(13) +
   CHR$(0)
 9 X = 0
10 X = X + 1
11 A$ = MID$(ME$,X,1)
12 IFASC(A$) = 0THEN15
13 PRINT A$
14 GOTO 10
15 RETURN: REM if called as a
   subroutine.
```

PRINT ME$ would be faster and more efficient, but I just wanted to show you the logic behind the program.

A Scrolling Routine

My final program for this month is a little routine to scroll the screen one position to the left. It will only work on the VIC, but you can learn a lot just be examining it and trying to figure out how it works (just remember the VIC has a 22-column screen). If you VIC owners want to use it, simply type in the BASIC loader program given. SYS7168 will scroll the screen one position to the left. A line like

```
20 FORI = 1TO22:SYS7168:NEXT
```

will scroll the whole screen out of sight. The program doesn't need to be left at 7168 either. You can put it anywhere in free memory without changing it by replacing the POKE7168 + I,A in line 10 with POKExxxx + I,A where XXXX = the location you want it in. Just remember to SYS to this new location to activate it, otherwise the computer could freeze up.

Before explaining a bit of how the program works, I have to make a comment about it: it isn't very efficient. I have a better routine to scroll the screen, but it uses addition instruc-

tions, and since we haven't covered these yet, I felt it best to write a program to exclude these. So although the routine isn't the best there is, it does make good use of the concepts we've learned so far.

Comments

Basically, what SCROLL LEFT does is move each character on a line on position left and place a blank in the last (22nd) position. It does this for every line until it reaches the end of screen memory, at which time it returns to BASIC. It uses the indirect indexed addressing mode to move screen memory around. The Y-register is the pointer to the next byte to be moved, and locations $0 and $1 hold the indirect address $1E00 or $1F00, the two pages of screen memory. It has to make several checks to determine when it reaches the end of a line, and when to leave the program. An interesting piece of code is found in lines 35 and 36:

```
CLC
BCC LOOP
```

The instruction CLC clears (sets to 0) the carry flag. Because the carry is now clear, the program is forced to branch back to LOOP. It's like saying in BASIC C=0:IFC=0THENGOTO 100, or something similar. I just thought I'd mention this to you.

Next Month

Next month we're going to explore the intricate world of mathematical functions in assembly language, so be sure to pick up the next issue of Commander.□

PROGRAM 1

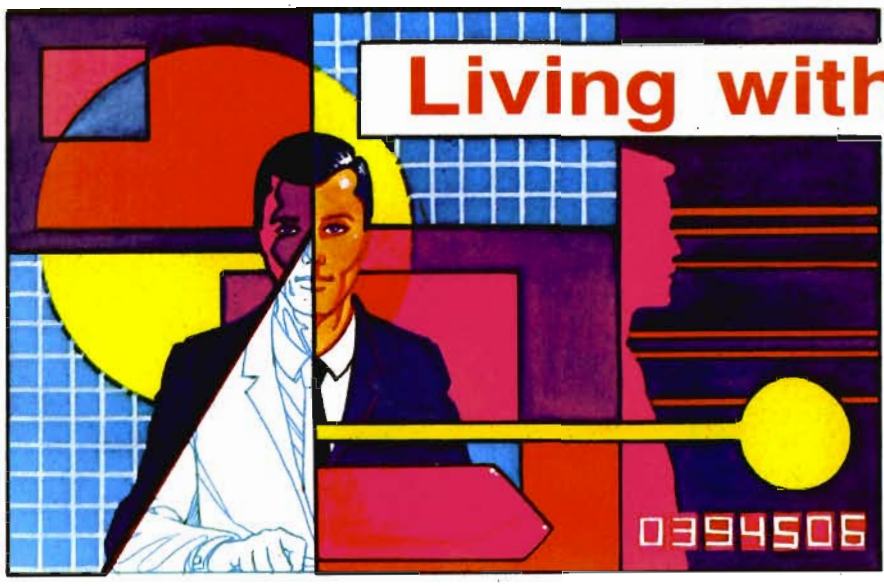
```
1 ********************
2 * PRINT MESSAGE *
3 ********************
4 ;
5 ;
6 ;
7           ORG $033C
  ;CASSETTE BUFFER
  (828 DEC.)
8 ;
9           LDX #$FF
  ;X=POINTER
10 LOOP      INX
11      LDA MESSGE,X
  ;GET NEXT CHARACTER
12          BEQ EXIT
  ;IF 0 THEN EXIT
13          JSR $FFD2
  ;PRINT CHARACTER
14          BNE LOOP
  ;NEVER ZERO,
  SO GO BACK TO LOOP
15 EXIT     RTS
16 ;
17 ;
18 MESSGE ASC .HELLO.
19          BYT 13,0
   ;CARRIAGE RETURN
   AND A ZERO
```

PROGRAM #2 - SCROLL LEFT

```
10 FORI=0TO57:READA:POKE7168+I,A:NEXT
100 DATA169,0,133,0,169,30,133,1,160,
    1,162,0,177,0,136,145,0,200,200,192,0
110 DATA208,9,230,1,165,1,201,32,208,16,96,
    232,224,21,208,231,169,32,136,145
120 DATA0,200,200,24,144,219,173,0,31,141,
    255,30,200,232,232,208,210
READY.
```

DISASSEMBLY OF PROGRAM #2

```
  1 ********************
  2 * SCROLL LEFT *
  3 ********************
  4 ;
  5 ;
  6 ;
  7           ORG $1C00   ;STARTS AT 7168 DEC.
  8 ;
  9 START     LDA #$00    ;PUT SCREEN ADDRESS
 10           STA $00     ;$1E00 (7680)
                          IN LOCATIONS
 11           LDA #$1E    ;$0 AND $1
                          (IN REV. ORDER)
 12           STA $01
 13           LDY #$01    ;INITIALIZE POINTERS
 14 LOOP      LDX #$00
 15 GETCHR    LDA ($00),Y;GET FIRST CHARACTER
 16           DEY
 17           STA ($00)   ;STORE IT ONE
                           COLUMN BACK
 18           INY
 19           INY
 20           CPY #$00        ;MOVE POINTER TO NEXT
 21           BNE NEXTCO      ;CHARACTER
 22           INC $01
 23           LDA $01
 24           CMP #$20        ;INCREMENT BASE ADDRESS
 25           BNE PAGE        ;TO NOW POINT TO $1F00
 26           RTS             ;IS IT NOW $2000?
 27 NEXTCO    INX             ;NO, GOTO PAGE
 28           CPX #21         ;RETURN TO BASIC
 29           BNE GETCHR
 30           LDA #$20        ;22 COLUMNS?
 31           DEY
 32           STA ($00),Y
 33           INY
 34           INY             ;PUT A BLANK IN THE LAST COLUMN
 35           CLC
 36           BCC LOOP        ;MOVE POINTER TO NEXT LINE
 37 PAGE      LDA $1F00       ;CLEAR CARRY TO FORCE A RETURN
 38           STA $1EFF       ;WITH BCC
 39           INY             ;GET CHAR
 40           INX             ;AND STORE IT ONE POSITION BACK
 41           INX
 42           BNE GETCHR      ;INCREMENT POINTERS APPROPRIATELY
                              ;WILL ALWAYS BRANCH TO GETCHR
```



Living with TOTL. LABEL 2.6

by Colin F. Thompson
Santa Monica, CA

I traded in my Apple//+ on a VIC-20. No, this isn't a science fiction tale. The story begins nearly two years ago when the recession was in full bloom. Economic times were tough; I was looking for ways to cut costs in our department. My primary target for cost cutting was a 30-page cross-reference list, used by thousands of our dealers and salesmen across the country. I relied on the list in my daily work, and knew it was never up-to-date and not too accurate. Once a year the list was updated, typed by a secretary, sent to the printers and typeset. 15,000 copies were printed and distributed. Each update required hundreds of man-hours of research, and the typesetting charges were astronomical. I could find no reason it needed to be typeset. The worst part of this operation was that the once-a-year update was too infrequent. The list was outdated soon after it was printed. This looked like a job for a computer.

Apple Fever

After doing a month's research on microcomputers and software, I submitted a proposal. I could do the entire job on an Apple//+, print the list on a dot matrix printer and deliver camera-ready copy to the printers. The data would remain on disk and be easily updated. I chose Stoneware's DB Master as the software to do the job. At $250, the Relational Data Base Management System was not cheap or easy to learn. I had no programming skills, but I was willing to learn. A year passed and my proposal was accepted on a trial basis. We would lease the hardware for 6 months for $1800.

I spent two months learning DB Master, and setting it up. The first computer generated list was well received. It was up-to-date and legible. The list was printed two times in the next four months. Each list was more accurate than the last. I was having a blast with the Apple, learning BASIC and, yes, playing some games. At last the lease expired, and I was informed the $6,000 purchase price for the Apple was not in the budget, leaving me without a computer. I began withdrawal symptoms immediately.

Capt. Kirk— Call the Computer Room

To my horror, I found I was hooked on computing. Days later, as I listlessly watched TV, Captain James T. Kirk introduced me to the "Wonder Computer of the Eighties." The computer junkie in me forced my body off the couch and out to the nearest VIC store. I gazed longingly at the full-sized keyboard, asked questions and read the owner's manual. The 22 giant characters on each line seemed strange, but that didn't prevent me from buying my VIC. If it's good enough for Jim Kirk, it's good enough for me.

In the next two weeks, I emerged from my apartment only once. I bought a disk drive and returned quickly. My two week visit inside the VIC revealed one astounding fact: the VIC is the technological superior of the Apple. If I could find the right software for the job, the list could be done on my little "Game Machine." I submitted a proposal to the company. All I needed was a good $500 printer. To my surprise the proposal was promptly accepted. I think others were suffering without an up-to-date list also. I bought an NEC 8023A printer (with the Company's $500) and began the search for software.

Enter the Chicken

Most software companies I called thought I was deranged for attempting to use the VIC in the office. I wasted nearly two months looking for a database manager for the VIC. My boss, 3000 miles away, started calling me daily, asking when the list would be ready. He expected to see results sometime during his lifetime. It was clearly time to re-evaluate my software needs. After carefully laying out the job requirements on paper, I got a pleasant surprise: a database manager was not necessary. I needed a list manager. List managers are a cross between mailing label programs and database managers.

The list consisted of 1100 records (printed lines), and seven fields per record. Each printed page had a title, date, page number, and column headings. This kind of job was child's play for DB Master, but comparable software for the VIC did not exist. My previous software search had turned up a mailing label program that met some of the requirements. It would print any number of lines, alphabetically sorted, and each line could have any number of fields.

That would be a good start. All I had to add was the title, page numbers, etc. For a novice programmer, this was a monumental project. My call to TOTL Software to order the cassette based TOTL.LABEL 1.0 was answered by a pleasant woman who turned out to be the author, Ann Palmer-McCarty. I told her of my plans and she thought it could be done by a skilled programmer. I ordered it anyway.

Duck Soup

To get the time I needed to modify the program, I tried a bit of misdirection. I called the secretaries of each of the company officers and offered to provide them with all the mailing labels they could ever use. They responded with a total of about 1000 different names. Before I began keying in all these labels, my luck took a turn for the better. The disk version, TOTL.LABEL 2.0 arrrived. That speeded up the process considerably. Within two weeks, secretaries all over the country were getting their labels. At last the NEC printer was earning its keep. As a side benefit, I was by then, completely familiar with the operation of the program. The diversion seemed to work. My boss stopped calling when his secretary got her labels.

Breaking and Entering

I'm never completely satisfied with any program I buy. I like to go into a program and tinker. By making changes to commercial programs, I improve my programming skills and end up with a more useful program. TOTL.LABEL 2.0 (TL) soon became my favorite target for making changes. My first foray into TL changed the screen and character colors and the printer's logical file number. (My NEC uses LFN 130 instead of the usual 4). My next change challenged my modest programming skills. I wanted to print the same label more than once. TL didn't have this option, so I broke in and made some changes.

The secretaries had asked for about 100 labels of each name. TL printed each name in the list once, requiring me to print each list 100 times. That's not very efficient. To solve the problem I wrapped a for-next loop around the print statement, changed the print menu and much to my amazement, it worked. After this change was made I could ask for 100 of each label to be printed before TL went on to the next label in the list.

I had convinced the company to buy the NEC printer by offering to use the VIC to print a large list. The list had 1100 lines of data, alphabetically sorted. Each line was 130 columns wide and had seven fields. I began the job by defining a label with eight lines: the first for alphabetizing, and seven for data. My first programming task was to make each line of the label print AFTER the last, not UNDER the last. That was easily done by adding a semicolon after the print statement. The next job was to separate each field so they lined up under a column title. The NEC has a Tab function which was difficult to understand, but I finally made it space the fields properly. The page title and column headings proved to be troublesome, but eventually everything lined up correctly. The last hurdle was the tallest.

I wanted to print 57 lines of data on each page, then skip three lines, print the column titles and print another 57 lines. I wrote a complex subroutine to count lines, interrupt the print routine, check status, change printer codes, print the title and page number and turn on the coffee pot. It took me three months to make the changes and another month to enter the data.

Success

The VIC generated list turned out to be more useful than the Apple version because it had an extra 50 columns on each line. The added space allowed more data and comments. Now that I had a program that printed labels sideways, I found lots of uses for it. For example, I changed our customer file into a telephone book. I soon had more than 20 versions of the program printing different lists. My boss was elated; he wasn't too sure that I could replace an Apple with a VIC.

At long last the project was complete. I benefited from this job in an unexpected way. At the start I was a novice programmer, and ended up with a lot of confidence and now rate myself an intermediate level programmer. It was worth the effort.

Using TOTL.LABEL

Even though TL can store its files on disk, it is not a random access program. It uses sequential files on both tape and disk. To change or add a label you must load the entire file into memory. This is a time consuming event. Changes can be made easily after the file is loaded, but if there are more than 40 or 50 labels in a file, the loading time is fairly long. Most other operations such as printing or saving are rapidly done. My 1100 line list, for example, takes over six hours to load and print. For several months, Ann kept promising me a faster version. Something with some machine language routines to speed up the Load and Browse functions. A few weeks ago, I was surprised and pleased with I opened my mail and found a diskette marked TOTL.LABEL 2.1.

Chickspeed

TL2.1 has two sections. The M/L loader, Chickspeed, is loaded first. It stores some M/L routines at the top of memory, resets all pointers and automatically loads the BASIC TL2.1. The entire process takes about one minute. Once Chickspeed is in place, it doesn't have to be loaded again, unless you turn off power to the computer. I did some comparison tests matching 2.0 against 2.1.

The results were better than I expected. For the test I used a label file with 100 labels in it. From disk TL2.0

loaded it in six minutes, 18 seconds. TL2.1 loaded it in 29 seconds. Very impressive. Printing time was never a problem with 2.0, but 2.1 is about 25% faster. Both versions have a feature called Browse. It allows you to see the first line of each label on the screen. A number next to each line is used to Select a label to be printed, or Display the full label. The Browse had been painfully slow before. TL2.1 sped it up so that the lines fairly fly onto the screen.

VIC Specs

TOTL.LABEL 2.1 is quite easy to learn. The manual is complete and printed with a good quality printer. I used the manual for only a few days and then never looked at it again. The memory requirements are modest, considering the power of the program. TL2.0 needs 8K and the Chickspeed 2.1 needs 16K expansion. Labels may be printed 1 or 2 wide. If you are a registered owner of TL2.0, you may upgrade to 2.1 for a small charge. Write to TOTL for the details. TOTL offers a 30-day warranty against defective tapes or diskettes.

C-64

The C-64 owners who have skimmed this article looking for information on their 40 column wonders should begin reading now. TOTL.LABEL is available for the C-64 also. As you might expect, the 64 version, TL2.6, has more features than the VIC.

The C-64 version allows the RS-232 user to configure the channel from a menu. It also prints multiple copies of labels next to each other. Labels can be printed up to three wide and there is a report format that prints the data for each label on one or two lines across the page (sideways). This would be useful in making lists from your data. Both have the machine language loader, will print to any printer including an RS-232, and will do the Commodore to ASCII translation. 40/80 column boards can now be used and the color variables are easily changed to colors that look good on your TV or monitor.

Its low cost and high versatility make TOTL.LABEL 2.6 an excellent value, whether you use it at home or in the office.

What Else?

Mailing labels can be stuck on things other than envelopes. If your printer can expand, condense, double-strike or print graphics, you can use those features on a label. The followng is a list of some of the uses I make of labels: Auto maintenance tags for oil changes, etc.; Telephone number tags for the phone; Office extension or comm line numbers; File tabs; Return address labels; Property tags for equipment; Program titles for cassettes and diskettes; Photograph ID's for the back of a picture or a photo album page; Name tags for luggage.

If you have a use for mailing labels I haven't mentioned, write and let me know about it. Send examples. I'll assemble a list of the most creative uses and print it here in a couple of months. Let's hear from you. While you are in a letter writing mood, let me know if there is a piece of Business or Home Management software you would like to see reviewed. This is really YOUR column. I know how difficult it is to buy software without first trying it. I'll answer all letters. Please include a SASE to: Colin F. Thompson, BASF Systems Corp., 1307 Colorado Avenue, Santa Monica, CA 90404.

Unfinished Business

Last month's review of Quick Brown Fox raised a point about the View function. When Viewing your document on a 22 column VIC, QBF scrolls the first few lines off the top of the screen. Although it's a very minor complaint, I asked the manufacturer to look into this lapse in an otherwise flawless program. QBF's superb tech support staff is still working on my complaint.

What Next?

Next month's review will focus on a "spreadsheet" program for the VIC-20. Of the many spreadsheets on the market, I chose Practicalc, from Computer Software Associates, to make my life a little easier.□

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

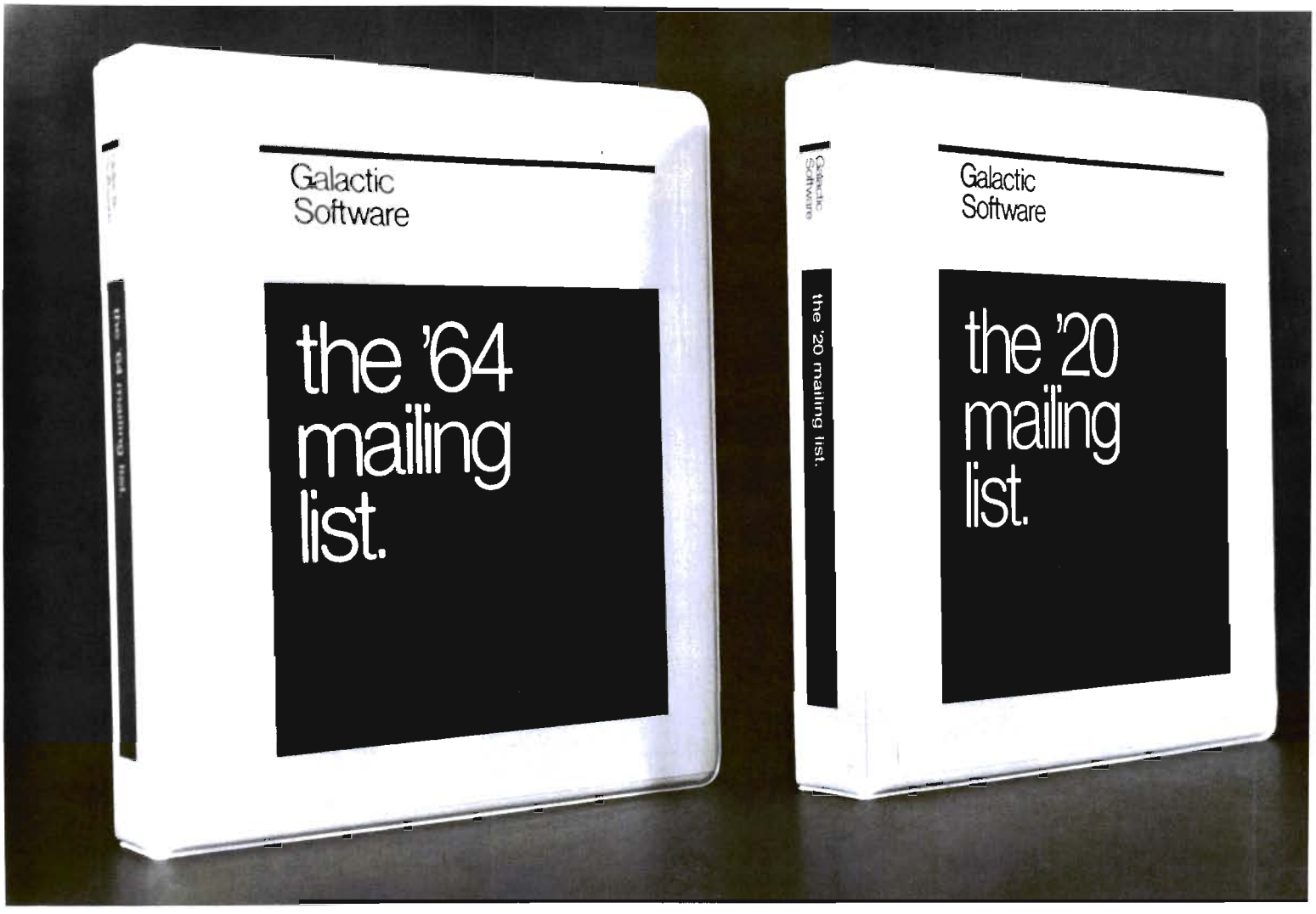
Order your copy today or see your dealer.
Will you get the top score?

RULES

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

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

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.

Deluxe COMSTAR F/T PRINTER — $349.00

The Comstar is an excellent addition to any micro-computer system. (Interfaces are available for Apple, VIC-20, Commodore-64, Pet, Atari 400 and 800, and Hewlett Packard) At only $349. the Comstar gives you print quality and features found only on printers costing twice as much. Compare these features.

- **BI-DIRECTIONAL PRINTING** with a LOGIC SEEKING CARRIAGE CONTROL for higher through-put in actual text printing. 80 characters per second.
- **PRINTING VERSATILITY:** standard 96 ASCII character set plus block graphics and international scripts. An EPROM character generator includes up to 224 characters.
- **INTERFACE FLEXIBILITY:** Centronics is standard. Options include EIA RS232C, 20mA Current Loop. (Add $20.00 for RS232)
- **LONG LIFE PRINT HEAD:** 100 million character life expectancy.
- **THREE SELECTABLE CHARACTER PITCHES:** • 10, 12 or 16.5 characters per inch. 132 columns maximum. Double-width font also is standard for each character pitch.
- **THREE SELECTABLE LINE SPACINGS:** 6, 8 or 12 lines per inch.
- **PROGRAMMABLE LINE FEED:** programmable length from 1/144 to 255/144 inches.
- **VERTICAL FORMAT CONTROL:** programmable form length up to 127 lines, useful for short or over-sized preprinted forms.
- **FRICTION AND TRACTOR FEED:** will accept single sheet paper.
- **224 TOTAL CHARACTERS**
- **USES STANDARD SIZE PAPER**

If you want more try —

Premium Quality COMSTAR F/T SUPER-10" PRINTER — $449

For $449.00 you get all of the features of the Comstar plus 10" carriage, 100 cps, 9 x 9 dot matrix with double strike capability for 18 x 18 dotmatrix. High resolution bit image (120 x 144 dot matrix), underlining, backspacing, 2.3K buffer, left and right margin settings, true lower descenders, with super and subscripts, and prints standard, Italic, Block Graphics, special characters, plus 2K of user definable characters. For the ultimate in price performance the Comstar F/T Super 10" leads the pack!

WE HAVE THE LOWEST PRICES

We sell to customers and you save the profit margin normally made by computer stores, department stores and distributors, we are willing to take a smaller margin to develop volume. WE LOVE OUR CUSTOMERS — OUR PRICES PROVE IT!

IMMEDIATE REPLACEMENT WARRANTY

If your printer fails because of warranty defect within 180 days from the date of purchase you simply send your printer to us via United Parcel Service prepaid. We will "immediately" send you a replacement printer at no charge via United Parcel Service prepaid. This warranty applies to all products we sell because WE LOVE OUR CUSTOMERS!

15 DAY FREE TRIAL

OTHER OPTIONS

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

Add $20.00 shipping, handling and insurance. Illinois residents please add 6% tax. Add $40.00 for CANADA, PUERTO RICO, HAWAII, ALASKA orders. WE DO NOT EXPORT TO OTHER COUNTRIES. Enclose cashiers check, money order or personal check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail available!! Canada orders must be in U.S. dollars.

PROTECTO ENTERPRIZES (FACTORY-DIRECT)
BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

COMSTAR F/T

ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmn
opqrstuvwxyz 1234567890
ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890

SUPER-10" ABCDEFGHIJKLMNOPQRSTUVWXYZ
ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890

Circle No 34

Arithmetic at Nursery School

by Dennis G. Smith
Marshall, MI

The VIC's potential in education is just beginning to be realized. In many ways, it is ideally suited for classroom use, especially in the early grades. There has been a bandwagon movement in education to purchase the bigger personal computers. Impressive simulation routines and multiple-choice of subprograms have convinced some school administrators and teachers that 32K, 48K, or 64K machines are necessary in the school environment. While simulations are valuable enrichment exercises, basic education can be packaged in smaller units. The VIC's low cost, color and sound capabilities, large print for brief and easy reading, and 3.5K limitation for short loading are ideal features for an elementary classroom computer.

The program offered here was written for my four-year-old daughter, Carol, who has been showing the ability to retain numerical facts. She, like most kids, loves to play with the VIC. I am sure, however, that this program can easily be used in grades, K-4.

The program title is "Addition and Subtraction Facts." A more appropriate name might be "Ands and Take-aways", however. The program flashes simple addition and subtraction problems onto the screen. It is a colorful program and uses arcade sounds. If the correct answer is typed, then the problem is blown away by a pistol, and a new one replaces it. If the wrong answer is typed, the bullet misses the problem, and the correct answer is given. If ten consecutive problems are answered correctly, a happy-face message is seen and heard. The opportunity to begin again is given after a wrong response and after ten problems are correctly answered. A teacher or parent will have to walk a young child through the program a few times at first, but my daughter can easily use it by herself now. As it is listed now, the most difficult problems are $9+4=?$ and $9-4=?$.

The number of problems given per game as well as the problem difficulty can easily be changed by editing line 20. For example, if you redefine the variables as follows:

```
20  K=15: A=1: B=1
```

then only the problems, $0+1=?$, $1+1=?$, $1+0=?$, $1-0=?$, $1-1=?$ can be displayed; and 15 problems, randomly selected from these, must be answered correctly to get the happy face. Answers in this program cannot exceed two digits; and therefore, the maximum values (A and B) must be chosen with this in mind.

This program was designed to introduce the youngster to arithmetic problems in the equation format. In the beginning it is advisable to use the simplest problems. If $A=8$ and $B=1$, then the identity equations and operations with unity will be generated for whole numbers, 0-8. After these are mastered, then increase the value of B, so that problems such as $5+3=?$ and $7-2=?$ can be displayed. The problems will remain on the screen until the child answers; this provides as much time as needed to count out the answer on fingers or mentally. Eventually the answers will become mostly cemented into memory as we have them as adults. I omitted the speed factor from this program, because my daughter showed extreme frustration when she was pressured to type in the answer quickly.

For older kids,the program could be used for more advanced mental arithmetic. For instance, problems such as $15+23=?$ and $62-29=?$ can be displayed. $A=50$ and $B=49$ yield the widest range of two-digit problems.

In my classroom, the seventh-graders hardly noticed the difference between a game and an educational tool. To this teacher, it seems criminal not to use kids' fascination with computers to their own advantage—starting as soon as possible.

I will save a copy of this program for you. Send $4, a self addressed mailer, and a blank cassette to Dennis G. Smith, 225 Highfield Road, Marshall, MI 49068.□

```
3 REM + AND - FACTS
5 V=36878:S1=36874
6 S4=36877:CO=30720
7 SB=36879:S3=36876
10 REM A+B=C OR A-B=C
11 REM K=# OF PROBLEMSPER GAME
12 REM A,B=MAX VALUES
20 K=10:A=9:B=4
28 POKESB,124
29 PRINT"[CLEAR]";
30 PRINT"[REV]                              [OF
F]";
31 PRINT"[REV]ADDITION & SUBTRACTION[OF
F]";
32 PRINT"[REV]                              [OF
F]";
33 PRINT"[REV]        FACTS                 [OF
F]";
34 PRINT"[REV]                              [OF
F]"
35 PRINT"[DOWN] [REV]3+4=7[OFF]   [REV]
5-2=3[OFF]"
36 PRINT"[DOWN][DOWN]      [REV]1+1=2[O
FF]  [REV]0+0=0[OFF]"
37 PRINT"[DOWN][DOWN] [REV]1+6=7[OFF]
      [REV]8-5=3[OFF]"
38 PRINT"[DOWN][DOWN]    [REV]3-3=0[OFF
]  [REV]6-5=1[OFF]"
50 PRINT"[DOWN][DOWN]   PRESS ANY KEY"
60 GETM$:IFM$=""THEN60
65 T=190:GOSUB1000
70 POKESB,169:PRINT"[BLACK]"
80 PRINT"[CLEAR]     [REV]DESCRIPTION[O
FF]      "
90 PRINT"[DOWN][DOWN]  ANSWER";K;"RIGHT
"
100 PRINT"IN A ROW AND YOU WIN."
110 PRINT"[DOWN][DOWN]  MISS A PROBLEM
AND"
120 PRINT"YOU MUST START OVER."
130 PRINT"[DOWN][DOWN][DOWN][DOWN][DOWN
][DOWN][DOWN][DOWN][DOWN]  PRESS ANY KE
Y"
140 GETM$:IFM$=""THEN140
150 POKESB,222
155 T=210:GOSUB1000
159 PRINT"[CLEAR]"
160 PRINT"[DOWN][DOWN]WHAT KIND OF PROB
LEM ?"
170 PRINT"[DOWN][DOWN]          +"
180 PRINT"[DOWN]        OR"
190 PRINT"[DOWN]          -"
200 GETO$:IFO$=""THEN200
210 IFO$<>"+"ANDO$<>"-"THEN200
240 IFO$="+"THEN260
```

```
250 PRINT"[UP][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT][RIGHT]X":T=240
251 GOSUB1000
252 GOTO265
260 PRINT"[UP][UP][UP][UP][UP][RIGHT][R
IGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]X":
T=240
261 GOSUB1000
265 FORN=1TO1000:NEXTN
266 T=230
267 GOSUB1000
270 PRINT"[DOWN][DOWN][DOWN][DOWN][DOWN
][DOWN]  PRESS ANY KEY TO"
271 PRINT"BEGIN THE PROBLEMS."
280 GETM$:IFM$=""THEN280
285 PRINT"[CLEAR]"
290 REM GUN
300 FORI=1TO12
310 READP,Q
320 POKEP,Q
330 POKEP+CO,0
340 NEXTI
345 FORN=1TO1000:NEXTN
350 DATA7922,160,7923,160,7900,233
360 DATA7901,160,7879,233,7880,160
370 DATA7857,95,7858,160,7859,160
380 DATA7860,160,7861,160,7862,120
389 J=0
390 REM PROBLEM
400 IFJ=KTHEN700
401 N1=INT(RND(1)*(A+1))
410 N2=INT(RND(1)*(B+1))
420 IFO$="-"ANDN2>N1THEN400
430 IFO$="-"THENC=N1-N2:GOTO460
440 IFO$="+"THENC=N1+N2
445 T=240:GOSUB1000
450 PRINT"[HOME][DOWN][DOWN][DOWN][DOWN
][DOWN][DOWN][DOWN][DOWN][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT]";N1;"+";N2;"= ?";
455 GOTO469
460 T=240:GOSUB1000
461 PRINT"[HOME][DOWN][DOWN][DOWN][DOWN
][DOWN][DOWN][DOWN][DOWN][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT]";N1;"-";N2;"= ?";
469 PRINT"[HOME][DOWN][DOWN][DOWN][DOWN
][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][D
OWN][DOWN][DOWN][DOWN][DOWN][DOWN]"
470 PRINT"TYPE ANSWER."
490 GETT$:IFT$=""THEN490
491 IFC<10 THEN 496
492 IFVAL(T$)<>INT(C/10)THEN510
493 GETN$:IFN$=""THEN493
```

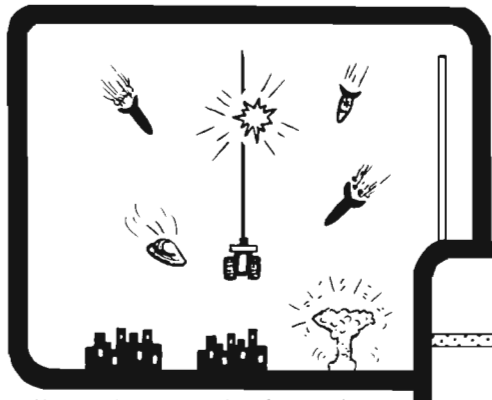
```
494 IFC-(10*VAL(T$)+VAL(N$))=0THEN600
495 GOTO510
496 IFVAL(T$)=CTHEN600
510 REM WRONG
515 POKEV,15:Q=7863:R=39
520 FOR I=0TO8
525 POKEQ+I,R:POKEQ+I+CO,0:POKES3,200-8
*I
526 FORF=1TO30 :NEXTF
527 POKEQ+I,32
535 IFI=3THENQ=7885
540 IFI>3THENQ=Q+22
545 NEXTI
550 POKES1,200:POKES3,0
555 FORI=1TO100:NEXTI
560 POKES1,0:POKEV,0
565 FORF=1TO100:NEXTF
570 IFO$="-"THEN575
571 PRINT"[CLEAR][DOWN][DOWN][RIGHT][RI
GHT]";N1;"+";N2;"= ";C:GOTO580
575 PRINT"[CLEAR][DOWN][DOWN][RIGHT][RI
GHT]";N1;"-";N2;"= ";C
580 FORF=1TO200:NEXTF
581 REM MENU
582 T=240:GOSUB1000
583 PRINT"[DOWN][DOWN][DOWN][DOWN][DOWN
]PRESS 'S' TO BEGIN"
584 PRINT"AGAIN."
585 PRINT"[DOWN][DOWN]PRESS 'C' TO CHAN
GE TO";
586 PRINT"+ OR -."
587 PRINT"[DOWN][DOWN]PRESS 'R' TO RUN"
588 PRINT"PROGRAM AGAIN."
589 PRINT"[DOWN][DOWN]PRESS 'E' TO END."
590 GETM$:IFM$=""THEN590
591 IFM$<>"S"ANDM$<>"C"ANDM$<>"R"ANDM$<
>"E"THEN 590
592 RESTORE:PRINT"[CLEAR]"
593 IFM$="S"THEN 300
594 IFM$="C"THEN160
595 IFM$="R"THEN5
596 GOTO800
600 REM CORRECT
610 J=J+1
630 REM FIRE
640 D=120:POKEV,15
644 FORN=1TO2
645 FORX=0TO3
650 POKE7863+X,D
655 POKE7863+X+CO,7
660 POKES3,200+15*X
665 NEXTX
670 D=32
671 NEXTN
675 POKES4,200:POKES3,0
681 FORX=1TO30:NEXTX
```

```
682 PRINT"[HOME][DOWN][DOWN][DOWN][DOWN
][DOWN][DOWN][DOWN][DOWN][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT]▌¬o| ⌒|—⌝\|—"
683 FORX=1TO30:NEXTX
684 PRINT"[HOME][DOWN][DOWN][DOWN][DOWN
][DOWN][DOWN][DOWN][DOWN][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT]\|x┘ └x\/\| |/"
685 FORX=1TO30:NEXTX
686 PRINT"[HOME][DOWN][DOWN][DOWN][DOWN
][DOWN][DOWN][DOWN][DOWN][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT]                 "
690 POKES4,0:POKEV,0
695 FORN=1TO200:NEXTN
696 GOTO400
700 REM K CORRECT
705 POKEV,15:POKES3,230
710 FORF=1TO400:NEXTF
715 POKES3,245
720 FORF=1TO400:NEXTF
725 POKES3,0
726 PRINT"[CLEAR][DOWN][DOWN][DOWN][RIG
HT][RIGHT][RIGHT][RIGHT]GOOD !"
727 PRINT"[DOWN][DOWN][DOWN]YOU ANSWERE
D";J;"RIGHT"
728 PRINT"IN A ROW."
730 FORF=1TO20
735 READP,Q
740 POKEP,Q:POKEP+CO,0:POKES3,200+2*F
745 NEXT F
750 DATA7953,78,7954,99,7955,99,7956,99
751 DATA7957,77,7974,78,7980,77,7996,10
1
752 DATA7998,81,8000,81,8002,103,8018,1
01
753 DATA8024,103,8040,101,8046,103,8062
,77
754 DATA8064,77,8065,100,8066,78,8068,7
8
760 POKES3,0:POKEV,0
765 PRINT"[HOME][DOWN][DOWN][DOWN][DOWN
][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][D
OWN][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN
][DOWN][DOWN][DOWN][RIGHT][RIGHT][RIGHT
]PRESS ANY KEY."
770 GETM$:IFM$=""THEN770
774 PRINT"[CLEAR]"
775 GOTO 581
800 END
1000 POKEV,15:POKES3,T
1001 FORI=1TO50:NEXTI
1002 POKEV,0:POKES3,0
1003 RETURN
READY.
```

D.E.S.-SOFT ™

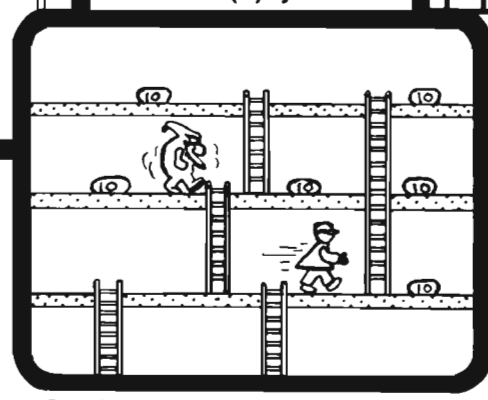
a division of
DES-Data Equipment Supply Corp.

LASER COMMAND by Bob Burnett



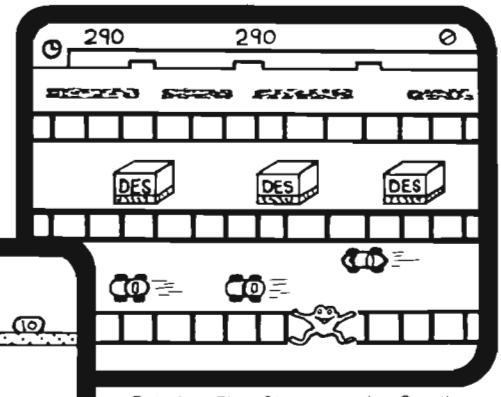
You are the commander of a squadron of laser ships. It is your duty to defend the cities of Earth against the alien onslaught. Spectacular graphics and machine code for super fast arcade fun. VIC-20 and cassette, joystick.
$20.00

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

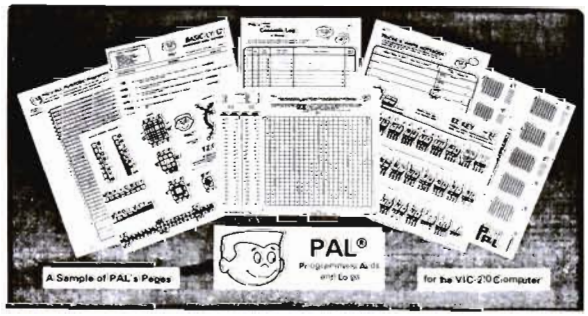
Watch the maze change as you pass thru the revolving doors. This is a really cute one. Machine language. VIC-20 w/8K expander, cassette. Joystick and keyboard.
$20.00

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

The Basics of Basic

by Tim Parker
Ontario, Canada

This series is being written in response to a large number of comments about the programs appearing in this (and other) magazines: namely that they seem to be total gibberish to many readers. A large number of writers tend to assume that most readers will have a sufficient aquaintance with BASIC and any machine's quirks that are required to get the most out of an article. I confess that I am probably one of the worst offenders. I hardly ever document my programs, as they all seem so simple to me. Alas, a mistake if ever there was one!

In the next few issues, I hope to take a quick yet thorough tour through BASIC as a language specifically applied to Commodore machines, then go into a few operation details on the different machines such as the VIC-20, VIC-64, the PETs and CBM machines.

At this point, though, I will begin at the beginning. I hope veteran users, and those who will quickly become veterans, will not be too bored by the following, but it is always best to begin at the beginning.

BASIC is an acronym for Beginners All-Purpose Symbolic Instruction Code. That really doesn't tell anyone much. When deciphered, and the intent of BASIC is included, it boils down to the following. A general purpose language applicable to most programming situations was sorely needed that didn't require a degree in computer science to learn. The obvious approach was to relate functions to their everyday English meaning. A PRINT statement should print something, while a STOP statement should halt operations. Naturally, some definitions had to be created. These were done with the most logical approach possible that still retained some sensible meaning.

BASIC quickly grew in popularity, and was refined in complexity and ability to the state it now exists in. Although several different BASICs have been created, most with common features, there is no set standard to be adhered to. A de facto standard was approached by the massive adoption of a BASIC version by Microsoft. It quickly became universal in its use, on almost every machine imaginable. Thereafter, most BASIC's have been patterned on Microsoft's lead.

To learn BASIC requires only a computer and a well written manual. There are dozens of BASIC instruction books on the market, and most are well written. Some, unfortunately, are not. The only way to learn a language is to write programs in it. That is why a book is so useful. Examples are usually given of all functions, and most popular books approach BASIC in a tiered level, introducing the complexity levels one by one. There is no point in repeating the BASIC instruction books here. To do so would require over a year's worth of articles. So a neophyte BASIC user should rush out and browse through the local bookstore or library's shelves. Find a book that is easy to read, and appears to be written in a competent manner. Avoid the "cutesy" books. They tend to slow you down, rather than help.

Several things require pointing out. First is that for all Commodore computers (in fact, for any computer that doesn't compile the program) line numbers must be used. Each line of instruction code has to be prefaced by a number that is usually less than 65,536. It doesn't matter what order the lines are written in, as the BASIC interpreter will arrange them in ascending order. Remember that they will be executed in order, though. It also doesn't matter what number you start with, or what interval you use between line numbers. There are a couple of considerations to this, though. Inevitably, you will want to add lines in between the existing lines. Therefore, leave enough line numbers free between each line for expansion. Also, it is nice to be able to structure your programs to an extent, so that specific line numbers do specific jobs. If subroutines are included (if you don't understand what a subroutine is, don't worry) it is convenient to give them a numerical sequence to themselves.

Examine a few published programs, and see if they follow the above guidelines. Personally, I usually leave ten lines between each line number for expansion (i.e. I increment by ten: 210, 220, etc.) because ten is a convenient number, and it is easy to count by tens! When starting a program, I'll usually begin at one hundred or so, as I will probably want to add some instructions or definitions at the top of the program. Subroutines, if they are major (such as Klingon ships moving, or pirates attacking), will be done in blocks of thousands. As an example, if a Trek game is in the works (See Commander, January), the different functions such as moving, firing, computers, etc, will be at 1000's, 2000's, 3000's, etc. This allows fast access to the required routine, and helps a programmer "remember" where he did this or that.

Most BASICs allow multiple commands per line. By this I mean that more than one instruction can be given per line number, as long as they are separated by a colon. (This is a technique to save memory...more on that in another column.) While it may be easy to type many commands on one line, for many applications, it is not too good an idea. Usually, programmers will do functions such as INPUTs and response analysis on separate lines, although to "compartmentalize" the program, they can be combined.

One big feature that should be included in every program, but seldom is, is documentation. Judicious use of the REM (remark) statement will help in later analysis and error fixing. This is great advice, but no one ever follows it (especially me)! I have had several cases where I have written programs for specific tasks, such as games, calculations, sorts, etc. and they have worked perfectly. They are then stored on a shelf for a few months to gather dust. At some point later, I'll dig it out and attempt to modify it, but completely forget what each variable does, or why that loop is there, or what that cryptic GOSUB is doing. Usually, I spend more time analyzing the program than I saved in the first place. (Several times I have found it easier to rewrite the whole thing!) The moral is that documentation will save headaches and lot of hassle in the future. It doesn't have to be elaborate. Just stick in a REM that says what each variable does, and one at the start of a loop or subroutine that explains its purpose.

While mentioning variable names, it's a good idea to mention that in the same sense that documentation can help your programming, variable names can do the same. Try to use a name that is somewhat indicative of its purpose. Most BASICs on the Commodore machines only recognize two characters at the beginning of the variable. It is difficult to give a descriptive label in two letters, but it beats using X's and Y's all over the place! Remember also that variables can be "reused" to save memory. If a loop at the top used variable X1 as a counter, and that is its only use, when another loop occurs, use the same variable. Each variable named requires memory space.

Ensure that the variable types are known. With Commodore BASIC versions, this is not a difficult task. Most character variables (those that are not just numbers) must have a dollar sign tagged on the end to identify itself. Some BASICs require other identifiers, too.

Begin programming with the "easy" commands until their use is determined. Granted, there are only so many things that can be done with PRINT, INPUT, and GOTO, but there is no point in using the commands such as LEFT$, CHR$, etc. if the basics are not mastered. This may seem obvious, but appears to be the major downfall of most programmers at the early stage.

For those who are eager to get on without, and don't want to waste time reading book after book of theory, use the instruction manual that came with your computer. Most come with an explanation of BASIC. Write a specific program, not one of their trivial examples (although do read their examples to see how it's done). Design a program that you see as a challenge to your programming skills at that point. The satisfaction gained by finishing the task is tremendous, and will aspire you to greater heights!

At this point, some definitions can be introduced for future use. First, the difference between interpretive and compiled languages should be understood. An interpretive BASIC uses an "interpreter" routine to decipher any code you may have written. For example, the instruction PRINT "TEST" means absolutely nothing to the computer as it stands. The interpreter has to read the above, and realize that it is telling the computer to send to the screen the word TEST. The interpreter is exactly that. It understands what you say in one language, and translates it to a language the computer can understand. (That is why when a programmer writes in a language the machine understands directly, it is called machine language programming.) The interpreter is usually a pretty big program, which itself is written in machine code. All that just to make your programming easier!

A compiled language takes the instructions you write, and like an interpreter, changes it all to machine code. The difference is, the program is changed to code, and can be saved on storage media as a machine code program. An interpreter requires that it interprets every time through. Obviously, a compiled language will be much quicker, as it doesn't have to go through the translation step. It also has several problems, in that if a minor change is made to a program, it has to be recompiled. Compilation itself is a longish project.

The major fault to compiled programs is that they cannot easily be debugged. With interpretive BASICs, if a line is not behaving correctly, you simply change the line and RUN the program again. With a compiled program, the source code has to be reloaded, changed, and compiled.

Enough for this month. In the next installment, a look at some of the more advanced features of BASIC, and how they can best be applied. □

Machine Language I/O: Part Two of Three

by Howard N. Rotenberg
Toronto, Canada

In my last article on machine language I/O, we discussed two aspects of this somewhat difficult and undocumented subject. The first part dealt with a machine language input routine and the second with a method to open files on the disk drive. The article was accompanied by two small sample programs. This article will side step a little bit from what seems the logical pattern of the total concept that I am trying to get across. It will all come together as mentioned in part one of the article, when we get to part three. If you have not read part one, I recommend that you do if you want to follow the flow and understand the routines that I am piecing together. If you have just joined in and wish to start from here, I am sure that it will present no problem as this section should be able to stand on its own.

I would like to briefly explain the routines that will be used in our two sample programs. The two programs will be a small PET to PET terminal program for the IEEE modem followed by another terminal program that will converse in ASCII to any other computer or mainframe. I will go through the routines in the order that they appear in the first program listing. Note: Both programs use the same routines.

The Routines

OPENI—This routine is often referred to as SET INPUT DEVICE. The X register holds the logical file number of the open file. This will make the IEEE device the talker and BASIC 4 will clear DS$ and ST. The three errors that may occur are: FILE NOT OPEN, NOT INPUT FILE or DEVICE NOT PRESENT.

Status

This is the status byte that ST is computed from. Most of us use it in our BASIC programs to detect different I/O conditions. The possible values are:

1) 1—for Timeout on Write when the IEEE bus is the listener.

2) 2—for Timeout on Read when the IEEE bus is the talker.

3) 4—for a short block on a cassette read, verify or load.

4) 8—for a long block on a cassette read, verify or load.

5) 16—for an Unrecoverable Read Error on cassette read or Any mismatch on a tape verify or load.

6) 32—for a checksum error on a cassette read, verify or load.

7) 64—for End of File marker.

8) 128—for end of tape on cassette read, verify or load and Device not present on the IEEE bus.

WRITE—This is the same routine called by a BASIC print or print#. It is used to output the contents of the accumulator to any device specified. The values of A, X and Y are not changed.

GETCHR—This will take the one character from the keyboard input buffer. The values of X and Y are not changed.

OPENO—This routine is often refered to as Set Output Device. The X register holds the logical file number and the IEEE device becomes a listener.

CLEAR—This will clear open channels to the IEEE bus. Before calling this routine, the file number should be held in the X register.

OPEN—This routine is identical to the BASIC open. The memory locations $D1-$D4 must be previously set up. These locations will be discussed shortly.

CLOSE—This routine follows the same rules as the BASIC open statement with the exception that the location $D1 is of no consequence.

DFAULT—This routine will set the computer back to its default devices as on power up. The screen (device 3) is made the output device and the keyboard (device 0) is set to the input device. Any output device on the IEEE bus is UNLISTED and any input device is sent the UNTALK command. This routine does not affect the X and Y registers nor does it close any open files.

TALK—This routine will set the desired device to be the talker. EG. If you print the listing to your printer, the printer is called the listener while your computer is the talker. If you are spooling then your disk drive will have been set to be the talker.

LISTEN—This routine sets the desired device to be the listener. The previous example covers this subroutine also.

UNTALK and LISTEN—I will just briefly say that these two commands are the opposite of their counterparts previously covered. The actual routines used along with the registers and the IEEE bus lines are a bit complicated to cover at this time.

GETIEE—This routine will set one character from the IEEE bus and return it in the accumulator. It must return a character within 64 milliseconds or ST will be set to 2 indicating a timeout.

CHKSTP—This simply checks for the stop key and returns a 0 if it was pressed.

The only locations that we will use in the program that I have not mentioned in the above routines are $D1 to $D4. The location $D1 must contain the number of characters in the file number. We will use a 0 in our exam-

ple since we will be opening a file to a modem that needs no file name. Just the same, the location must be set. The location $D2 must contain the current logical file number that we are about to open. The last two locations, $D3 and $D4, must contain the current secondary address or the R/W command and the current device number respectively. Since we are not using a secondary address, we will set $D3 to 255 and $D4 to our device number which happens to be 5.

This covers most of the information that will be of importance to the workings of the sample programs, so with this under our belt, we will carry on with the workings of the programs.

Programs

The program that we will mainly discuss will be Program 1. This is the PET to PET terminal program. The second program is almost identical with the exception of two additional routines that are used to translate from PET ASCII to ASCII and VICA-VERSA. I will discuss those routines after we are finished with the main topic.

The constants are declared at the beginning, followed by the load address which happens to be $9000 or decimal 36864. This may of course be changed at the user's discretion.

The first thing we must do is to open the files we are going to use. In this case we will be opening a file to an IEEE modem with a device number of five. We will also use five as the logical file number to open with. As stated earlier, we will store the logical file number and device number in the memory locations $D2 and $D4 respectively. Since we are using no file name, we store a zero for the number of characters in the file name in $D1. Lastly, after storing 255 into memory location $D3, indicating no secondary address, we are ready to jump to the subroutine OPEN.

The file to the modem is now open providing we did not get any of the possible errors that were mentioned earlier. The main body of the program is a series of seven instructions that will call the appropriate subroutines for our program. I will use these instructions for our guideline.

The first thing we do is JSR to a subroutine called RECV. This routine is used to get one character from the modem and display it. The beginning of RECV saves the contents of the accumulator since other routines also use it. We send a talk command to the modem to inform it that it will be the device to send us any information it has in its buffer. We must open the modem for input since we want it to get any character that may come over the telephone line. When we open it for input, we must use the X register for the file number. Considering that the modem is now open for input, we may restore our accumulator. We then use the subroutine GETIEE to get a character from the modem buffer if one exists. At this point we check the status and if it is not equal to zero then we skip the instruction to display the character. If the status was zero then there was a valid character to display and we do so. Before going back to our main routine, we must clear the channel and tell the modem that it is no longer a talker. The X register, which still contains our file number, is used for these subroutines.

This takes us back to our main program which will not check for the depression of the stop key. If it was pressed, we branch to QUIT to end the program. I will discuss QUIT after the rest of the program.

We now JSR to GETCHR which will look at our keyboard buffer to see if we have entered any characters. If there is nothing there, it branches back to MAIN and goes back to check the modem again. If it fails the test then we must have entered some character so the program will go to the subroutine called XMIT.

This routine once again starts off by saving the contents of the accumulator on the stack since it now contains the character we want to transmit. The modem is sent the listen command since it will be receiving our character. It is now opened for output, once again using the X register since it will be transmitting the character. After the open, we may restore our accumulator's contents to get ready to send our character. We use the JSR WRITE that now acts just like a BASIC print#5. As you can see from this that a single subroutine may be used to send a character to whatever device we want by defining what the output device is to be. We then send a clear command just as before to clear the channel followed by an unlisten to cancel our listen command. One difference here is that we use the subroutine DFAULT to restore our default devices. I.E. Keyboard as input and Display as output.

At this point, we return to our main program and continue until the stop key is pressed. We will now assume that the stop key has been pressed and causes the branch to QUIT, which will get us back to BASIC after a little cleanup. All we have to do now is clear the channels we have been using and close any files that we may have opened. Since we only used one, this job is quite brief. It is important to notice that the X register has been used to clear the channel and the accumulator has been used to close the file. This is a must for a good close and clean exit from the program.

ASCII Program

The second part of this article just makes the first terminal program more flexible. It deals with the second program which is nearly a carbon copy of the first as far as the main routines are concerned. The big difference is that this program may be used as an ASCII terminal program which will enable you to converse with almost any other computer.

There are two additional subroutines that do all the work for us. These are called TOASC and FASCII. Their purpose is to take the Commodore's character set and translate it to the standard form of codes used by most other systems. There are a few different ways to do this conversion. One of the fastest would be to construct a table of the character codes to convert. This would, however, take up more memory. I chose to calculate the value of the proper character for transmission or reception. This is

similar to the way you would reverse upper and lower case characters in BASIC.

The routine FASCII is called from the receiving subroutine if there was a character received. This will convert the ASCII character received into the Commodore character set before displaying it. The other routine TOASC is called by the main program if there is a character in the keyboard buffer to send. This will ensure that the translation will be done before sending it. I choose not to explain the routines that convert since I hope that the comments will be sufficient. It is also not really an I/O operation and I would like to stay within that capacity in these articles.

Upgrade Basic Anyone?

The subroutines that I have used in the terminal programs are for BASIC 4. If you are still using UPGRADE BASIC, then the substitutions should be made. (See BASIC Upgrade box.) Note: The stars represent no change.

Subroutine Substitutions

OPENI	$F7AF	$F770	
STATUS	$96	$96	*
WRITE	$FFD2	$FFD2	*
GETCHR	$FFE4	$FFE4	*
OPENO	$F7FE	$F7BC	
CLEAR	$F2A6	$F272	
OPEN	$F563	$F524	
CLOSE	$F2E2	$F2AE	
DFAULT	$FFCC	$FFCC	*
TALK	$F0D2	$F0B6	
UNTALK	$F1AE	$F17F	
LISTEN	$F0D5	$F0BA	
UNLISN	$F1B9	$F183	
GETIEE	$F1C0	$F18C	
CHKSTP	$F335	$F301	

Conclusion

This takes us to the end of part two. In the last part, we will bring everything together and hopefully all questions will be answered. Once again, I would like to say that all listings have been assembled using COMMODORE's assembler.

```
LINE# LOC   CODE        LINE
00001  0000            ;PUT"@0:MS"
00002  0000            ;
00003  0000            ;+++++++++++++++++++++++++++++++++++++
00004  0000            ;+        ASCII TERMINAL PROGRAM     +
00005  0000            ;+        VERSION 1.0                +
00006  0000            ;+        82/04/18                   +
00007  0000            ;+        HOWARD ROTENBERG           +
00008  0000            ;+++++++++++++++++++++++++++++++++++++
00009  0000            ;
00010  0000            OPENI   = $F7AF       ;OPEN FOR INPUT
00011  0000            OPENO   = $F7FE       ;OPEN FOR OUTPUT
00012  0000            STATUS  = $96         ;ST
00013  0000            WRITE   = $FFD2       ;PRINT A CHR
00014  0000            GETCHR  = $FFE4       ;GET A CHR
00015  0000            CLEAR   = $F2A6       ;CLEAR ALL CHANNELS
00016  0000            OPEN    = $F563       ;OPEN FILE
00017  0000            CLOSE   = $F2E2       ;CLOSE FILE
00018  0000            DFAULT  = $FFCC       ;RESTORE DEFAULT DEVICE
00019  0000            TALK    = $F0D2       ;SEND TALK
00020  0000            UNTALK  = $F1AE       ;SEND UNTALK
00021  0000            LISTEN  = $F0D5       ;SEND LISTEN
00022  0000            UNLISN  = $F1B9       ;SEND UNLISTEN
00023  0000            GETIEE  = $F1C0       ;HANDSHAKE BYTE IN
00024  0000            CHKSTP  = $F335       ;CHECK FOR STOP KEY

00026  0000            ;
00027  0000                    * = $9000     ;LOAD ADDRESS
00028  9000            ;
00029  9000            ;       OPEN FILES
00030  9000            ;
00031  9000  A9 05             LDA #5        ;GET DEVICE NUMBER
00032  9002  85 D2             STA $D2       ;STORE LFN
00033  9004  85 D4             STA $D4       ;STORE DEVICE NUMBER
00034  9006  A9 00             LDA #0
00035  9008  85 D1             STA $D1       ;NO FILE NAME
00036  900A  A9 FF             LDA #255
00037  900C  85 D3             STA $D3       ;NO SECONDARY ADDRESS
00038  900E  20 63 F5          JSR OPEN      ;OPEN MODEM
00039  9011            ;
00040  9011            ;       MAIN PROGRAM
00041  9011            ;
00042  9011  20 74 90  MAIN    JSR RECV      ;RECEIVE CHAR
00043  9014  20 35 F3          JSR CHKSTP    ;STOP PRESSED
00044  9017  F0 0E             BEQ STPRES    ;YES
00045  9019  20 E4 FF          JSR GETCHR    ;GET CHARACTER
00046  901C  F0 F3             BEQ MAIN      ;NOTHING THERE
00047  901E  20 2A 90          JSR TOASC     ;TRANSLATE TO ASCII
00048  9021  20 92 90          JSR XMIT      ;TRANSMIT CHARACTER
00049  9024  4C 11 90          JMP MAIN      ;LOOP BACK
00050  9027  4C A9 90  STPRES  JMP QUIT      ;GO TO END AND CLOSE FILES
00051  902A            ;
00052  902A            ;       CHANGE PET TO ASCII
00053  902A            ;
00054  902A  C9 41      TOASC   CMP #$41       ;<= LOWER CASE 'A'
00055  902C  30 18             BMI ADELET     ;YES THEN CHECK FOR DELETE
```

```
LINE# LOC   CODE        LINE

00056  902E  C9 5B        CMP #$5B     ;>= LOWER CASE 'Z' ($5A)
00057  9030  10 06        BPL NXTCHK   ;YES GOTO NEXT CHECK
00058  9032  18           CLC          ;NO
00059  9033  69 20        ADC #$20     ;CONVERT TO UPPER CASE
00060  9035  4C 43 90     JMP EXIT     ;RETURN TO MAIN ROUTINE
00061  9038  C9 C1     NXTCHK CMP #$C1  ;<= UPPER CASE 'A'
00062  903A  30 07        BMI EXIT     ;YES THEN NO CHANGE
00063  903C  C9 DB        CMP #$DB     ;>= UPPER CASE 'Z' ($DA)
00064  903E  10 03        BPL EXIT     ;YES SO NO CONVERT
00065  9040  38           SEC          ;NO
00066  9041  E9 80        SBC #$80     ;CONVERT TO LOWER CASE
00067  9043  29 7F     EXIT   AND #$7F  ;MASK 7'TH BIT
00068  9045  60           RTS          ;BACK TO MAIN ROUTINE
00069  9046  C9 14     ADELET CMP #$14  ;IS IT A DELETE
00070  9048  D0 F9        BNE EXIT     ;NO SO GET OUT
00071  904A  18           CLC          ;YES
00072  904B  A9 08        LDA #$8      ;CHANGE TO BACKSPACE
00073  904D  4C 43 90     JMP EXIT     ;AND GET OUT
00074  9050              ;
00075  9050              ;     CHANGE ASCII TO PET
00076  9050              ;
00077  9050  C9 41     FASCII CMP #$41    ;<= UPPER CASE 'A'
00078  9052  30 16        BMI PDELET ;YES THAN CHECK FOR BACKSPACE
00079  9054  C9 5B        CMP #$5B     ;>= UPPER CASE 'Z' ($5A)
00080  9056  10 06        BPL CHKNXT   ;YES GOTO NEXT CHECK
00081  9058  18           CLC          ;NO
00082  9059  69 80        ADC #$80     ;CONVERT TO PET LOWER CASE
00083  905B  4C 69 90     JMP OUT      ;RETURN TO MAIN ROUTINE
00084  905E  C9 61     CHKNXT CMP #$61  ;<= LOWER CASE 'A'
00085  9060  30 07        BMI OUT      ;YES SO NO CHANGE
00086  9062  C9 7B        CMP #$7B     ;>= LOWER CASE 'Z' ($7A)
00087  9064  10 03        BPL OUT      ;YES SO NO CONVERT
00088  9066  38           SEC          ;NO
00089  9067  E9 20        SBC #$20     ;CONVERT TO PET UPPER CASE
00090  9069  60        OUT    RTS      ;BACK TO MAIN ROUTINE
00091  906A  C9 08     PDELET CMP #$8   ;IS IT A BACKSPACE
00092  906C  D0 FB        BNE OUT      ;NO SO GET OUT
00093  906E  18           CLC          ;YES
00094  906F  A9 14        LDA #$14     ;CHANGE TO PET DELETE
00095  9071  4C 69 90     JMP OUT      ;AND GET OUT
00096  9074              ;
00097  9074              ;     RECEIVING ROUTINE
00098  9074              ;
00099  9074  48        RECV   PHA       ;SAVE CHARACTER
00100  9075  20 D2 F0     JSR TALK     ;SEND TALK
00101  9078  A2 05        LDX #5       ;FILE NUMBER
00102  907A  20 AF F7     JSR OPENI    ;OPEN MODEM FOR INPUT
00103  907D  68           PLA          ;RESTORE CHARACTER
00104  907E  20 C0 F1     JSR GETIEE   ;GET CHAR FROM MODEM
00105  9081  A4 96        LDY STATUS   ;GET STATUS
00106  9083  D0 06        BNE NOBYTE   ;NOT = 0
00107  9085  20 50 90     JSR FASCII   ;TRANSLATE TO PET
00108  9088  20 D2 FF     JSR WRITE    ;WRITE CHARACTER TO CRT
00109  908B  20 A6 F2  NOBYTE JSR CLEAR ;CLEAR CHANNEL
00110  908E  20 AE F1     JSR UNTALK   ;UNTALK
```

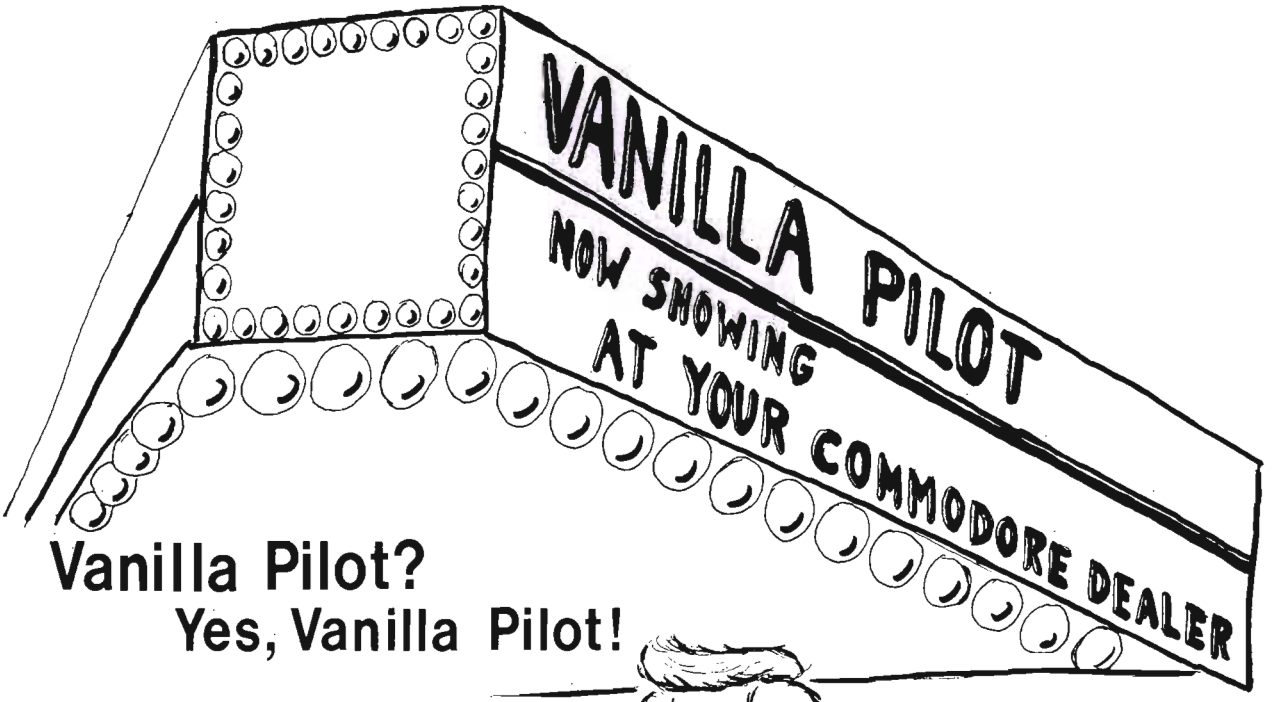
```
 LINE# LOC   CODE        LINE
00111  9091  60                  RTS          ;BACK TO MAIN ROUTINE
00112  9092                ;
00113  9092                ;     TRANSMIT ROUTINE
00114  9092                ;
00115  9092  48            XMIT  PHA          ;SAVE CHARACTER
00116  9093  20 D5 F0           JSR LISTEN   ;SEND LISTEN
00117  9096  A2 05               LDX #5       ;FILE NUMBER
00118  9098  20 FE F7           JSR OPENO    ;OPEN CHANNEL FOR OUTPUT
00119  909B  68                 PLA          ;RESTORE CHARACTER
00120  909C  20 D2 FF           JSR WRITE    ;PRINT#5
00121  909F  20 A6 F2           JSR CLEAR    ;CLEAR CHANNEL
00122  90A2  20 B9 F1           JSR UNLISN   ;SEND UNLISTEN
00123  90A5  20 CC FF           JSR DFAULT   ;RESTORE DEFAULT DEVICE
00124  90A8  60                 RTS          ;BACK TO MAIN ROUTINE
00125  90A9                ;
00126  90A9                ;     CLOSE FILES
00127  90A9                ;
00128  90A9  A2 05          QUIT  LDX #5       ;FILE NUMBER
00129  90AB  20 A6 F2           JSR CLEAR    ;CLEAR CHANNELS
00130  90AE  A9 05               LDA #5       ;FILE NUMBER
00131  90B0  20 E2 F2           JSR CLOSE    ;CLOSE FILE
00132  90B3  60                 RTS          ;EXIT TERMINAL MODE
00133  90B4                     .END


ERRORS = 00000


SYMBOL TABLE

SYMBOL VALUE
 ADELET   9046    CHKNXT   905E    CHKSTP   F335    CLEAR    F2A6
 CLOSE    F2E2    DFAULT   FFCC    EXIT     9043    FASCII   9050
 GETCHR   FFE4    GETIEE   F1C0    LISTEN   F0D5    MAIN     9011
 NOBYTE   908B    NXTCHK   9038    OPEN     F563    OPENI    F7AF
 OPENO    F7FE    OUT      9069    PDELET   906A    QUIT     90A9
 RECV     9074    STATUS   0096    STPRES   9027    TALK     F0D2
 TOASC    902A    UNLISN   F1B9    UNTALK   F1AE    WRITE    FFD2
 XMIT     9092

END OF ASSEMBLY


          CROSS REFERENCE.....PAGE    1


           ADELET $9046    55    69
           CHKNXT $905E    80    84
           CHKSTP $F335    24    43
           CLEAR  $F2A6    15   109   121   129
           CLOSE  $F2E2    17   131
           DFAULT $FFCC    18   123
           EXIT   $9043    60    62    64    67    70    73
```

```
FASCII $9050    77   107
GETCHR $FFE4    14    45
GETIEE $F1C0    23   104
LISTEN $F0D5    21   116
MAIN   $9011    42    46    49
NOBYTE $908B   106   109
NXTCHK $9038    57    61
OPEN   $F563    16    38
OPENI  $F7AF    10   102
OPENO  $F7FE    11   118
OUT    $9069    83    85    87    90    92    95
PDELET $906A    78    91
QUIT   $90A9    50   128
RECV   $9074    42    99
STATUS $0096    12   105
STPRES $9027    44    50
TALK   $F0D2    19   100
TOASC  $902A    47    54
UNLISN $F1B9    22   122
UNTALK $F1AE    20   110
WRITE  $FFD2    13   108   120
XMIT   $9092    48   115
```

```
LINE# LOC   CODE   LINE

00001  0000          ;PUT"@0:MODEM2.SRC"
00002  0000          ;
00003  0000          ;+++++++++++++++++++++++++++++++++++++++++
00004  0000          ;+       SAMPLE TERMINAL PROGRAM        +
00005  0000          ;+    FOR PET TO PET COMMUNICATIONS     +
00006  0000          ;+     TO SHOW EXAMPLE OF OPENING       +
00007  0000          ;+ AN IEEE DEVICE FOR INPUT AND OUTPUT  +
00008  0000          ;+            BY HOWARD ROTENBERG        +
00009  0000          ;+              TORONTO CANADA           +
00010  0000          ;+++++++++++++++++++++++++++++++++++++++++
00011  0000          ;
00012  0000          OPENI  = $F7AF          ;OPEN FOR INPUT
00013  0000          STATUS = $96            ;ST
00014  0000          WRITE  = $FFD2          ;PRINT A CHR
00015  0000          GETCHR = $FFE4          ;GET A CHR
00016  0000          OPENO  = $F7FE          ;OPEN FOR OUTPUT
00017  0000          CLEAR  = $F2A6          ;CLEAR ALL CHANNELS
00018  0000          OPEN   = $F563          ;OPEN FILE
00019  0000          CLOSE  = $F2E2          ;CLOSE FILE
00020  0000          DFAULT = $FFCC          ;RESTORE DEFAULT DEVICE
00021  0000          TALK   = $F0D2          ;SEND TALK
00022  0000          UNTALK = $F1AE          ;SEND UNTALK
00023  0000          LISTEN = $F0D5          ;SEND LISTEN
00024  0000          UNLISN = $F1B9          ;SEND UNLISTEN
00025  0000          GETIEE = $F1C0          ;HANDSHAKE BYTE IN
00026  0000          CHKSTP = $F335          ;CHECK FOR STOP KEY
00027  0000          ;
```

```
00028  0000                    * = $9000 ;LOAD ADDRESS
00029  9000              ;
00030  9000              ;       OPEN FILES
00031  9000              ;
00032  9000  A9 05             LDA #5    ;GET DEVICE NUMBER
00033  9002  85 D2             STA $D2   ;STORE LFN
00034  9004  85 D4             STA $D4   ;STORE DEVICE NUMBER
00035  9006  A9 00             LDA #0    ;GET A 0 FOR
00036  9008  85 D1             STA $D1   ;NO FILE NAME
00037  900A  A9 FF             LDA #255  ;GET 255 TO
00038  900C  85 D3             STA $D3   ;STORE SECONDARY ADDRESS
00039  900E  20 63 F5          JSR OPEN  ;OPEN MODEM
00040  9011              ;
00041  9011              ;       MAIN PROGRAM
00042  9011              ;
00043  9011  20 24 90 MAIN    JSR RECV  ; RECEIVE CHAR
00044  9014  20 35 F3          JSR CHKSTP;STOP PRESSED
00045  9017  F0 3D             BEQ QUIT  ; YES QUIT
00046  9019  20 E4 FF          JSR GETCHR;GET CHARACTER
00047  901C  F0 F3             BEQ MAIN  ;WHEN NOTHING
00048  901E  20 3F 90          JSR XMIT  ;TRANSMIT CHARACTER
00049  9021  4C 11 90          JMP MAIN  ;LOOP BACK
00050  9024              ;
00051  9024              ;       RECEIVING ROUTINE
00052  9024              ;
00053  9024  48         RECV    PHA       ;SAVE CONTENTS OF ACCUMULATOR
00054  9025  20 D2 F0          JSR TALK  ;SEND TALK
00055  9028  A2 05             LDX #5    ;FILE NUMBER


 LINE# LOC   CODE      LINE


00056 902A 20 AF F7          JSR OPENI ;OPEN MODEM FOR INPUT
00057 902D 68                PLA       ;RESTORE CONTENTS OF ACCUMULATOR
00058 902E 20 C0 F1          JSR GETIEE;GET CHAR FROM MODEM
00059 9031 A4 96             LDY STATUS;GET STATUS
00060 9033 D0 03             BNE NOBYTE
00061 9035 20 D2 FF          JSR WRITE ;WRITE CHARACTER TO CRT
00062 9038 20 A6 F2 NOBYTE   JSR CLEAR ;CLEAR CHANNEL
00063 903B 20 AE F1          JSR UNTALK;UNTALK
00064 903E 60                RTS       ;BACK TO MAIN ROUTINE
00065 903F              ;
00066 903F              ;       TRANSMIT ROUTINE
00067 903F              ;
00068 903F 48            XMIT    PHA       ;SAVE CONTENTS OF ACCUMULATOR
00069 9040 20 D5 F0          JSR LISTEN;SEND LISTEN
00070 9043 A2 05             LDX #5    ;FILE NUMBER
00071 9045 20 FE F7          JSR OPENO ;OPEN CHANNEL FOR OUTPUT
00072 9048 68                PLA       ;RESTORE CONTENTS OF ACCUMULATOR
00073 9049 20 D2 FF          JSR WRITE ;PRINT#5
00074 904C 20 A6 F2          JSR CLEAR ;CLEAR CHANNEL
00075 904F 20 B9 F1          JSR UNLISN;SEND UNLISTEN
00076 9052 20 CC FF          JSR DFAULT;RESTORE DEFAULT DEVICE
00077 9055 60                RTS       ;BACK TO MAIN ROUTINE
```

```
00078  9056                    ;
00079  9056                    ;       CLOSE FILES AND RETURN TO BASIC
00080  9056                    ;
00081  9056  A2 05             QUIT    LDX #5          ;FILE NUMBER
00082  9058  20 A6 F2                  JSR CLEAR       ;CLEAR CHANNELS
00083  905B  A9 05                     LDA #5          ;FILE NUMBER
00084  905D  20 E2 F2                  JSR CLOSE       ;CLOSE FILE
00085  9060  60                       RTS
00086  9061                          .END

ERRORS = 00000

SYMBOL TABLE

SYMBOL VALUE
 CHKSTP   F335    CLEAR    F2A6    CLOSE    F2E2    DFAULT   FFCC
 GETCHR   FFE4    GETIEE   F1C0    LISTEN   F0D5    MAIN     9011
 NOBYTE   9038    OPEN     F563    OPENI    F7AF    OPENO    F7FE
 QUIT     9056    RECV     9024    STATUS   0096    TALK     F0D2
 UNLISN   F1B9    UNTALK   F1AE    WRITE    FFD2    XMIT     903F

END OF ASSEMBLY

          CROSS REFERENCE.....PAGE    1

   CHKSTP $F335    26    44
   CLEAR  $F2A6    17    62    74    82
   CLOSE  $F2E2    19    84
   DFAULT $FFCC    20    76
   GETCHR $FFE4    15    46
   GETIEE $F1C0    25    58
   LISTEN $F0D5    23    69
   MAIN   $9011    43    47    49
   NOBYTE $9038    60    62
   OPEN   $F563    18    39
   OPENI  $F7AF    12    56
   OPENO  $F7FE    16    71
   QUIT   $9056    45    81
   RECV   $9024    43    53
   STATUS $0096    13    59
   TALK   $F0D2    21    54
   UNLISN $F1B9    24    75
   UNTALK $F1AE    22    63
   WRITE  $FFD2    14    61    73
   XMIT   $903F    48    68
```



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

AARDVARK — THE ADVENTURE PLACE

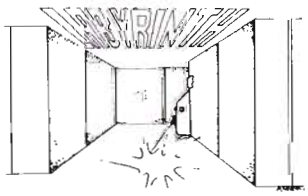
TRS-80 COLOR COMMODORE 24 VIC-20 SINCLAIR/TIMEX TI99

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

Pie Graph

by Eric Giguere
Peace River, Alberta, Canada

The Super Expander cartridge from Commodore offers the VIC owner easy access to some remarkable graphic capabilities. These graphics can be used not only for entertainment, but also for more practical purposes, such as creating graphs. This is the object behind this article—to show you how to use the Super Expander in the creation of graphs. Actually, we're only going to deal with pie graphs, but you can apply some of these methods to other kinds as well.

What is a Pie Graph?

Figure 1 is an example of a pie graph. It is a circle divided up into various "pieces", each representing a specific percentage of the pie as a whole. Direct comparisons are easily made, as the larger the percentage, the bigger the piece. It's a convenient way of showing how resources or money is being used, and how much is being used exactly (of course, it doesn't just apply to money—it can be used for anything that totals up to 100%). Like they say, a picture is worth a thousand words, and that's easy to see when compared with the chart in Figure 2, which lists the same information show in the pie graph. From which one do you absorb the most information? Isn't it easier to make conclusions from the graph than from the chart? That's the reasoning behind the extensive use of charts in business. Hey! There's a real use for these graphs with the Super Expander! It can be used for business purposes! Got your interest now? Good, because it isn't that hard and it **is** useful to be able to use pie graphs, if only to impress your friends!

Creating the Graph

Type in Program I (Pie Graph Demo) and RUN it (you can exclude all REM statements and lines that only have a colon ":" in them). This is a demonstration of what you can achieve with the Super Expander. All the program does is plot a circle, choose a random angle, and then color in the two parts of the pie. It also alternates the pie colors for each drawing. It will continue to run until you hit a key, at which time it will stop and return to the text mode. By now you're probably impressed with what can be done and want to know how I did it, so I won't keep you in suspense any longer. After you read this, you'll probably think how easy it is!

First, we have to set up some variables, as done in line 15. Here we define the function FNR(Z) to equal the formula for generating a number between 0 and Z. We then define all other permanent variables. X and Y are the coordinates for the center of the screen, RX and RY are the circle's radius in X and Y, and C is a constant used in our formulas in lines 70 and 75. Notice that the X radius (RX) is less than the Y radius (RY). This is to account for the fact that the pixels (dots) on the screen are shorter than they are tall, and a circle with RX and RY the same would actually be an oval. Thus both variables are different to correct the problem and make sure the circle is indeed a circle.

Line 20 randomly selects the two colors in which the pie will be filled (but keeps the background and border colors the same), and then draws the circle with the CIRCLE command in the middle of the screen as defined by X and Y. We then proceed to line 25, which draws a straight line from the center of the circle to its right side. T is used by the formulas in the X-Y routine as the degree (0 to 359) that you wish to use. In this case it equals zero (0), and so the line drawn by DRAW represents the start of the circle.

Line 30 gets a random number for the degree, and gosubs 70 to find the proper X-Y location it needs. The DRAW in line 35 then draws a line from the center to the appropriate point on the edge of the circle. This is the division line, and the pie is now separated into two parts. We could stop here, but I chose to add some color to the graph, and so used the PAINT commands in lines 40 and 45. The coding preceding the paints is used to find an offset from the division line so that the PAINT routine will fill in the space between the lines (otherwise it won't do anything). This done, line 50 prints out the angle of the division line at the top corner of the screen and then waits while the VIC counts to a thousand. The keyboard buffer is then checked to see if a key has been pressed, and if not then it returns to draw another graph.

Lines 70 and 75 are the important parts of the program. They use SIN and COS to find the actual point on the edge of the circle that the angle T refers to, and returns these in the variables X% and Y%. This routine was from an article in the March 1983 issue of COMPUTE! magazine, and though I'm not exactly sure how it works, it does the job just fine, even after a bit of careful modification. It sure saved me a lot of trouble!

Pie Graphs and Percentages

Program 2 (Pie Graph 1) is also mainly a demonstration, but it can be modified for more practical purposes. It lets you enter a percentage from 1 to 99 and will then make a graph using that figure. Although the program uses only the integer portion of your figure

so that it can be neatly displayed on the screen, the program can be quite accurate, down to about .5%. About the only thing different in Program 2 as compared to Program 1 is line 125, which calculates the proper angle for your percentage by multyplying it by 360 and dividing it by 100. The rest is pretty well much the same as in Program 1. If you wish to stop the program, simply hit 'Q' after it has finished with the graph.

Conclusion

Many useful adaptations can be made to these programs. You can add a printout module, labels, etc. In any case, these programs are good as demonstrations of the VIC's graphics and the easy use of Super Expander commands (few computers have a PAINT command). I just hope that you enjoy these programs, and if there is enough interest, I might publish a few more programs like these for use with the Super Expander.□

Figure 1: Example Pie Graph showing the imaginary sales of an imaginary product.

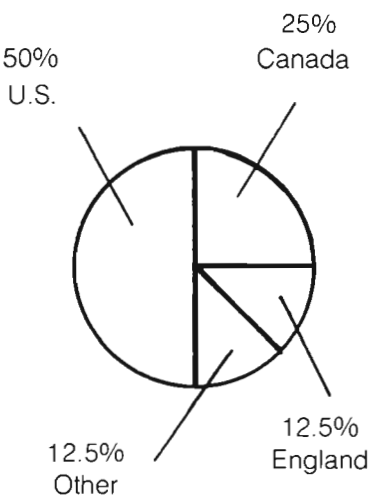


Figure 2: A chart showing the same information in Figure 1.

%	Place
50.0	U.S.A.
25.0	Canada
12.5	England
12.5	Other

Program 1

```
1 REM PIE GRAPH DEMO
2 REM
3 REM (C) 1983 BY
4 REM
5 REM.ERIC GIGUERE
6 REM
7 REM BOX 901, PEACE RIVER
8 REM ALBERTA CANADA
9 REM
10 GRAPHIC1
14 REM SET UP VARIABLES
15 DEFFNR(Z)=INT(RND(1)*Z):X=511:Y=511:RX=300:RY=400:C=3.14/180
17 :
18 REM CHANGE COLORS
19 REM & DRAW CIRCLE
20 COLOR0,1,FNR(16)+1,FNR(16)+1:CIRCLE1,X,Y,RX,RY
23 :
24 REM DRAW LINE AT 0 DEGREES
25 T=0:GOSUB70:DRAW1,X,Y TOX%,Y%
28 :
29 REM GET RANDOM DEGREE
30 T=FNR(350)+5:GOSUB70
33 :
34 REM DRAW DIVISION
35 DRAW1,X,Y TOX%,Y%
37 :
38 :
39 REM FILL IN CIRCLE
40 T=T-5:M=1:GOSUB70:PAINT3,X%,Y%
45 T=T+10:GOSUB70:PAINT2,X%,Y%:M=0
47 :
48 :
49 REM PAINT FIRST AREA
50 T$=STR$(T):CHAR0,0,T$:FORI=1TO1000:NEXT
52 :
53 :
54 REM CHECK TO SEE IF KEY PRESSED
55 GETA$:IFA$<>""THEN:GRAPHIC4:END
60 SCNCLR:GOTO20
66 :
67 :
68 REM CALCULATE X-Y
69 REM POSITION
70 X%=X+(RX-M*100)*COS(T*C)
75 Y%=Y+(RY-M*100)*SIN(T*C)
80 RETURN
```

Program 2

```
10 POKE36879,27:PRINTCHR$(142)"■■■      PIE GRAPH1"
15 PRINT"■■ (C)1983 BY E.GIGUERE":PRINT"■──────────────────────"
20 PRINT"YOU WILL BE ASKED TO  ENTER A PERCENTAGE."
25 PRINT"■A PIE GRAPH WILL THEN BE GENERATED FROM
   THE FIGURE YOU GAVE."
30 PRINT"■YOU MAY PRESS ANY KEY TO START OVER AGAIN
   WHEN THE GRAPH IS FINISHED."
35 PRINT"■■■ HIT ANY KEY TO START "
40 GETA$:IFA$=""THEN40
45 POKE36879,27:PRINT"■■":IFA$="Q"THENEND
50 PRINT"PERCENTAGE TO GRAPH? ";:GOSUB1000
55 P=INT(IN):IFP<1ORP>99THENPRINT"■VALUE OUT OF RANGE.■":GOTO50
100 GRAPHIC1:X=503:Y=X:RX=300:RY=400:C=3.14/180:M=0
110 COLOR0,1,10,15:CIRCLE1,X,Y,RX,RY
120 T=0:GOSUB300:DRAW1,X,YTOX%,Y%
125 T=P*360/100:GOSUB300:DRAW1,X,YTOX%,Y%
130 T=T-5:M=1:GOSUB300:PAINT3,X%,Y%
135 T=T+10:GOSUB300:PAINT2,X%,Y%
140 S$=MID$(STR$(P),2)+" %":REGION1
145 CHAR10,16,S$
195 WAIT197,64,64:GRAPHIC4:GETA$:GOTO45
300 X%=X+(RX-M*100)*COS(T*C)
310 Y%=Y+(RY-M*100)*SIN(T*C)
320 RETURN
1000 OPEN1,0:INPUT#1,IN$:CLOSE1:PRINT:IN=VAL(IN$):RETURN
READY.
```

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™

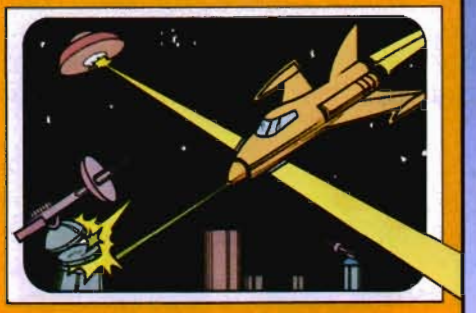
SAVE NEW YORK

Joystick controller required.

CREATIVE SOFTWARE PRESENTS

ASTROBLITZ

A GAME ON CARTRIDGE FOR THE COMMODORE 64™



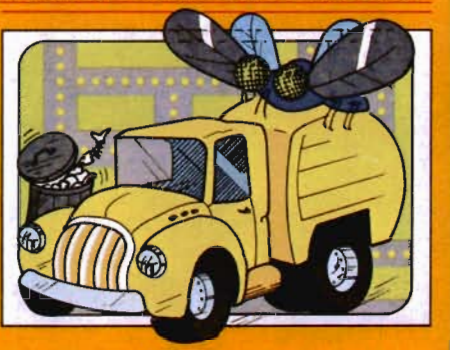
Joystick controller required.

THIS GAME IS ALSO AVAILABLE FOR THE VIC-20.

CREATIVE SOFTWARE PRESENTS

TRASHMAN

A GAME ON CARTRIDGE FOR THE COMMODORE 64™



Joystick controller required.

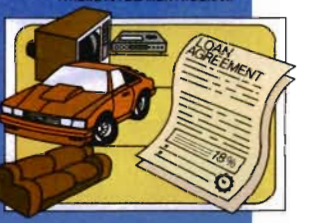
THIS GAME IS ALSO AVAILABLE FOR THE VIC-20.

CAR COSTS

A HOME APPLICATION PROGRAM

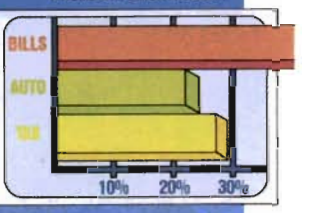
LOAN ANALYZER

A HOME APPLICATION PROGRAM



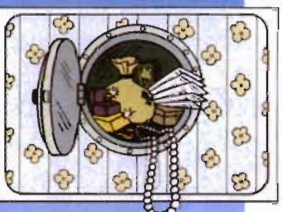
HOUSEHOLD FINANCE

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 ASCII, Inc.
230 East Caribbean Drive
Sunnyvale, CA 94086

*Based on survey of distributors and retailers.

"VIC-20," "COMMODORE" and "COMMODORE 64" are trademarks of COMMODORE ELECTRONICS, LTD.

Circle No 44

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.

Character Editor for Commodore 64

by Garry G. Kiziak
Burlington, Ontario

The characters that are normally PRINTed or POKEd to the screen on the COMMODORE 64 are stored in ROM. There are two different character sets, each occupying 2K (2048) bytes of memory.

The GRAPHICS character set (the usual uppercase characters and the special graphics characters) is enabled with a POKE 53272,21 while the TEXT set (the lowercase characters) is enabled with a POKE 53272,23.

It is possible to have character sets stored in RAM on the 64—actually 8 sets are possible but only 5 are usable. The RAM character sets are numbered 0 to 7. Each set occupies 2K of memory and is enabled by poking location 53272 with an appropriate value. Table 1 contains a summary of the locations used by these characters sets and the values required to enable them. (See Table 1.)

Character set 0 is not usable since this part of memory is used by the operating system and screen memory. Notice that the pokes that enable character sets 2 and 3 in fact enable the ROM character sets (The VIC chip "sees" the ROM sets in these locations even though they are actually stored in high memory). Thus character sets 0, 2, and 3 are not normally usable. I say normally because it is possible to have the VIC chip "look" at a different bank in which they are usable, but for our purposes we shall only consider sets 1, 4, 5, 6, and 7 as being available. Since these sets are located in the same part of memory that a BASIC program normally occupies, it may be necessary to move your BASIC program up in memory whenever you use a RAM character set (see [1] for details on how this is done). If your program is relatively small (6K-12K), relocation may not be necessary. In such cases, character sets 5, 6, or 7 can be used and can even be saved with the BASIC program itself (again, see [1] for details).

The CHARACTER EDITOR presented in this article will allow you to create, edit, etc. characters in any of these five character sets. The character sets that you create can be saved to disk or cassette for later use in your own programs.

Entering the Programs

In order to use the Character Editor, you will have to type in two separate programs, the CHAR BOOT program and the CHAR EDITOR program. The CHAR BOOT program alters the pointers in zero page so that the Character Editor can be loaded above Character Set 7. It also stores a short machine language program and the data for a sprite, used by the editor, into memory. Finally, it loads the Character Editor into memory and runs it. The CHAR BOOT program (listing 1) has been listed showing all the cursor control characters within square brackets to simplify entering the program. Cassette users should leave out the ",8" in line 35 and make certain that the CHAR EDITOR program (listing 2) is saved immediately after the CHAR BOOT program on tape.

How to Use the Editor

Once the Character Editor is up and running, you will see an 8x8 grid in the upper half of the screen, at the left hand side. It is called the EDIT GRID. On the bottom half of the screen, you will see 64 characters enclosed by a border, with the label "CHARACTER SET 2" at the top, and a box cursor flashing over the "@" character.

There are two distinct modes in the editor, Character Selection Mode and Edit Mode.

In Edit Mode, several commands are available that allow you to create and edit the characters in your character set.

In Character Selection Mode, there are commands to help you select the character set you want to work with, the character you want to edit, as well as commands to load and save your completed character sets.

Character Selection Mode

When you first run the editor and see the box cursor flashing over the "@" character, you are in Character Selection Mode. The label "CHARACTER SET 2" reminds you that you are presently viewing the first 64 characters in character set 2, the normal Graphics set. As we mentioned earlier, this set is in ROM and cannot be edited. One of the commands available to you in this mode allows you to change from one character set to another. To do this, press any one of the keys 1, 2, 3, 4, 5, 6, or 7. Pressing any of these keys at this time (except for 2 or 3) will produce garbage on the screen. This occurs simply because you don't have any characters stored in the necessary RAM locations. Press

the 2 or 3 key now to make the screen readable once again.

About the only useful command that you can use at this time involves pressing the COMMODORE key and the "L" key simultaneously. You should see the message "LOAD FROM ROM" displayed at the top of the screen and just below it the prompt "GRAPHICS OR TEXT" with the G of GRAPHICS and T of TEXT highlighted in reverse. Press one of these keys (G or T) now. Immediately the program will ask you "TO WHICH SET:". Answer 1 for the time being (you could choose 1, 4, 5, 6, or 7). Immediately, character set 1 will be displayed and the characters will be loaded from ROM into that character set. When the flashing box cursor returns, you are back in Character Selection Mode and ready to proceed. Here are the commands available to you in Character Selection Mode.

Keystroke / Action

CRSR-RT, -LFT, -UP, -DWN—These keystrokes move the box cursor to any of the 64 characters displayed on the screen. Notice that wraparound occurs to the next line horizontally and the same column vertically.

Return—This selects the character inside the box cursor for editing. The character is displayed on the Edit Grid in an enlarged format, and to the right of the Edit Grid, in normal format, in each of the 16 available colours. At this point you are in Edit Mode and you can proceed to edit the character on the grid.

CTRL-N—This displays the NEXT 64 characters of the present character set at the bottom of the screen. (There are 256 characters in each character set, but only 64 are displayed at any given time. Pressing CTRL-N four times in succession will bring you back to the original 64 characters.

CTRL-P—Displays the PREVIOUS 64 characters of the present character set.

CTRL-B—Allows you to change the BACKGROUND colour of the screen. Actually, like the Sprite Editor in [1] it steps through the 16 available colours.

Continued on page 56

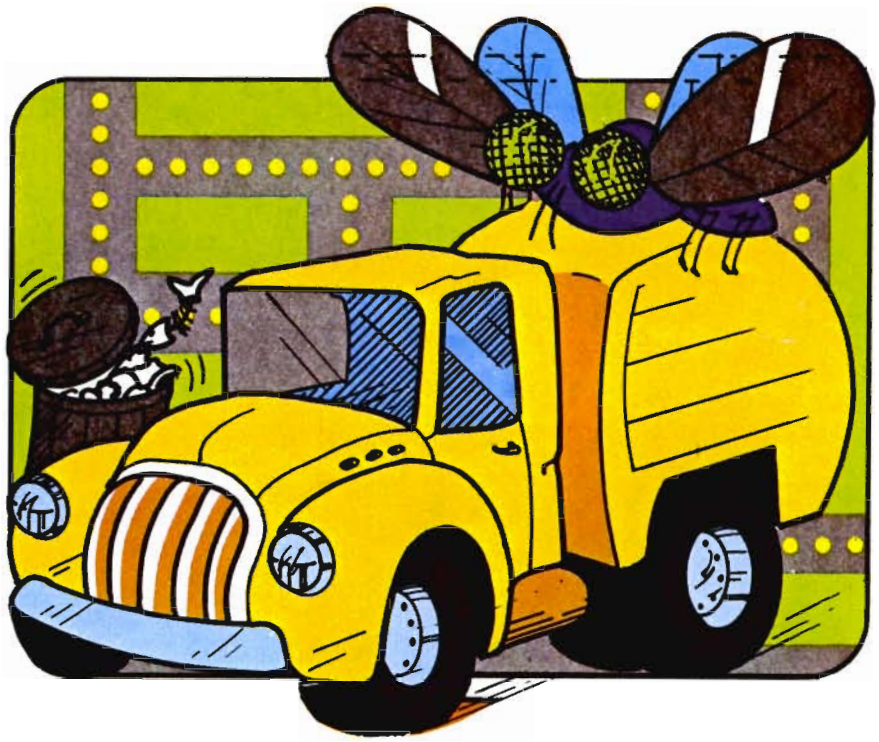
Table 1

Character Set	Location	Enable With
0	0 - 2047	POKE 53272,17
1	2048 - 4095	POKE 53272,19
2	4096 - 6143	POKE 53272,21
3	6144 - 8191	POKE 53272,23
4	8192 - 10239	POKE 53272,25
5	10240 - 12287	POKE 53272,27
6	12288 - 14335	POKE 53272,29
7	14336 - 16383	POKE 53272,31

Listing 1

```
10 REM
15 REM  CHRACTER EDITOR BOOT
20 REM
25 POKE 53280,14:POKE 53281,6
30 PRINT "[CLEAR][WHITE][DOWN][DOWN][DO
WN][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN]
[DOWN][DOWN][DOWN][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][
RIGHT]LOADING ...[DOWN][DOWN]":GOSUB100
35 PRINT "[HOME][BLUE]LOAD"CHR$(34)"CHA
R EDITOR"CHR$(34)",8"
40 FOR I=0 TO 63:READ X:POKE 704+I,X:NEXT
45 FOR I=0 TO 79:READ X:POKE 896+I,X:NEXT
50 FOR I=0 TO 18:READ X:POKE 976+I,X:NEXT
55 FOR I=1 TO 8:READ X:POKE 630+I,X:NEX
T
60 POKE 198,8
65 POKE 44,64:POKE 16*1024,0:CLR:NEW
100 PRINT"   [REV]   [OFF] [REV] [OFF]
[REV] [OFF] [REV]   [OFF] [REV]   [OFF]
 [REV]   [OFF] [REV]   [OFF] [REV]   [O
FF] [REV]   [OFF] [REV]   [OFF]
110 PRINT"   [REV] [OFF]   [REV] [OFF]
[REV] [OFF] [REV] [OFF] [REV] [OFF] [RE
V] [OFF] [REV] [OFF] [REV] [OFF] [REV]
[OFF] [REV] [OFF]   [REV] [OFF]  [REV]
 [OFF]   [REV] [OFF] [REV] [OFF]
120 PRINT"   [REV] [OFF]   [REV] [OFF]¯
[REV] [OFF] [REV] [OFF]¯[REV] [OFF] [RE
V] [OFF]¯ [REV] [OFF]¯[REV] [OFF] [REV
] [OFF]   [REV] [OFF]  [REV] [OFF]¯  [
REV] [OFF]¯
130 PRINT"   ——— - - - - - - ▪ - - ——  -
 ——— - ▪
140 PRINT"          [REV]   [OFF] [REV]
   [OFF] [REV] [OFF] [REV]   [OFF] [REV
]   [OFF] [REV]   [OFF]
150 PRINT"          [REV] [OFF]   [REV]
 [OFF] [REV] [OFF] [REV] [OFF]  [REV] [
OFF]  [REV] [OFF] [REV] [OFF] [REV] [OF
F] [REV] [OFF]
```

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, a Creative Software package front and proof of purchase slip.

Entries must be mailed to Commander, Creative Software 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 Creative Software and their families may not participate. First prize will be $250, second prize $100, third prize $50.

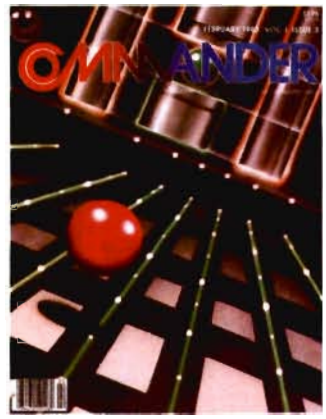
The contest will run until a first prize is awarded. Creative Software will notify Commander Magazine of the winner(s).

The Game Contest is a continuing feature of Commander Magazine aimed at providing entertainment for and promoting competition among our readers. Creative Software has graciously provided us with this Game Contest.

DEADLINE FOR ENTRIES—
AUGUST 1, 1983

Trashman may be purchased from Creative Software or any one of its fine dealers:
230 East Caribbean Drive
Sunnyvale, CA
1-408-745-1655

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) 565-6818

☐ **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

CTRL-E—Steps through the 16 EDGE (Border) colours.

Any of the # keys 1 to 7—Displays the corresponding character set for editing. Remember that character sets 2 and 3 can be displayed but cannot be edited.

CTRL-L—Loads a character set from disk into memory. You are prompted for the FILENAME of the set. If it is found, it is loaded into the same set (1, 4, 5, 6, or 7) that it was orginally saved to.

SHIFT-L—LOADS a character set from cassette into memory.

C=-L—LOADS the ROM character set that you specify (GRAPHICS or TEXT) into the character set that you specify.

CTRL-S—SAVES the character set, presently being displayed (edited), to disk. You are prompted for a FILENAME and the number of the set you wish to save that set to. Notice that even if you are presently editing character set 1, you can save it to character set 4, 5, 6, or 7 if you wish.

SHIFT-S—SAVES the character set presently being edited to cassette.

CTRL-K—Allows you to select a character for editing by pressing the corresponding key (rather than by moving the box cursor to it and pressing RETURN). As soon as CTRL-K is pressed the message "WHICH KEY" is displayed on the screen. Press RETURN and you are back in Character Selection Mode. (Do this if you pressed CTRL-K by accident or if you have changed your mind.) Pressing any other key on the keyboard will display that character on the Edit Grid and put you in Edit Mode. Shifted characters and characters obtained by first pressing the COMMODORE key can be selected by this means—control characters, of course, cannot. Even reverse video characters can be selected by preceding the key of your choice with the RVS (actually CTRL-9) key.

I found this method of selecting a character useful when I attempted to create a wide (i.e. double width) character set. Each character in this set is actually two characters. I found it convenient to store the left half of a

```
160 PRINT"           [REV] [OFF]¯  [REV]
 [OFF] [REV] [OFF] [REV] [OFF]  [REV] [
OFF]  [REV] [OFF] [REV] [OFF] [REV] [OF
F]¯▌
170 PRINT"                 ▀▀▀▀ ▀▀▀▀ ▀  ▀  ▀▀▀▀ ▀
 ▪[HOME]
180 RETURN
1000 DATA 255,255,0,255,255,0,192,3,0,1
92,3,0,192,3,0,192
1010 DATA 3,0,192,3,0,192,3,0,192,3,0,1
92,3,0,192,3
1020 DATA 0,192,3,0,192,3,0,255,255,0,2
55,255,0,0,0,0
1030 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0
1040 DATA 160,7,177,251,162,0,42,62,157
,3,232,224,8,208,247,136
1050 DATA 16,240,160,7,185,157,3,145,25
1,136,16,248,96,234,234,234
1060 DATA 234,234,234,234,234,169,148,1
33,251,169,5,133,252,162,7,160
1070 DATA 7,169,46,126,157,3,144,2,169,
81,145,251,136,16,242,56
1080 DATA 165,251,233,40,133,251,165,25
2,233,0,133,252,202,16,224,96
1090 DATA 162,8,160,0,177,251,145,253,2
00,208,249,230,252,230,254,202,208,242,
96
2000 DATA 19,13,32,32,82,85,78,13
READY.
```

Listing 2

```
100 PT=160:POKE 953,PT:X1=52:Y1=166:IF
V=0 THEN GOSUB 1120
110 PRINT LEFT$(V$,5)TAB(18)LEFT$(BL$,9
)
120 PRINT:PRINT TAB(18)LEFT$(BL$,16)
130 GOSUB 610:GOSUB 1160:GOSUB 620:GOSU
B 860:GOSUB 650
140 FOR I=0 TO 7:P=PEEK(L+I):POKE Z+I,P
:NEXT
150 GOSUB 570:X=0:Y=0:GOSUB 590
160 P=1148+X+Y*40:Q=PEEK(P):R=Q
170 R=(NOTRAND128)OR(NOT128ANDR)
180 POKE P,R
190 FOR I=1 TO 30:GET A$:IF A$="" THEN
NEXT:GOTO 170
200 POKE P,Q
210 IF A$="[RIGHT]" THEN X=(X+1)AND7:GO
TO 160
220 IF A$="[LEFT]" THEN X=(X-1)AND7:GOT
O 160
230 IF A$="[DOWN]" THEN Y=(Y+1)AND7:GOT
O 160
```

character in its normal location (e.g. the A key) and the right half of the character in the shifted character (i.e. the SHIFT-A key). Trying to select a key by moving the box cursor to it each time was very time consuming and error prone, especially when some characters are so similar in appearance (e.g. SHIFT-H and SHIFT-Y).

Q—QUITS the editor. Once you have quit the editor, you may restart it, with all the characters in tact, simply by typing RUN. The CHAR BOOT program does not have to be reloaded and run. Note: Some of these commands do not work when viewing character set 2 or 3 since these sets cannot be edited.

Edit Mode—Once the character you have selected is displayed in the EDIT GRID, there are a number of commands available that simplify the creation or editing of this character. If you typed in the SPRITE EDITOR in [1] then you will find that the following commands work in exactly the same way.

CLR—Clears the edit grid (and the character being edited).

HOME—Places the cursor in the upper left corner (the home position) without affectng anything else on the screen.

CRSR-RT,-LFT, -UP, -DWN—Used to move the flashing cursor around the edit grid without affecting anything else on the screen.

.—Turns the block under the flashing cursor (and the corresponding pixel of the character) on.

SPACE—Turns the block under the flashing cursor off.

DEL—Turns the block to the left of the cursor off (taking into account wraparound).

RETURN—Moves the cursor to the beginning of the next line.

F1,F3,F5,F7—Moves the entire grid up, down, left, or right one line or column.

CTRL-R—REVERSES the entire grid (and character).

CTRL-B—Changes the BACKGROUND colour of the screen.

CTRL-E—Changes the EDGE

```
240 IF A$="[UP]" THEN Y=(Y-1)AND7:GOTO 160
250 IF A$=DE$ THEN X=(X-1)AND7:P=1148+X
+Y*40:POKE P,46:T=Z+Y
260 IF A$=DE$ THEN POKE T,PEEK(T)ANDH(X
):GOTO 160
270 IF A$=" " THEN POKE P,46:T=Z+Y:POKE
 T,PEEK(T)ANDH(X):X=(X+1)AND7:GOTO 160
280 IF A$="." THEN POKE P,PT:T=Z+Y:POKE
 T,PEEK(T)ORM(X):X=(X+1)AND7:GOTO 160
290 IF A$="[CLEAR]" THEN GOSUB 620:GOSU
B 640:X=0:Y=0:GOTO 160
300 IF A$="Q" THEN GOSUB 620:GOSUB 640:
DC=32:GOSUB 590:DC=255:GOSUB 650:GOTO 140
310 IF A$="[HOME]" THEN X=0:Y=0:GOTO 16
0
320 IF A$=CR$ THEN X=0:Y=(Y+1)AND7:GOTO
 160
330 IF A$="■" THEN FOR I=1 TO 7:POKE Z+
I-1,PEEK(Z+I):NEXT:POKE Z+7,0:GOSUB 570
340 IF A$="■" THEN 160
350 IF A$="■" THEN FOR I=7 TO 1 STEP -1
:POKE Z+I,PEEK(Z+I-1):NEXT:POKE Z,0
360 IF A$="■" THEN GOSUB 570:GOTO 160
370 IF A$="■" THEN FOR I=0 TO 7:POKE Z+
I,(PEEK(Z+I)*2)AND255:NEXT
380 IF A$="■" THEN GOSUB 570:GOTO 160
390 IF A$="■" THEN FOR I=0 TO 7:POKE Z+
I,(PEEK(Z+I)/2):NEXT
400 IF A$="■" THEN GOSUB 570:GOTO 160
410 IF A$="[REV]" THEN FOR I=0 TO 7:R=P
EEK(Z+I):R=(NOTRAND255)OR(NOT255ANDR)
420 IF A$="[REV]" THEN POKE Z+I,R:NEXT:
GOSUB 570:GOTO 160
430 IF A$="■" THEN BG=(BG+1)AND15:POKE
53281,BG:GOTO 160
440 IF A$="[WHITE]" THEN BR=(BR+1)AND15
:POKE 53280,BR:GOTO 160
442 IF A$="■"ANDY<7 THEN FOR I=7 TO Y+1
 STEP-1:POKEZ+I,PEEK(Z+I-1):NEXT:POKEZ+
Y,0
444 IF A$="■"ANDY<7 THEN GOSUB 570:GOTO
 160
446 IF A$="■"ANDY<7 THEN FOR I=Y TO 6:P
OKEZ+I,PEEK(Z+I+1):NEXT:POKEZ+7,0
448 IF A$="■"ANDY<7 THEN GOSUB 570:GOTO
 160
449 IF (A$="■"ORA$="■")ANDY=7 THEN POKE
 Z+Y,0:GOSUB 570:GOTO 160
450 IF A$="£" THEN GOSUB 580:GOSUB 570:
GOTO 160
455 IF A$="■" THEN FOR I=0 TO 3:Z2=PEEK
(Z+I):POKEZ+I,PEEK(Z+7-I):POKEZ+7-I,Z2:
NEXT
456 IF A$="■" THEN GOSUB 570:GOTO 160
460 IF A$<>"■" AND A$<>"■" THEN 160
```

(Border) colour of the screen.

CTRL-F—FLIPS the character upside down.

£—(The pound key) ROTATES the character 90 degrees.

CTRL-I—INSERTS a blank line at the current cursor position, moving everything on or below this line down.

Additonal Commands

CTRL-D—DELETES the line at the current cursor position, moving everything below this line up and blanking out the bottom line (like CTRL-K in the SPRITE EDITOR).

CTRL-A—Allows you to ASSIGN the character being edited to some character in the character set. When CTRL-A is pressed, the box cursor will flash over the character originally selected. The edited character can be assigned to this character or to another character by first using the cursor keys and/or CTRL-N or CTRL-P to position the box cursor over the desired character and then pressing RETURN This approach is most useful when creating a character set the first time. As you create a character, you assign it to its original position and then move on to the adjacent character.

CTRL-K—Allows you to assign the character being edited to some character by pressing the desired key. This command works in exactly the same way as in Character Selection Mode. (See the comments there to see when this method might be more useful than CTRL-A.)

Q—Pressing Q in Edit Mode allows you to reenter Character Selection Mode without assigning the edited character to any character in the character set. The edited character is, however, lost.

Some Comments

1) A command to flip a character sideways was not included since this can be achieved quickly by first flipping it upside down and then rotating it 180 degrees (i.e. press CTRL-F, then the "£" key twice).

2) The changes made to any character being edited are always

```
464 REM ******************************
465 REM *                            *
466 REM *  ASSIGN CHARACTER BEING    *
467 REM *     EDITED TO A KEY        *
468 REM *                            *
469 REM ******************************
470 PRINT "[HOME][DOWN][DOWN][DOWN][DOW
N][DOWN][DOWN][DOWN][RIGHT][RIGHT][RIGH
T][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][R
IGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT]   ASSIGN
TO    "
480 PRINT "[RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT]      WHICH      "
490 IF A$="■" THEN PRINT "[RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT]    CHARACT
ER    ":FK=2:GOSUB 660:FK=0
500 IF A$="■" THEN PRINT "[RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT]       KEY
     ":GOTO 1400
510 FOR I=0 TO 7:POKE L+I,PEEK(Z+I):NEX
T:GOSUB 540
520 IF L<>Z THEN GOSUB 640
530 GOSUB 650:GOTO 140
540 PRINT "[HOME][DOWN][DOWN][DOWN][DOW
N][DOWN][DOWN][DOWN]";:FOR I=0 TO 2:PRI
NT TAB(17)"                ":NEXT:RETUR
N
565 REM ******************************
566 REM *                            *
567 REM *  PRINT CHARACTER ON GRID   *
568 REM *                            *
569 REM ******************************
570 FOR I=0 TO 7:POKE 925+I,PEEK(Z+I):N
EXT:SYS 933:RETURN
575 REM ******************************
576 REM *                            *
577 REM *   ROTATE EDIT CHARACTER    *
578 REM *                            *
579 REM ******************************
580 HI=INT(Z/256):LO=Z-256*HI:POKE 251,
LO:POKE 252,HI:SYS 896:RETURN
584 REM ******************************
585 REM *                            *
586 REM *   PRINT EDIT CHARACTER     *
587 REM *     IN ALL 16 COLORS       *
588 REM *                            *
589 REM ******************************
590 J=0:FOR I=0 TO 15:K=IAND7:IF K=0 TH
EN J=J+80
```

made in a buffer, not to the character itself. (Notice you don't see the character that you are editing changing on the bottom half of the screen, only the characters to the right of the Edit Grid.) This buffer is actually the last character of the character set. This means that if you want to have a character in this position in your final character set, then you must make certain that it is the last character you edit before saving that set to disk or tape.

3) The Save routine (either to disk or tape) saves the entire character set (2048 bytes). In some cases, you might want to save just part of the character set. I did not include this feature in the program simply because I didn't find a need for it. If you do, you can add this feature quite easily, or simply quit the editor (by pressing Q) and make appropriate changes to line 1080 or line 1880. Then rerun the program and save the partial character set. When working with disk, this will cause no problem at all, however, when working with cassette, if you wish to reload the partial character set for further editing, you will also have to change line 2105 in the load routine accordingly.

4) For cassette users, it would be totally impractical to have a complete character set stored in DATA statements so you have to either use the DATA file approach or save the character set as part of your BASIC program. Disk users can load a character set just like a Sprite Table. (See [1] for details.)

5) In some circumstances, it might be practical to store a partial character set in DATA statements. Therefore listing 3, is a short program (CHARS TO DATA) that will perform the necessary conversion (it will even do an entire character set if you wish). With the character set in memory (e.g. after pressing Q in the Character Editor), load and run this program. You will be asked for the starting line number for the DATA statements, the increment, the number of the character set (1,4,5,6,7), the number of the character in that set at which to begin (0 to 255), and the number of characters to convert. After the conversion

```
600 POKE CL+2*K+J+54272,I:POKE CL+2*K+J
,DC:NEXT:RETURN
605 REM ******************************
606 REM *                            *
607 REM *   CHANGE CHARACTER SET     *
608 REM *                            *
609 REM ******************************
610 CB=CS*2048:Z=CB+255*8:POKE V+24,17+
2*CS:RETURN
615 REM ******************************
616 REM *                            *
617 REM *        PRINT GRID          *
618 REM *                            *
619 REM ******************************
620 PRINT "[HOME][DOWN][DOWN][RIGHT][RI
GHT][RIGHT][REV] ■■■■■■■■ [OFF]":FOR I=
1 TO 8:PRINT "[RIGHT][RIGHT][RIGHT]▮...
.....[REV]▮":NEXT
630 PRINT "[RIGHT][RIGHT][RIGHT][REV] "[
OFF]■■■■■■■■[REV]" ":RETURN
635 REM ******************************
636 REM *                            *
637 REM *   CLEAR EDIT CHARACTER     *
638 REM *                            *
639 REM ******************************
640 FOR I=0 TO 7:POKE Z+I,0:NEXT:RETURN
644 REM ******************************
645 REM *                            *
646 REM *    SELECT A CHARACTER      *
647 REM *        FOR EDITING         *
648 REM *                            *
649 REM ******************************
650 POKE V+21,0:POKE V+16,FL:POKE V,X1:
POKE V+1,Y1
660 R=0
670 R=1-R:POKE V+21,R
680 FOR I=1 TO 20:GET A$:IF A$="" THEN
NEXT:GOTO 670
690 POKEV+21,1:A=VAL(A$):IFA>0ANDA<8AND
FK=0THENCS=A:GOSUB610:POKE1570,CS+176:G
OTO670
700 IF A$="■" AND FK=0 THEN BG=(BG+1)AN
D15:POKE 53281,BG:GOTO 670
710 IF A$="[WHITE]" AND FK=0 THEN BR=(B
R+1)AND15:POKE 53280,BR:GOTO 670
720 IF A$="■" THEN TT=(TT+1)AND3:GOSUB
860:GOTO 670
725 IF A$="■" THEN TT=(TT-1)AND3:GOSUB
860:GOTO 670
730 IF A$="Q" AND FK=0 THEN POKE V+21,0
:PRINT"[CLEAR]";:END
735 IF A$=" " AND FK=0 THEN POKE V+21,0
:GOSUB 1600:GOTO 670
740 IF A$="■" AND FK=0 THEN POKE V+21,0
:GOSUB 880:GOTO 670
```

is complete, delete lines 0-12 and save the resulting DATA statements. You can then append these DATA statements to any BASIC program in which you wish to use those characters (see below).

Appending a BASIC Program

Appending one BASIC program to the end of another can be done quite easily on the 64. Here are the steps required.

1) Make certain that the program to be appended has line numbers greater than the largest line number in the other BASIC programs. (The CHARS TO DATA program allows you to choose the starting line number for the DATA statements so this should be no problem.)

2) Turn the computer off and then on again to make certain that certain pointers have not been altered by the previous program that you used. Then load the original program into memory.

3) In direct mode, type
PRINT PEEK(45)
If this number is 0 or 1, then add a dummy line (such as 0 REM) to your program and repeat step 3.

4) Again in direct mode, type
POKE 43,PEEK(45)-2:POKE 44,
PEEK(46):CLR

5) If you try to list your programs, nothing will list, but it is still there—hidden. Now load in the program to be appended. After loading, you can list that program but the original is still not there.

6) Now type in
POKE 43,1:POKE 44,8:CLR
The original program should now list along with the appended program. If you had to type in a dummy line in step 3., delete it now and save the resulting program to disk or tape.

References

[1]A SPRITE EDITOR FOR THE COMMODORE 64—by Garry G. Kiziak, Volume 1, Issue 3 (February 1983) of the COMMANDER.

```
742 IF A$="L" AND FK=0 THEN POKE V+21,0
:GOSUB 1980:GOTO 670
745 IF CS=2 OR CS=3 THEN 670
750 IF A$="[RIGHT]" THEN Z1=(Z1+1)AND63
760 IF A$="[LEFT]" THEN Z1=(Z1-1)AND63
770 IF A$="[UP]" THEN Z1=(Z1-16)AND63
780 IF A$="[DOWN]" THEN Z1=(Z1+16)AND63
785 IF A$="■" AND FK=0 THEN FK=1:GOTO 1
400
790 IFA$=CR$THENGOSUB540:GOSUB620:DC=32
:GOSUB590:DC=255:P=Z1+TT*64:L=CB+P*8:RETURN
810 IF A$="[HOME]" AND FK=0 THEN POKE V
+21,0:GOSUB 970:GOTO 670
815 IF A$="●" AND FK=0 THEN POKE V+21,0
:GOSUB 1770:GOTO 670
820 Y1=INT(Z1/16):X1=Z1-Y1*16:Y1=166+16
*Y1:POKE V+21,0
830 IF X1<13 THEN X1=52+X1*16:FL=0:GOTO 850
840 IF X1>12 THEN X1=-12+(X1-12)*16:FL=
1
850 POKE V+16,FL:POKE V,X1:POKE V+1,Y1:
POKE V+21,1:GOTO 670
860 J=1548:FORI=0TO63:K=IAND15:IF K=0 T
HEN J=J+80
870 POKE 2*K+J,I+TT*64:NEXT:RETURN
874 REM ********************************
875 REM *                              *
876 REM *   LOAD A CHARACTER SET       *
877 REM *        FROM DISK             *
878 REM *                              *
879 REM ********************************
880 PRINT "[HOME][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGH
T][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][R
IGHT][RIGHT][RIGHT][RIGHT]_____________
__"
890 PRINT "[RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT][RIGHT][REV]     LOAD
 [OFF]"
900 PRINT "[RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT][RIGHT][REV]   FROM DISK
 [OFF]"
910 LI=7:COL=18:LE=16:PRINT LEFT$(V$,5)
TAB(18)"FILENAME:"
920 MSG$="[OFF][OFF]":GOSUB 1230:FL$="0
:"+IN$
930 IF IN$="" THEN PRINT LEFT$(V$,5)TAB
(18)"          ":GOSUB 1160:RETURN
940 OPEN 1,8,15,"I0"
950 OPEN 2,8,0,FL$:GOSUB 1340:CLOSE2:CL
OSE1
960 LOAD FL$,8,1
```

```
964 REM ******************************
965 REM *                            *
966 REM *   SAVE A CHARACTER SET     *
967 REM *          TO DISK           *
968 REM *                            *
969 REM ******************************
970 PRINT "[HOME][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGH
T][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][R
IGHT][RIGHT][RIGHT][RIGHT]_______________"
980 PRINT "[RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT][RIGHT][REV]     SAVE
 [OFF]"
990 PRINT "[RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT][RIGHT][REV]    TO DISK
 [OFF]"
1000 LI=7:COL=18:LE=16:PRINT LEFT$(V$,5
)TAB(18)"FILENAME:"
1010 MSG$="[OFF][OFF]":GOSUB 1230:FL$="
0:"+IN$+",PRG,WRITE"
1020 IF IN$="" THEN PRINT LEFT$(V$,5)TA
B(18)"          ":GOSUB 1160:RETURN
1030 LI=9:COL=19:LE=1:MSG$="[OFF]SAVE T
O SET?[OFF]":GOSUB 1230:TS=VAL(IN$)
1040 IF TS<1 OR TS=2 OR TS=3 OR TS>7 TH
EN 1030
1050 OPEN 1,8,15,"I0":GOSUB 1340
1060 OPEN 2,8,1,FL$:GOSUB 1340
1070 PRINT#2,CHR$(0);CHR$(8*TS);
1080 FORI=0TO7:K=CB+256*I:FORJ=0TO255:P
RINT#2,CHR$(PEEK(K+J));:NEXT:GOSUB1340:
NEXT:CLOSE2
1090 CLOSE1:PRINT LEFT$(V$,5)TAB(18)LEF
T$(BL$,9):PRINT:PRINT TAB(18)LEFT$(BL$,
16)
1100 PRINT LEFT$(V$,9)TAB(18)LEFT$(BL$,
18):GOSUB 1160
1110 RETURN
1114 REM ******************************
1115 REM *                            *
1116 REM *   INITIALIZE VARIABLES     *
1117 REM *     AND ENABLE CURSOR      *
1118 REM *                            *
1119 REM ******************************
1120 PRINT "[CLEAR][WHITE]":V=53248:CL=
1024+5*40+18:DC=255:CS=2
1130 V$="[HOME][DOWN][DOWN][DOWN][DOWN]
[DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][DO
WN][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN]
[DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][DO
WN]":BL$="
     "
```

```
1140 POKE 2040,11:POKE 53271,0:POKE 532
77,0:DE$=CHR$(20):CR$=CHR$(13)
1150 FOR I=0 TO 7:M(I)=2↑(7-I):H(I)=255
-M(I):NEXT:RETURN
1155 REM ******************************
1156 REM *                            *
1157 REM *      SCREEN DISPLAY        *
1158 REM *                            *
1159 REM ******************************
1160 PRINT "[HOME][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT]  ___________________  "
1170 PRINT "[RIGHT][RIGHT][RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT]  [REV]   CHARACTER   [OFF] "
1180 PRINT "[RIGHT][RIGHT][RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT]  [REV]     EDITOR    [OFF] "
1190 PRINT LEFT$(V$,13)TAB(11)"_________
_________"
1200 PRINT "[RIGHT][RIGHT] ▬▬▬▬▬[REV
] CHARACTER SET"CS"[LEFT] [OFF]▬▬▬▬▬▬
▬"
1210 FOR I=1 TO 9:PRINT "[RIGHT][RIGHT]
[REV]▮[RIGHT][RIGHT][RIGHT][RIGHT][RIGH
T][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][R
IGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT]▮":NEXT
1220 PRINT "[RIGHT][RIGHT]"[REV]▬▬▬▬▬
▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬[HOME]":RETU
RN
1225 REM ******************************
1226 REM *                            *
1227 REM *       INPUT ROUTINE        *
1228 REM *                            *
1229 REM ******************************
1230 Y9=2:IN$="":UC=0:UB$=LEFT$(BL$,LE)
:GOSUB 1330:UB$=" ":UC=3
1240 UT=TI
1250 GET Z9$:IF Z9$="" THEN 1310
1260 IF Z9$=CR$ THEN Y9=2:GOSUB 1330:PR
INT "[LEFT][LEFT] ":RETURN
1270 IF Z9$=DE$ THEN ON -(LEN(IN$)=0) G
OTO 1310:IN$=LEFT$(IN$,LEN(IN$)-1):GOTO
1310
1280 IF (ASC(Z9$)AND127)<32 OR Z9$=CHR$
(34) THEN 1310
1290 IF LE=LEN(IN$) THEN 1310
1300 IN$=IN$+Z9$
1310 GOSUB 1330:IF TI-UT<10 THEN 1250
```

COMPU $ENSE

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

```
1320 Y9=3-Y9:GOTO 1240
1330 PRINT LEFT$(V$,LI)TAB(COL-1)MID$(M
SG$,Y9)UB$IN$MID$("[REV] [OFF]",Y9,UC)"
 ";:RETURN
1335 REM ******************************
1336 REM *                            *
1337 REM *   CHECK FOR DISK ERRORS    *
1338 REM *                            *
1339 REM ******************************
1340 INPUT#1,A$,B$,C$,D$
1350 IF VAL(A$)=0 THEN RETURN
1360 PRINT"[CLEAR][DOWN][DOWN][RIGHT][R
EV]DISK ERROR:[OFF] "B$
1370 CLOSE2
1380 END
1395 REM ******************************
1396 REM *                            *
1397 REM * CONVERT ASCII TO SCREEN    *
1398 REM *                            *
1399 REM ******************************
1400 RV=0:POKE V+21,0:IF FK=1 THEN PRIN
T"[HOME][DOWN][DOWN][DOWN][DOWN][DOWN][
DOWN][DOWN][DOWN]"TAB(21)"WHICH KEY"
1410 GET A$:IF A$="" THEN 1410
1412 IF A$=CR$ANDFK=1 THEN GOSUB 540:FK
=0:GOTO 670
1413 IF A$="[REV]" THEN RV=1:GOTO 1410
1414 IF A$="[OFF]" THEN RV=0:GOTO 1410
1415 IF A$=CR$ THEN GOSUB 540:GOSUB 590
:POKE V+21,1:GOTO 160
1420 A=ASC(A$):IF A<32 OR (A>127ANDA<16
0) THEN 1410
1430 IF A<64 THEN 1500
1440 IF A<96 THEN A=A-64:GOTO 1500
1450 IF A<128 THEN A=A-32:GOTO 1500
1460 IF A<192 THEN A=A-64:GOTO 1500
1470 IF A<224 THEN A=A-128:GOTO 1500
1480 IF A<255 THEN A=A-64:GOTO 1500
1490 A=94
1500 IF RV=1 THEN A=A+128
1505 GOSUB 540:GOSUB 620:SS=TT:TT=INT(A
/64):Z1=A-64*TT:IF TT<>SS THEN GOSUB 860
1510 Y1=INT(Z1/16):X1=Z1-Y1*16:Y1=166+1 6*Y1
1520 IF X1<13 THEN X1=52+X1*16:FL=0:GOTO 1540
1530 IF X1>12 THEN X1=-12+(X1-12)*16:FL
=1
1540 POKEV+16,FL:POKEV,X1:POKEV+1,Y1:PO
KEV+21,1:IF FK=1 THEN FK=0:A$=CR$:GOTO
790
1550 DC=32:GOSUB 590:DC=255:P=Z1+TT*64:
L=CB+P*8:GOTO 510
1595 REM ******************************
1596 REM *                            *
1597 REM * LOAD A ROM CHARACTER SET   *
1598 REM *                            *
1599 REM ******************************
```

```
1600 PRINT "[HOME][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT][RIGHT][RIGHT]_______________""
1610 PRINT "[RIGHT][RIGHT][RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][REV]     LOAD
  [OFF]"
1620 PRINT "[RIGHT][RIGHT][RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][REV]   FROM ROM
  [OFF]"
1630 PRINT"[HOME][DOWN][DOWN][DOWN][DOW
N]"TAB(18)"_          _"
1640 PRINTTAB(18)"[REV]G[OFF]RAPHICS-[R
EV]T[OFF]EXT:";:POKE204,0
1650 GETA$:IFA$<>"G"ANDA$<>"T"ANDA$<>CR
$THEN1650
1652 POKE204,1:PRINT"[HOME][DOWN][DOWN]
[DOWN][DOWN]"TAB(17)"                ":P
RINTTAB(17)"                  "
1653 IF A$=CR$ THEN1730
1654 LI=6:COL=19:LE=1:MSG$="[OFF]TO WHI
CH SET:[OFF]":GOSUB1230:SE=VAL(IN$)
1655 IFSE<>1AND(SE<40RSE>7)THEN1654
1656 CS=SE:POKE1570,CS+176
1660 PRINT"[HOME][DOWN][DOWN][DOWN][DOW
N][DOWN]"TAB(18)"                    "
1680 SR=53248:IFA$="T"THEN SR=55296
1687 HI=SR/256:LO=SR-256*HI
1688 POKE 251,LO:POKE 252,HI:POKE 253,0
:POKE 254,CS*8
1690 POKE56334,PEEK(56334)AND254:POKE1,
PEEK(1)AND251
1700 SYS 976
1710 POKE1,PEEK(1)OR4:POKE56334,PEEK(56
334)OR1
1720 GOSUB 610
1730 GOSUB 1160:RETURN
1764 REM ******************************
1765 REM *                            *
1766 REM *   SAVE A CHARACTER SET     *
1767 REM *        TO CASSETTE         *
1768 REM *                            *
1769 REM ******************************
1770 PRINT "[HOME][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT][RIGHT][RIGHT]_______________"
1780 PRINT "[RIGHT][RIGHT][RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][REV]     SAVE
  [OFF]"
```

```
1790 PRINT "[RIGHT][RIGHT][RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][REV]  TO CASSETTE
  [OFF]"
1800 LI=7:COL=18:LE=14:PRINT LEFT$(V$,5
)TAB(18)"FILENAME:"
1810 MSG$="[OFF][OFF]":GOSUB 1230:FL$=IN$
1820 IF IN$="" THEN PRINT LEFT$(V$,5)TA
B(18)"          ":GOSUB 1160:RETURN
1830 LI=9:COL=19:LE=1:MSG$="[OFF]SAVE T
O SET?[OFF]":GOSUB 1230:TS=VAL(IN$)
1840 IF TS<1 OR TS=2 OR TS=3 OR TS>7 TH
EN 1830
1845 PRINT "[HOME][DOWN][DOWN][DOWN][DO
WN]"TAB(18)"PRESS:   [DOWN]":PRINT TAB(
18)"RECORD & PLAY     [DOWN]"
1847 PRINT TAB(18)"THEN PRESS RETURN"
1849 GET A$:IF A$<> CHR$(13) THEN 1849
1851 PRINT "[HOME][DOWN][DOWN][DOWN][DO
WN]"TAB(18)"    SAVING   [DOWN]"
1852 KL=INT((15-LEN(FL$))/2)
1853 PRINTTAB(18)RIGHT$(BL$,KL)FL$LEFT$
(BL$,20-KL-LEN(FL$))
1855 PRINT TAB(18)"[DOWN]
   "
1860 OPEN 2,1,1,FL$
1870 PRINT#2,CHR$(TS);
1880 FOR I=0 TO 2047:PRINT#2,CHR$(PEEK(
CB+I));:NEXT:CLOSE2
1890 PRINT LEFT$(V$,5)TAB(18)LEFT$(BL$,
15):PRINT:PRINT TAB(18)LEFT$(BL$,16)
1900 PRINT LEFT$(V$,9)TAB(18)LEFT$(BL$,
18):GOSUB 1160:GOSUB 2500
1910 RETURN
1974 REM ******************************
1975 REM *                            *
1976 REM *   LOAD A CHARACTER SET     *
1977 REM *       FROM CASSETTE        *
1978 REM *                            *
1979 REM ******************************
1980 PRINT "[HOME][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT][RIGHT][RIGHT]_____________
   ___"
1990 PRINT "[RIGHT][RIGHT][RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][REV]     LOAD
  [OFF]"
2000 PRINT "[RIGHT][RIGHT][RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][REV] FROM CASSETT
E [OFF]"
```

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

Circle No. 59

CM6-A

```
2010 LI=7:COL=18:LE=14:PRINT LEFT$(V$,5
)TAB(18)"FILENAME:"
2020 MSG$="[OFF][OFF]":GOSUB 1230:FL$=IN$
2030 IF IN$="" THEN PRINT LEFT$(V$,5)TA
B(18)"          ":GOSUB 1160:RETURN
2040 PRINT "[HOME][DOWN][DOWN][DOWN][DO
WN]"TAB(18)"INSERT DATA TAPE[DOWN]":PRI
NT TAB(18)"PRESS PLAY ON TAPE[DOWN]"
2050 PRINT TAB(18)"THEN PRESS RETURN"
2060 GET A$:IF A$<>CHR$(13) THEN2060
2070 PRINT "[HOME][DOWN][DOWN][DOWN][DO
WN]"TAB(18)"    LOADING      [DOWN]"
2075 KL=INT((15-LEN(FL$))/2)
2080 PRINT TAB(18)RIGHT$(BL$,KL)FL$LEFT
$(BL$,20-KL-LEN(FL$))
2090 PRINT TAB(18)"[DOWN]
  "
2100 OPEN 2,1,0,FL$
2105 GET#2,A$:TS=ASC(A$)*2048:FORI=0TO2
047:GET#2,A$:POKETS+I,ASC(A$+CHR$(0)):NEXT
2110 CLOSE2:PRINT "[HOME][DOWN][DOWN][D
OWN][DOWN]"TAB(18)"                  [DOWN]"
2120 PRINT TAB(18)"                   ":G
OSUB 1160:GOSUB 2500:RETURN
2500 RESTORE:FOR I=0 TO 98:READ XZ:POKE
 896+I,XZ:NEXT:POKE 953,PT:RETURN
2510 DATA 160,7,177,251,162,0,42,62,157
,3,232,224,8,208,247,136
2520 DATA 16,240,160,7,185,157,3,145,25
1,136,16,248,96,234,234,234
2530 DATA 234,234,234,234,234,169,148,1
33,251,169,5,133,252,162,7,160
2540 DATA 7,169,46,126,157,3,144,2,169,
81,145,251,136,16,242,56
2550 DATA 165,251,233,40,133,251,165,25
2,233,0,133,252,202,16,224,96
2560 DATA 162,8,160,0,177,251,145,253,2
00,208,249,230,252,230,254,202,208,242,
96

READY.
```

Listing 3

```
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
```

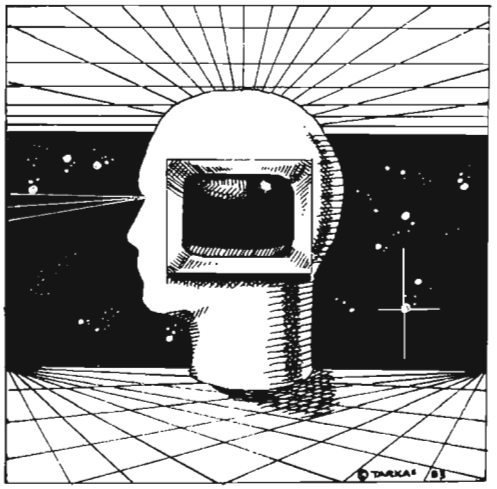
```
4 IFT<16THENPRINT",";:GOTO2
5 L=L+I:PRINT"[BLUE][HOME][DOWN][DOWN]0
DATA";L;"[LEFT],";I;"[LEFT],";S;"[LEFT]
,";E:PRINT"[HOME][DOWN][DOWN][DOWN][DOW
N][DOWN]RUN";
6 PRINT"[HOME]";:POKE631,13:POKE632,13:
POKE633,13:POKE634,13:POKE198,4:END
7 PRINT"[CLEAR][WHITE]"TAB(2)"_________
_____________________"
8 PRINTTAB(2)"[REV] CHARACTER DATA TO D
ATA 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]WHICH CHARACTER SET ";S:INPUT"[DOWN
][RIGHT][RIGHT][RIGHT][RIGHT]BEGIN AT W
HICH CHARACTER ";C
11 INPUT"[DOWN][RIGHT][RIGHT][RIGHT][RI
GHT]HOW MANY CHARACTERS ";H:S=S*2048+C*
8:E=S+H*8
12 PRINT"[CLEAR]":GOTO5

READY.
```


Circle No. 14

A Comprehensive Editor/Assembler for the VIC-20

Part I

by Eric Giguere
Peace River, Alberta



A few months ago I promised the readers of my column on assembly language that I would publish an assembler for the VIC. True to my word, I now present to you here the program EDIT/ASM. There is also the added benefit that the program is easily adaptable to any Commodore machine with only a few changes. But more on that later, as I now proceed to introduce you to EDIT/ASM.

What Is An Editor/Assembler?

Put very simply, an editor/assembler is a program or group of programs that lets you enter assembly language code, modify (edit) it, save it, and then convert it into actual machine language code (assemble it). If assembly language is totally new to you and you are interested in learning about it, I suggest that you either buy a book on it (there are several good titles available), or at least read my monthly series on it (Assembly Language Programming on the VIC-20), starting with the first issue of Commander. Otherwise this program will be of absolutely no use to you.

Most editor/assemblers have two parts—an editor and an assembler. EDIT/ASM is no exception, and consists of two separate yet similar programs (in BASIC). The first, which I am presenting here, is the Editor Module. It allows you to enter code, edit it with useful commands, and then save it for use by the second module. This module, the Assembler Module, then takes your program and converts it into actual machine code, which you will then be able to load and run (as long as your coding works). I've patterned EDIT/ASM after the Apple's EDASM, and so users of the Commodore assembler may find this one a bit strange at first. But it does work, and I'm sure you'll find it very useful in all your assembly language work.

Hardware Requirements

To be really workable, EDIT/ASM requires a 13K VIC with disk drive, but it can also work with cassette (changes are given later on). It will work with an 8K VIC (VIC + 3K expander), but it doesn't leave much of a workspace. But it will work!

The Editor Module

The editor is probably the most important part of an assembler, as it is the part that interacts with the user. Bearing this in mind, I have tried to make the editor in EDIT/ASM as easy to use as possible. Source code (the assembly language instructions) can be easily entered and edited, and makes full use of labels, equates and comments. In fact, you get quite a deal for the price of a magazine. Let's now see what happens when we RUN EDIT/ASM.

First, the screen should turn to the normal blue border/white background combination and the screen clear. In the upper left corner there will be the word "LIMIT" and a question mark, thus indicating the computer is waiting for you to input some information. What it actually wants is the maximum number of lines you wish to use during this run of the assembler. It will accept any number greater than zero, as long as you have the memory space for it. It will then DIM that amount of memory for use by the array that holds the source code in memory. A practical ceiling for LIMIT (represented by the variable LI in the listing) is 1000. I doubt that many people are going to use more than that, and if they do, they'll have two problems to contend with: 1) the line numbers are truncated to three numbers when listed, so you'll only see lines from 1 to 999; and 2) the garbage collection (the discarding of unwanted variables) becomes very time consuming, producing noticeable time delays before the actual execution of the commands. That is why you should keep your LIMIT down to a practical limit (the reason I put the inputting of LIMIT into the program is that you could then use EDIT/ASM in any VIC memory configuration 8K + without having to modify the variable LI manually inside the program. If you want to keep LI fixed, simply delete line 45 and replace it by 45 LI = n, where n is the fixed number.)

After having entered a value for LIMIT (let's say it was 100), the screen will clear once more and the title and copyright notice will appear just to remind you whose program this actually is. You should then notice a colon and a blinking cursor a few lines down. This is your signal that the computer is waiting for you to do something. I have used a colon (":") as a prompt to distinguish EDIT/ASM from BASIC (with commands like LIST and NEW, it's easy to confuse the two). It will appear whenever the editor is waiting for a command.

Now to do something. Type in HELP and hit RETURN. A list of all the available commands should appear, fol-

lowed by their correct syntax. Following is a list of each command and its purpose. Note: In each of the following, the period represents a space, and should be typed in as such. It has been used here for added clarity to the list.

ADD—allows you to add source code to the end of the present code in memory, or to start a new program if there is no code in memory. This mode is exited by hitting the "/" key and pressing RETURN.

CATALOG—an adaptation of a routine by Jim Butterfield, this command displays a list of all the programs on diskette, as if you had LOADed the directory. At any time the listing may be frozen by holding down a key, or aborted by hitting the SPACE bar, which will return you to the command mode.

DELETE.a(.b)—allows the deletion of the specified lines from memory. A line number is necessary and may use the following syntax:

DELETE.2—delete line #2
DELETE.-.3—delete from line 1 till line 3
DELETE.2.6—delete lines 2 to 6
DELETE.5.-—delete from line 5 to end of program

Remember: All periods represent spaces which must separate each part of the command. DELETE-5 is wrong because there is no space separating the hyphen ("-") and the number 5.

DISK—allows the sending of commands to the disk drive. Enter this command and press RETURN. A prompt ("]") will appear signifying that whatever you type next will be sent to the disk. If you don't wish to do so, simply hit the up-arrow key beside the RESTORE key and hit RETURN. You will be returned to the command mode. Another thing you can do is read the error channel. When prompted with the bracket, type in ERR and hit RETURN. You will be shown the current status of the error channel. Anything else is sent to the disk (you may use the DEL key to correct any errors—these will not be sent).

EDIT.a—this command allows you to change an existing line of code in memory, as specified by "a". The line will be listed to the screen and a cur-

```
||||||||||||||||||||||||||||||10 REM ***
12 REM *             *
14 REM *  EDIT/ASM   *
16 REM *-------------*
18 REM *    EDITOR   *
20 REM *    MODULE   *
22 REM *-------------*
24 REM *             *
26 REM * (C)1983 BY *
28 REM *             *
30 REM * E. GIGUERE *
32 REM *             *
34 REM ***************
35 :
37 :
39 :
45 POKE36879,27:PRINTCHR$(142)"[CLEAR]L
IMIT";:INPUTLI$:LI=VAL(LI$):IFLI<1THEN4
5
48 :
49 REM DEFINE VAR.
50 R$=CHR$(13):P$=CHR$(16):QU$=CHR$(34)
:CS$="ADINEDLILOSAQUNEDICADEPLHE"
60 OPEN15,8,15
70 DIMA$(LI+1):X=FRE(0):POKE788,194
80 PRINT"[CLEAR][DOWN][DOWN][DOWN][DOWN
][DOWN][DOWN][DOWN][RIGHT][RIGHT][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT]EDIT/ASM":
PRINT"[RIGHT][RIGHT][RIGHT][RIGHT][RIGH
T][RIGHT][RIGHT]─────────":PRINT"[DOWN][
RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGH
T](C)1983 BY"
85 PRINT"[DOWN][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT]ERIC GIGUERE[DOWN][DOWN][DOW
N][DOWN]"
98 :
99 REM GET COMMAND
100 PRINT"[BLUE][DOWN]:";:GOSUB1000:PRI
NT:Z$=IN$:GOSUB1010:COM$=LEFT$(A$,2)
102 Y=0:FORZ=1TO25STEP2:IFCOM$=MID$(CS$
,Z,2)THENY=Z/2+.5:Z=26
104 NEXTZ:IFY=0THENPRINT"[GREEN]ILLEGAL
 COMMAND[BLUE]":GOTO100
106 ONYGOTO110,130,160,185,250,300,350,
360,380,400,470,490,550
108 :
109 REM ADD
110 L=L+1:GOSUB1025:GOSUB1050:IFLEFT$(I
N$,1)="/"THENL=L-1:GOTO100
115 A$(L)=IN$:IFL=LITHENPRINTR$"[RED]WA
RNING - NO MORE ROOMSAVE FILE NOW.[BLUE
]":GOTO100
120 GOTO110
128 :
129 REM INSERT
```

sor will appear under it. Type in the new line, press RETURN, and it is automatically entered into memory. If you don't want to edit the line, hit the "/" key and press RETURN. The line will remain as it was.

HELP—displays a list of all the available commands in case you get stuck.

INSERT.a.b—allows the insertion of code between existing lines. The "a" specifies the line to insert at, and "b" specifies the number of lines to insert. Omission of any of these gives an error. The following is an example of using INSERT:

INSERT.3.10—will allow you to insert ten lines starting at line 3.

Insert may be aborted at any time by typing "/" as the first letter of the line, but it will leave the remaining lines that were inserted blank.

LIST(.a)(.b)—will list the present lines of code in memory to the screen. LIST by itself will list the entire program, while LIST with any other parameters will list only those. The following are acceptable syntax:

LIST.2—list line #2
LIST.-.3—list to line #3
LIST.3.4—list lines 3 to 4
LIST.5.-—list from lines 5 to end of program

At any time the listing may be frozen by holding down a key, or aborted by pressing the space bar.

LOAD(.filename)—will load a file from cassette or disk. Adds ".SOU" to the end of the filename to make sure it loads only source code created by this program—not object code (created by the assembler). If the file is too large to fit into the present LIMIT, an error message will appear and the operation will be aborted.

NEW—allows the creation of a new source code program. If typed accidentally, exit the program using QUIT. Then type in direct mode L = n, where n equals the last line of your source code before being NEWed, and press RETURN. Then type GOTO 100 and you will be returned to the command mode. A LIST should reveal that your program is now back in memory.

```
130 IFB=0ORC=0THEN1030
135 IF(B<10RB>L)OR(C+L>LI)THEN1030
140 FORI=LTOBSTEP-1:A$(I+C)=A$(I):A$(I)
=""":NEXT:L=L+C
150 TE=L:FORL=BTOB+C-1:GOSUB1025:GOSUB1
000:A$(L)=IN$:IFLEFT$(IN$,1)="/"THENL=B
+C
155 NEXTL:L=TE:GOTO100
158 :
159 REM EDIT
160 IFB<10RB>LTHEN1030
165 TE=L:L=B:PRINT"[DOWN][BLACK]";:GOSU
B1025:PRINTA$(L)"[BLUE][DOWN]"
170 GOSUB1025:GOSUB1050:IFLEFT$(IN$,1)=
"/"THENL=TE:GOTO100
175 A$(L)=IN$:L=TE:GOTO100
183 :
184 REM LIST
185 IFL=0THEN100
190 GOSUB1075:IFETHEN1030
195 TE=L:FORL=SLTOEL:GOSUB1025:Z$=A$(L)
:GOSUB1010:GOSUB1045
197 IFLEFT$(Z$,1)="*"ORLEFT$(Z$,1)=";"T
HENPRINTA$(L):GOTO202
200 PRINTA$;TAB(11)B$;TAB(15)C$:IFD$<>"
"THENPRINT"[BLUE]"D$"[BLACK]"
202 GETA$:IFA$=" "THENL=EL+1:GOTO210
205 WAIT197,64
210 NEXTL:L=TE:GOTO100
248 :
249 REM LOAD
250 IFB$=""THEN1030
255 PRINT#15,"I0":OPEN2,8,2,"0:"+B$+".S
OU,S,R":INPUT#15,E,E$:IFE>20THEN1040
260 INPUT#2,L:IFL>LITHENPRINT"[DOWN][PU
RPLE]FILE TOO LARGE[BLUE]":L=0:GOTO270
265 FORI=1TOL:INPUT#2,A$(I):NEXT
270 CLOSE2:GOTO100
298 :
299 REM SAVE
300 IFB$=""THEN1030
302 PRINT#15,"I0"
305 PRINT#15,"S:"B$+".SOU":OPEN3,8,3,"0
:"+B$+".SOU,S,W":INPUT#15,E,E$:IFE>20TH
EN1040
310 IFA$=""THENPRINT:GOTO402
315 FORI=1TOL:PRINT#3,QU$A$(I)QU$:NEXT
320 CLOSE3:GOTO100
348 :
349 REM QUIT
350 PRINT"[DOWN]TYPE 'GOTO 100' TO RE-S
TART.":POKE788,191:END
357 :
358 :
```

PLIST(.a)(.b)(.title)—allows the listing of your program onto a printer. PLIST by itself lists the whole program. The rest of the syntax is like LIST and DELETE. You may also add a title to the listing using the .title reference. After typing PLIST and the parameters, hit the space bar and type in the title you wish printed. It will be printed on the first line, and the listing will follow a couple of lines down. To get a complete listing with title, type PLIST, 3 spaces, and the title. The complete program will be listed with a title beforehand.

QUIT—allows the user to quit the program. Type GOTO 100 to re-enter program intact.

SAVE.filename—allows you to save a program to disk or tape. Tacks on a ".SOU" to the end to differentiate between source and object code.

That was the complete list of commands. As you can see, they are quite powerful and versatile, making for easy entering and editing.

Entering Source Code

It may be very well to have all these commands, but they amount to nothing if you cannot use them properly. Our first priority is to learn how to enter data. The following is an explanation.

After going through the opening procedures, type the command ADD. A number one (1) should appear on the line below, followed by a blinking cursor. The number refers to the line of code you are presently entering and is used for editing purposes only. It will not be present when the object code (the machine language program) is generated. You now have three choices as to what to do: 1) enter a line of code; 2) enter a comment; and 3) exit this mode. To exit, you need simply type the ("/") and hit RETURN. You will be returned to the command mode. The other two are more complicated.

Entering a line of code is what you will be doing most often. One line consists of an instruction, and can be accompanied by a label and a comment, as such:

```
359 REM NEW
360 L=0:GOTO100
377 :
378 :
379 REM DISK
380 PRINT"]";:GOSUB1050:IFIN$="↑"THEN10
0
385 IFIN$="ERR"THENINPUT#15,E,E$:GOTO10
42
390 PRINT#15,IN$:GOTO100
398 :
399 REM CATALOG
400 PRINT#15,"I0":OPEN4,8,0,"$0":NU$=CH
R$(0)
401 GET#4,A$,A$
402 GET#4,A$,A$
405 IFA$=""THEN465
410 GET#4,A$,B$
420 PRINTASC(A$+NU$)+ASC(B$+NU$)*256;
430 GET#4,A$
440 IFA$=""THENPRINT:GOTO402
450 PRINTA$;
452 GETA$:IFA$=" "THEN465
455 WAIT197,64
460 GOTO430
465 PRINT:CLOSE4:GOTO100
468 :
469 REM DELETE
470 GOSUB1075:IFEOR(SL=1ANDEL=L)THEN103
0
480 X=EL-SL+1:FORI=EL+1TOL:A$(I-X)=A$(I
):A$(I)="":NEXT:L=L-X:GOTO100
488 :
489 REM PRINT LIST
490 IFL=0THEN100
500 GOSUB1075:IFETHEN1030
510 CLOSE4:OPEN4,4:PRINT#4,R$:IFD$<>""T
HENPRINT#4,D$;R$
515 TE=L:FORL=SLTOEL:GOSUB1025:PRINT#4,
L$;:X$=LEFT$(A$(L),1)
517 IFX$=";"ORX$="*"THENPRINT#4,A$(L):G
OTO525
520 Z$=A$(L):GOSUB1010:GOSUB1045:PRINT#
4,A$;P$"13"B$;P$"17"C$;P$"29"D$
525 GETA$:IFA$=" "THENL=EL+1
530 NEXT:L=TE:PRINT#4,R$"END OF LIST"R$
:CLOSE4:PRINT:GOTO100
549 REM HELP COMMAND
550 PRINT"[BLACK]ALL SLASHES SHOULD BE
TYPED IN AS SPACES:[DOWN]"
555 PRINT"[BLUE][REV]AD[OFF]D":PRINT"[R
EV]CA[OFF]TALOG"
560 PRINT"[REV]DE[OFF]LETE[RED]/LINE#(/
LINE#)":PRINT"[BLUE][REV]DI[OFF]SK"
565 PRINT"[REV]ED[OFF]IT[RED]/LINE#":PR
INT"[BLUE][REV]HE[OFF]LP"
```

START LDA #$00 ;COMMENT

Here START is a label to refer to that line, LDA #$00 is the instruction, and ;COMMENT is a comment. These are all separated by spaces, and are entered the same way. First you enter the label. Then you hit the space bar and enter the instruction itself (such as LDA). If the instruction requires data (such as #$00), hit the space bar again and type in the data. The comment is not necessary, but if you wish to have it, you must again type the space bar and then enter your comment. If you've noticed something, it's probably that everything is separated by a space character. This is true, as the space is used when both listing and assembling to separate the **fields**, or different parts of an assembly listing. Thus if you don't want a label, you can simply hit the space bar and type in the instruction. Similarly, if you don't need any data following the instruction but want a comment, you should hit the space bar **twice** after the instruction and then type in your comment. This may all seem a bit strange at first, but you'll soon get used to it. Simply remember that once you hit the space bar you skip to the next field, and it will become a virtual habit with you.

As for entering comments, there are two distinct ways (these comments I am now referring to are different from those that may follow an instruction). You may either type in a "*" or a ";" and then your comment. Both are acceptable, and the "*" is used mainly for decoration. When listed, comments will appear as they were typed, and will not be separated into fields as will any other lines. They're just there to help document and beautify the program.

Special Pseudo-Opcodes

Opcodes are the commands that the computer recognizes, such as LDA and CMP. Pseudo-opcodes resemble opcodes, but in fact are instructions to the assembler, not the computer. They are used to tell the assembler to do something. The pseudo-opcodes are placed where a normal instruction would usually be, but are not outputted as part of any object code. The pseudo-opcodes I have

```
570 PRINT"[REV]IN[OFF]SERT[RED]/LINE#/N
O.":PRINT"[BLUE][REV]LI[OFF]ST[RED](/LI
NE#)(/LINE#)"
575 PRINT"[BLUE][REV]LO[OFF]AD/[RED]FIL
ENAME":PRINT"[BLUE][REV]NE[OFF]W"
580 PRINT"[REV]PL[OFF]IST[RED](/LINE#)(
/LINE#)":PRINT"[BLUE][REV]QU[OFF]IT"
585 PRINT"[REV]SA[OFF]VE[RED]/FILENAME"
:PRINT
590 GOTO100
995 :
996 :
997 :
998 :
999 REM INPUT ROUTINE
1000 CLOSEK:OPEN1,0:INPUT#1,IN$:PRINT:C
LOSE1:RETURN
1009 REM ROUTINE TO FIND FIELDS
1010 FORY=0TO2:SP(Y)=0:NEXT:Y=0:FORZ=1T
OLEN(Z$)
1012 X$=MID$(Z$,Z,1):IFX$=" "THENSP(Y)=
Z:Y=Y+1:IFY>2THENZ=LEN(Z$)+1
1014 NEXTZ:A$="":B$="":C$="":D$=""
1016 IFSP(0)=0THENA$=Z$:GOTO1024
1018 A$=LEFT$(Z$,SP(0)-1):IFSP(1)=0THEN
B$=MID$(Z$,SP(0)+1):GOTO1024
1020 B$=MID$(Z$,SP(0)+1,SP(1)-SP(0)-1):
IFSP(2)=0THENC$=MID$(Z$,SP(1)+1):GOTO10
24
1022 C$=MID$(Z$,SP(1)+1,SP(2)-SP(1)-1):
D$=MID$(Z$,SP(2)+1)
1024 A=VAL(A$):B=VAL(B$):C=VAL(C$):D=VA
L(D$):RETURN
1025 L$=MID$(STR$(L),2):L$=RIGHT$("    "
+L$+" ",4):PRINTL$;:RETURN
1028 :
1029 REM ERROR MESSAGE
1030 PRINT"[RED]ILLEGAL PARAMETERS[BLUE
]":GOTO100
1038 :
1039 REM DISK ERROR
1040 PRINT"[DOWN][RED]DISK I/O ERROR -"
1042 PRINT"[DOWN][BLACK][LEFT]"E"[LEFT]
,"E$:CLOSE2:CLOSE3:GOTO100
1043 :
1044 REM TRUNCATE STRINGS
1045 A$=LEFT$(A$,6):B$=LEFT$(B$,3):C$=L
EFT$(C$,10):RETURN
1050 WAIT197,64:POKE212,0:POKE204,0:X$=
"":Y$=CHR$(20):IN$=""
1055 POKE207,0:WAIT198,255:POKE207,255:
GETX$:IFX$>Y$THENIN$=IN$+X$:PRINT" [LEF
T]"X$;:GOTO1055
1060 IFX$=CHR$(20)ANDLEN(IN$)THENIN$=LE
FT$(IN$,LEN(IN$)-1):PRINT" [LEFT][LEFT]
 [LEFT]";:GOTO1055
```

included in my assembler go as follows:

ASC—places the ASCII value of a string in consecutive bytes of memory. Ex: MESSG ASC "HELLO"—will place the word HELLO in ASCII form in memory and give that location the label MESSG.

BYT—places individual values into consecutive memory locations. Ex: VALUES BYT 1,34,56,$FB—will place the values 1,34,56 and $FB at the location VALUES.

DST—defines storage space by telling the assembler to skip ahead a certain number of bytes without actually filling them in with a value. Ex: TABLE DST + 10—makes the assembler give the label TABLE to the assembler's present memory location and then skip ahead the specified number of bytes, in this case 10.

EQU—assigns a label a particular value that you specify. May be one or two bytes long, hexadecimal or decimal. Ex: BLANK EQU $20—gives the label BLANK the value $20 hex or 32 decimal.

ORG—defines the starting location of the code to be assembled. Can only be used once in program. Ex: ORG $033c—defines the program to start at $033C (828). No label is used.

These instructions will be discussed more in part 2.

```
1065 IFX$=CHR$(13)ANDLEN(IN$)>0THENPRIN
T" ":POKE204,1:POKE207,0:RETURN
1070 GOTO1055
1072 :
1073 :
1074 REM FIND PARAM.
1075 PRINT"[BLACK]";:IFB=0ANDC=0THENSL=
1:EL=L:GOTO1095
1080 IFB$="-"THENSL=1:EL=C:GOTO1095
1085 SL=B:EL=C:IFC$="-"THENEL=L:GOTO109
5
1090 IFC=0THENEL=SL
1095 E=0:IFSL<1OREL<SLOREL>LTHENE=1
1097 RETURN

READY.
```

Labels

One of the more useful parts of an assembler is the usage of labels. Labels are words that refer to a certain part of a program or to a certain value. For example, BNE L00P1 would mean to branch if not equal to the code with the label L00P1. A line like

STL00P LDA TABLE,X

would load the accumulator from the location defined by the label TABLE plus the X-register. TABLE may have been previously defined by an ASC,BYT,DST, or EQU statement. In any case, labels are easy to use and very useful.

Changes

Changes for cassette users follow the program listing. They are very few. The only real difference is that the cassette users lack the DISK and CATALOG commands, which have no use with tape. As for other Commodore computers, there are very few changes to be made. The deletion of all color commands is one. Another is the replacement of all WAIT 197,64 by an appropriate WAIT or GET (all the WAIT does is wait until you are not pressing any keys). Also, the two pokes to location 788 should be deleted. They disable the STOP key. And finally, the pokes in the input routine that turn on the cursor and then turn it off, as well as the WAIT command that waits until a keypress has been signaled into the keyboard buffer. (You may also wish to change the screen format.) But apart from these, there are no real changes to be made to the program proper.

Next Month

Next month I'll be presenting the assembler module and the instructions for using it. I won't be offering to make copies yet as you should have the instructions as how to use both parts of the utility. I'll leave you with a sample of the printer output of the editor module. You may contact me at: Box 901, Peace River, Alberta, Canada TOH 2X0. □

```
PROGRAM CHANGES FOR CASSETTE USERS:

DELETE LINES:
     60
     380-397
     400-460
     1040
CHANGE THE FOLLOWING LINES:
CS$ IN LINE 50 SHOULD READ "ADINEDLILOSAQUNEDEPLHE"

IN LINE 102 THE FOR...NEXT STATEMENT BECOMES:
     FORZ=1TO21  INSTEAD OF   FORZ=1TO25

106 ONVGOTO110,130,160,185,250,300,350,360,470,490,550

255 OPEN2,1,0,B$+".SOU"

305 OPEN3,1,1,B$+".SOU"

555 PRINT"[BLUE,RVS]AD[RVOFF]"

560 PRINT"[RVS]DE[RVOFF]LETE[RED]/LINE#(/LINE#)"

INSERT A [BLUE] CURSOR CONTROL IN LINE 565 JUST BEFORE THE [RVS]

THOSE ARE ALL THE CHANGES.
```

JOYSTICK ROUTINE

```
 1 ;*************
 2 ;** JOYSTICK **
 3 ;** ROUTINE1 **
 4 ;*************
 5 ;
 6           ORG $033C
   ;CASSETTE BUFFER
 7 ;
 8 DDRA      EQU $9113
 9 DDRB      EQU $9122
10 ;
11 PORTA     EQU $9111
12 PORTB     EQU $9120
13 TABLE     DST +5
14 ;
15 START     LDA #$00
16           STA DDRA
   ;PREPARE PORTA FOR
    INPUT
17           TAX
18 GETJ0     LDA PORTA
19           AND #$04
20           STA TABLE,X
21           INX
22 GETJ1     LDA PORTA
23           AND #$08
24           STA TABLE,X
25 GETJ2     LDA PORTA
26           AND #$10
27           STA TABLE,X
28           INX
29           INX
30 GETFB     LDA PORTA
31           AND #$20
32           STA TABLE,X
33           DEX
34 SETDDR    LDA #$7F
35           STA DDRB
36 GETJ3     LDA PORTB
37           AND #$80
38           STA TABLE,X
39           LDA #$FF
40           STA DDRB
41 EXIT      RTS
```

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

Circle No. 60

Wolf Whistle

Editor's Note—We at Commander would like to give Neil Harris, Editor of the Commodore Magazine, credit for the "Wolf Whistle" listing in "CB2 Sound" written by Louis Sanders, which was published in our December Premier Issue.

Answers to White Flashes on the Screen

Dear People:

In response to Vincent J. Monney, Jr.'s article in Bits & Pieces—Vincent's problem with white flashes when his 64 is in the RUN mode may be cured by replacing the 6510 Microprocessor. I had a similar problem. The chip is easily replaced if you are careful or take it to your service center and have them do it.

Sincerely, Richard Medrano, 3310 Kimber Court 133, San Jose, CA 95124.□

In your March/April issue, in the Bits & Pieces department on page 42, a reader asked about ". . . enormous white flashes on the screen whenever the computer is running."

This reader can take his/her 64 to the nearest Commodore repair facility to get the unit repaired. The repair person will be adding a .33 ut capacitor onto the circuit board to remove this problem.

This modification is already done on current production units and will be done at no charge to the customers on part production modules.

Regards, Tony LaMartina

Selective Keyboard Entry for Commodore 64

REF: Appendix M, page 429 of the Commodore 64 Programmers Reference Guide—I/O Ports (PRA, PRB, DDRA, DDRB).

PRA/PRB—Peripheral Register. Actual I/O Port.

DDRA/DDRB—Data Direction Register. Controls which pins of the peripheral registers are Input and Output.

A) If a BIT in the DDR is set to a one (1), the corresponding BIT in the PR is an Output.

B) If a BIT in the DDR is set to a zero (Ø), the corresponding BIT in the PR is an Input.

Example

DDR	1	Ø	1	Ø	1	Ø	1	Ø
PR	OUT	IN	OUT	IN	OUT	IN	OUT	IN

Example of Selective Keyboard Entry

```
JSR
 ↓
LDA  #$ØØ
STA  $DCØ3   DDRB (Input) All
LDA  #$FF
STA  $DCØ2   DDRA (Output) All
LDA  #$F7
STA  $DCØØ   PRA (enable row 3)
LDA  $DCØ1   PRB
CMP  #$F7    is the "8" key
  depressed
BNE  $_____
JSR  $_____  ⟶ SUBROUTINE
LDA  #$FF    ⟵
STA  $DCØØ   PRA
RTS
 ↵
```

Keyboard Selective Entry

*May only be used in a machine language routine.

1) Set DDRB to Input. $DCØ3 = $ØØ

2) Set DDRA to Output. $DCØ2 = $FF

3) Enable selected row by placing a zero Ø in the corresponding PRA BIT position. $DCØØ??

4) Read PRB and Compare to value corresponding to select key entry CMP $DCØ1 ??

5) Decide action to take

6) Clear PRA $DCØØ = FF

7) Return from subroutine

Note—

When key is depressed on a selected row, the corresponding BIT in the PRB will go low (Ø). All other BITS will remain high (1).

Sincerely,
Joseph E. Albritton

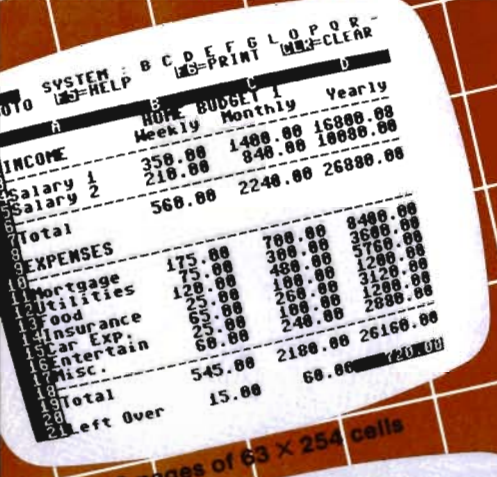
COMMODORE 64 KEYBOARD MATRIX (CIA #1)

ROW SELECT	MSB PB1	PB6	PB5	PB4	PB3	PB2	PB1	LSB PB0
	COLUMN READ							PRB INPUT $DC01
								DDRB = $DC03 = $00
PA0	↑↓	F5	F3	F1	F7	← →	RET	DEL
PA1	(L) SHFT	E	S	Z	4	A	W	3
PA2	X	T	F	C	6	D	R	5
PA3	V	U	J	B	8	G	Y	7
PA4	N	O	L	M	0	J	I	9
PA5	,	@	:	.	-	L	P	+
PA6	/	↑	=	(R) SHFT	HME	;	X	£
PA7	STOP	Q	Q	SPC	2	CTRL	*	1

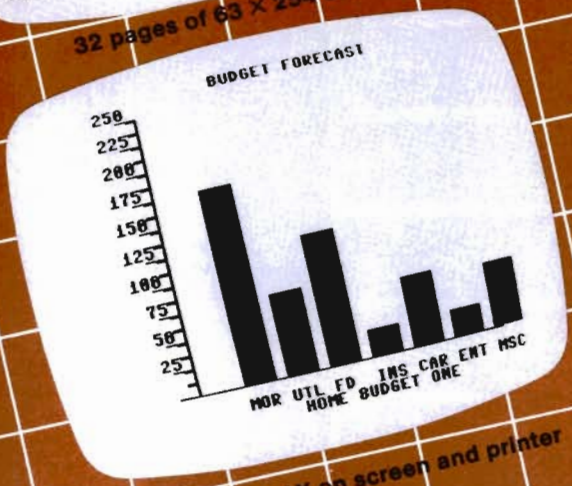
LSB

MSB

PRA OUTPUT $DC00
DDRA = $DC02 = $FF

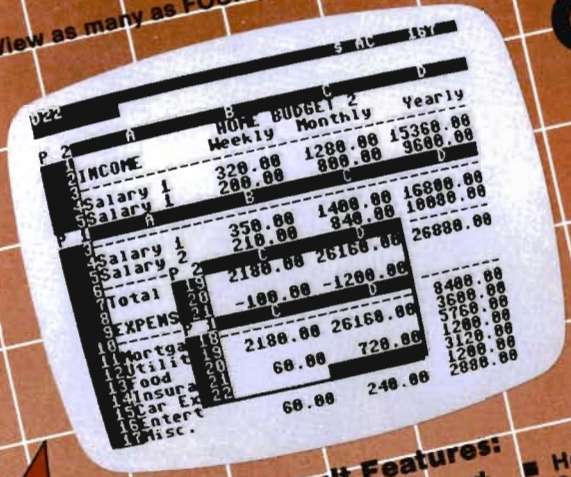


32 pages of 63 × 254 cells



GRAPHIC DISPLAY on screen and printer

View as many as FOUR pages at one time



the foxiest Calc Result™

PLANNING and FORECASTING TOOL AVAILABLE

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!"

News Releases

marketing a la carte

Donald E. Rosner recently joined **marketing a la carte** as a partner. Previously, Rosner was with CAPEX CORPORATION for 13 years as a Vice President. CAPEX (now Computer Associates) is a major supplier of software to the large-scale IBM computer users. During Rosner's career with Capex he had numerous responsibilities including marketing, sales, long range planning, and product acquisitions.

"The addition of Don Rosner to **marketing a la carte** gives us expertise in some very specialized software marketing areas, such as software distribution, acquisitions, and long range planning," explains Barbara J. Walter, partner and company founder.

marketing a la carte offers marketing and sales services to computer software companies. It specializes in:

- Solving software distribution problems for vertical markets.
- Developing video training courses and video brochures for software.
- Marketing communications projects: brochures, newsletters, direct mail, advertisements, press releases, etc.
- Acquiring or selling software products.
- Establishing dealer incentive programs for software companies selling through dealers.

Many software companies cannot support a separate marketing department or sales force. **marketing a la carte** began business a year ago to offer sales and marketing services on an "a la carte" basis. In addition, it offers the large software companies assistance on special marketing projects for which they do not have extra manpower.

marketing a la carte is located at 4929 North 43rd Street, Phoenix, AZ, (602) 952-2918. Circle No 87

Commodore Capability

In recognition of the growing base of Commodore computer owners, the growing popularity of the VIC-20 and Commodore 64 and the lack of good software for them (particularly for the 64), Synapse Software has established a line of Commodore game software.

Having assigned several key designers to the project, Synapse will be releasing 8 Commodore compatible games during the first half of 1983, primarily in April and May. All the games are updates and improvements on highly popular and successful Atari 400/800 products.

The line-up includes: Astropatrol, Fort Apocalypse, Gridworld, Drelbs, Survivor, Slamball, Pharoah's Curse, and Harrier, and will receive extensive, separate advertising support.

Synapse Commodore products will all remain faithful to the graphic excellence, challenge and playability that distinguish their other games; all will broaden the rather narrow horizons of Commodore software currently available; and all are guaranteed Instant Hits. Synapse, 5221 Central Avenue, Richmond, CA 94804, (415) 527-7751. Circle No 88

"PIPES" First in Home Education from Creative Software

Creative Software is pleased to announce the immediate release of "PIPES" for the VIC-20. This home concept education program, designed and written by John Doering, is the first in a series of titles intended to bring the worlds of education and games together.

The object of the program is to connect all of the houses in Gilroy, CA to the main water tower. You have just so much money to buy different lengths of pipe and you have just so many pieces of pipe available to you. The brilliant graphics and realistic sound make this educational piece play like a game while teaching children between the ages of 6-15 the concepts of spatial relationships and economics. Adults will enjoy the strategic challenge of "PIPES."

"PIPES" is a cartridge based program available on the Commodore VIC-20 and will be available for the Commodore 64 by late 1983.

Creative Software, dedicated to publishing a full-line of consumer software programs, was founded in 1977. Its offices are at 230 East Caribbean Drive, Sunnyvale, CA 94086. Circle No 89

Gypsy

It's a jungle out there! Here you are, a happy-go-lucky bug in the garden of your dreams . . . you have spotted an especially juicy Qok tree in some human's back yard and decided to settle down to some serious leaf munching.

But beware!! You are not the only one who considers this tree their home. Before you sink your teeth into the luscious foliage, be prepared to face enemies on all quarters. Nearby grazes the infamous Flying Ant of Tasmania, a stubborn and ill-tempered beast. He is busy about his own work,

but if you disturb him he will give you a nasty bite!

But ants are not your greatest worry. There is much greater danger here. The Poisonous Mushrooms at the foot of this tree are definitely to be avoided. Also in pursuit, the Fandango Bee, who has already become notorious in lower Sumatra because of her foul temper and selfish disposition. Last, but certainly not least of your worries, is the Locknest Spider who spends most of his time dangling from his sticky silk thread.

So eat while you can. Each leaf section is worth points on the widely accepted Blintz nutritional scale. The nectar from the flowers of Qok is also worth points, and you must take care not to travel on an empty stomach. But most important, avoid your predators: SURVIVAL is the name of the game in Gypsy.

For the: Atari 4/800 Diskette + Joystick, 32K: $26. Atari 4/800 Cassette + Joystick, 16K: $21. TI 99/4A Cassette available later this spring!

Microcomputer Games, 4517 Harford Rd., Baltimore, MD 21214.□
Circle No. 90

New for Apple IIe RGB Color Board

A new video board that provides the Apple IIe computer with RGB (red/green/blue) video signals that enhance the resolution and color quality of the supplied composite video, is now available. The board can be used with 80 column text so that color graphics and text are displayed on one RGB Monitor. In addition, each text line can be set to any one of 8 colors on any of 8 colors of background. SYNC signals are ± TTL composite or ± TTL separate horizontal and vertical, for universal RGB Monitor compatibility. The board plus into slot #7 and comes with 5' of ribbon cable for signal output. Optional cable connector and longer length cable is available.

Model VCB-2e—$169.00; Model VCB-A2, with VSS-80 RGB/80 column soft switch, is available for the Apple II.□
Circle No. 91

Commodore Computers Used to Teach Preschoolers

Preschoolers are stepping into the future as they use the Commodore PET and the Commodore 64 to develop basic skills. The children, ages three to six, attend Kindercare Learning Centers in three cities, Minneapolis, MI; Houston, TX; and Montgomery, AL, where an innovative computer learning program is available.

Since the preschoolers do not yet read, they are given directions by a natural voice recording played on a tape recorder connected to the computer. The children who use a light pen to answer questions, are being taught pre-math and pre-reading concepts memory skills, colors, shapes, and concepts such as over/under.

Working with Fisher Scientific, Inc., a Commodore Dealer specializing in educational sales, Kindercare, the largest nationwide childcare facility, started using the Commodore PET to teach preschoolers in June of 1982 at eight centers in Minneapolis. As this advanced educational technique proved to be successful, the program was expanded to 35 centers in Houston.

The program has been accepted with enthusiasm by both parents and students. The Commodore computers have proved to be so reliable and successful as a teaching tool that in January of 1983, the program was expanded once more and the Commodore 64 was installed in 11 centers in the Montgomery, AL area. Commodore Business Machines, Inc., 487 Devon Park Drive, Wayne, PA 19087.□
Circle No. 92

Mailing List for the VIC-20

Galactic Software announces the availability of the '20 Mailing List which gives the VIC user professional quality, low-priced software.

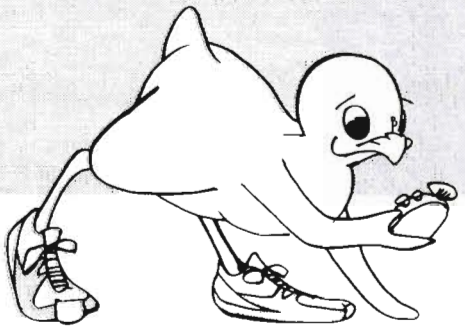
The '20 Mailing List comes in two versions: one for tape and one for disk. Both come in attractive binders with complete documentation.

The '20 Mailing List gives you capabilities reaching into alphabetizing upon entry, sorting and searching on all fields, printing labels and printing complete records. With each record containing Name, Address, City, State, Zip, three comment fields, you not only have a complete mailing list but a small data base manager.

The '20 Mailing List is completely menu driven and user friendly. It will run on any VIC that has a 16K or more expander.

The price is $25.95 for tape and $27.95 for the disk version.

Contact: Galactic Software, PO Box 10516, San Jose, CA 95157, (408) 247-4434.□
Circle No. 93



It's Time for TOTL SOFTWARE!

for the VIC 20™ and COMMODORE 64™
WORD PROCESSING AND MAILING LIST & LABEL
now available with **CHICKSPEED**

FAST PRINTING • LIGHTNING LOADS • SIMPLE COMMANDS

TOTL.TEXT 2.0 + CS VIC + 8K expansion	$25.00
TOTL.TEXT 2.5 + CS VIC + 16K expansion	$35.00
TOTL.TEXT 2.6 + CS Commodore 64	$40.00
TOTL.LABEL 2.1 + CS VIC + 16K expansion	$20.00
TOTL.LABEL 2.6 + CS Commodore 64	$20.00
TOTL TIME MANAGER 2.1 VIC + 8K expansion	$30.00
TOTL TIME MANAGER 2.6 Commodore 64	$35.00
time management, scheduling, reports	
RESEARCH ASSISTANT 2.0 VIC + 8K expansion	$30.00
RESEARCH ASSISTANT 2.0 Commodore 64	$35.00
key word cross-reference research tool	
TOTL.BUSINESS 3.0 VIC + 16K expansion	$85.00
TOTL.BUSINESS 3.6 Commodore 64	$95.00
business programs require disk and are shipped on disk	
One Megabyte Fuzzy Diskette	$25.00
computer novelty pillow	

All programs work with 40/80 column (VIC) and 80 column (64) adapters—compatible with tape or disk systems—shipped on cassette tape—available on disk $4.00 extra.

Quality You Can Afford

Available at your local dealer or by phone order

TOTL software inc.

1555 Third Ave., Walnut Creek, CA 94596

MasterCard VISA Call (415) 943-7877

Circle No 46

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

24K MEMORY EXPANSION BOARD for the VIC-20™

ORDER FACTORY DIRECT

$119.95

ADD $2.00 FOR POSTAGE & HANDLING
TEXAS RESIDENTS ADD 5% SALES TAX

- PLUGS INSIDE VIC-20 LEAVES EXPANSION SLOT FREE
- USES HIGHEST QUALITY PARTS
- 90 DAY WARRANTY
- COMPLETELY ASSEMBLED NO SOLDERING

DYNAMIC TECHNOLOGIES

2104 CHALICE ROAD
ARLINGTON, TEXAS 76014
(817) 261-8363

VIC-20 is a registered TM of Commodore Business Machines

Circle No 50

Don't just save money at K-12 MicroMedia's Software Sale

–Learn Something.

Because of our recent high volume purchase of *Creative Computing's* best educational and game software, we can offer you these well reviewed programs *well below list price* for *free no-risk 30-day evaluation.* So if you've been waiting for a chance to try before you buy, you've come to the right ad. And, because we have better things to do with our time than process a pile of returns, we're pretty sure that you're going to like what you see. Let us know.

SAVE 45%

PET Programs on Cassette

Board Games (8K)	$7.00
Conversational Games (8K)	7.00
Ecology Simulations I (16K)	12.50
Ecology Simulations II (16K)	12.50
Graphic Games (8K)	7.00
Haunted House Adventure (16K)	7.00
Sector 3 Air Traffic Controller (24K)	7.00
Social & Economic Simulations (16K)	12.50
Study Made Easy (16K)	12.50
Trucker/Streets of the City (32K)	12.50

TRS-80 Programs on Cassette

Air Traffic Controller (16K)	$12.50
Ecology Simulations II (16K)	12.50
IQ Test (16K)	7.00
Social & Economic Simulations (16K)	12.50
Super Invasion (32K)	7.00

Official school purchase orders and preview requests on school letterhead are welcomed. Individual orders must be prepaid. Include 5% for shipping and handling (minimum $2.00 charge). New Jersey residents include applicable 6% sales tax

Canadian funds add 15%

K-12 MICROMEDIA INC

172 Broadway, Woodcliff Lake, N.J. 07675 (201) 391-7555

Circle No 54

Use Any Cassette Recorder with New VIK-DUBBER

The VIK-DUBBER cassette interface allows VIC-20 and 64 users to save and load data using any standard cassette recorder. The VIK-DUBBER circuitry filters and enhances the cassette data, virtually eliminating bad loads. The VIK-DUBBER also includes several features to allow easier cassette use. It allows you to connect two cassette recorders together to make high quality backup copies of cassette programs, even machine language. It also has an indicator light and quiet audible tone to help you adjust the cassette volume for proper use, and to allow you to monitor the cassette data. Its power is from the computer, so no batteries are needed. The VIK-DUBBER comes in an attractive case, tested and ready for immediate use. It's available for $34.95 postpaid from Bytesize Micro Technology, PO Box 21123, Dept. GC, Seattle, WA 98111, (206) 236-BYTE. Include $2.50 for shipping outside the US, Canada or Mexico. Credit card and COD customers can call (1-800) 227-3000 toll free or (1-800) 792-3000 in CA. Ask for operator 225.□ Circle No 94

BE Software Offers Video Training

Business Enhancement Software, a supplier of accounting systems for the full line of Commodore computers, announces a major addition to its line: video training for customers.

"With the addition of video support, we are the first micro-software company to offer the complete product. When a customer leaves a computer store with our product, he not only gets the software to do the accounting, but he also gets the video cassette to show him how to use the software," said L. Russel Gale, President of BE Software. He adds, "Even our dealers get a complete product. We provide them with a five-minute video brochure that they can use in their stores to tell customers about our accounting software."

BE Software feels that video support for customers is a **necessity**. "We can no longer expect computer stores to know everything about every package they carry," Gale explains. "The vendor has to take responsibility. We would like to challenge the other micro software vendors to follow our lead and provide video training for their products."

In addition, BE Software has a toll free number that dealers and customers can use if they have any questions—(602) 271-9181.

BE Software is a division of Merrill Communications, a multi-million dollar high technology corporation. BE Software has been in business since 1977, handling all accounting needs for the Commodore line, and doing it at affordable prices. The company is located at 2949 West Osborn Road, Phoenix, AZ 85017.□ Circle No 95

Micro Computer Interference Control

A new 40 page catalog from Electronic Specialists presents their line of computer interference control products. Protective devices for smooth software operation include Equipment Isolators, AC power line filter/Suppressors, Line Voltage Regulators and AC Power Interrupters.

Descriptive sections are included outlining particular computer problems and suggested solutions. Typical applications and uses are highlighted. Request catalog 831.

Contact: Electronic Specialists, Inc., 171 South Main Street, Natick, MA 01760, (617) 655-1532.☐ Circle No 96

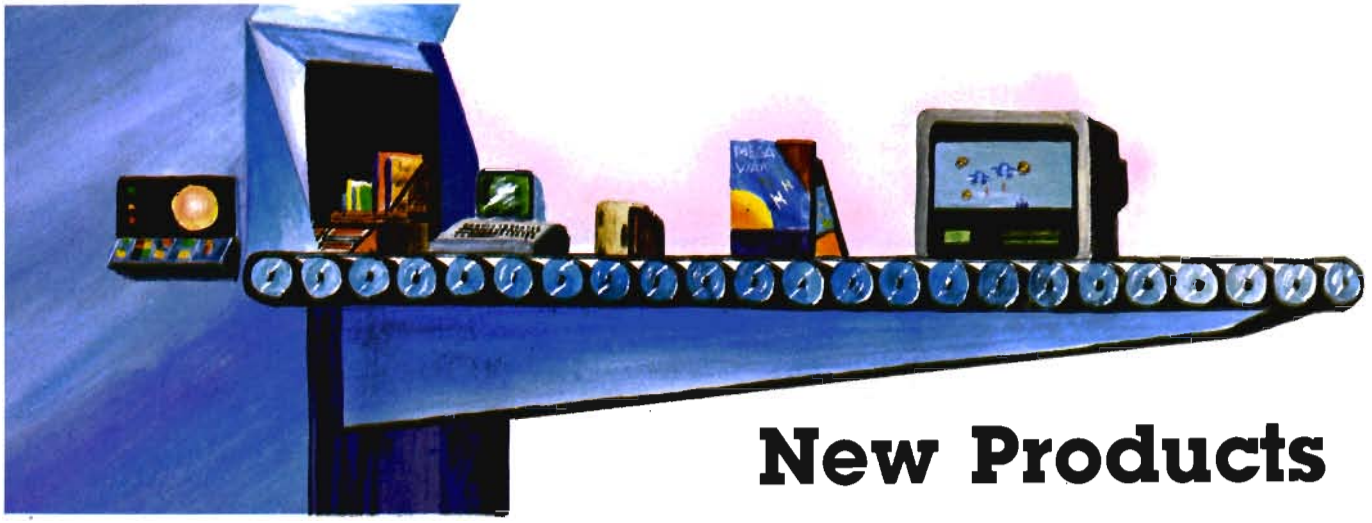
VIC-20 24K Memory Expansion Board

A fully assembled PC Board that fits inside the computer, leaving the expansion port free. Total installation time is less than five minutes and requires only a screw driver. A complete set of easy to follow pictorial instructions are included.

Memory can be increased or decreased via the selector switches which are accessible through the cassette port. With all switches in the "on" position, the computer has 28,159 Free Bytes of available memory at power up.

Price: $119.95 (factory direct).

Contact: Dynamic Technologies, 2104 Chalice Road, Arlington, TX 76014.☐ Circle No. 97



New Products

Quality Software From Scientific Educational Software

Multiplication Tables—Practice multiplication tables or administer a timed test. The test problems are presented in random order and the score is displayed at the end. Fun and easy to use. Does not require memory expansion. Order #1. $7.95.

Numeric Keypad I—Keypad and software that allow easy entry of numeric data into VIC-20. Enhances the VIC as a scientific and business tool. Requires no modification of the VIC. Contains automatic error detection and allows user to display, print, change, or delete entries. Develop your software around this device for efficient operation. Requires 8K memory expansion. Use with or without printer. Order #2. $39.95.

Numeric Keypad II—Similar to above but utilizes a heavy-duty keypad that is easy to use for extended periods in the home, office, school, or laboratory. Software allows you to save data files on tape for use later or to feed into your existing software with only minor modification; instructions provided. Requires 8K memory expansion. Use with or without printer. Order #3. $69.95.

Joystick Repair Kit—Contains the parts that break or wear out most frequently on Commodore or Atari joysticks. Use with your existing cord and base. Have one on hand when you need it. Available in black. Order #4. $3.95.

Mean, Standard Deviation, and Variance—Program calculates the mean, standard deviation, and variance of sets of numbers entered via keyboard. This program will run on the VIC with or without memory expansion. Order #104. $8.95.

Pearson r Correlation—r can be calculated for data entered by keyboard or from page. Program features mistake correction option for data entries. Enables user to save data files on tape. Printer is recommended but not required. Order #106. $15.95.

Super Randomizer—Program generates random numbers within a range specified by the user. User can specify integers, or real numbers and sample with or without duplication. Runs with or without memory expansion. Order #109. $6.95.

Super Checkbook Balancer—Program accepts as input statement balance, outstanding deposits, checks, miscellaneous charges, and credits. Outputs total outstanding deposits, total outstanding checks, and current account balance. Program features mistake correction options on all entries, and will run without memory expansion. Order #131. $8.95.

Histogram Identification Task—This three minute task was designed to assess the level of cognitive processing associated with spatial manipulations. The program requires the user to determine if a user-selectable number of histograms matches target histograms previously displayed at a different orientation. Response times, mean response times, standard deviation, user responses, histogram orientation and heights are recorded. Sixteen targets and sixteen comparison histograms are presented during a run. Randomization reduces the liklihood of administering identical histograms more than once during each running of the task. Requires 8K memory expansion. Order #144. $19.95.

Probability Change Detection Program—This three minute task was developed to determine the minimum amount of change from random probability that is required for detection by the user. Up to four dials are presented on the screen with pointers that move in random fashion until a bias occurs to which the subject responds. User presses the key with the number of the dial on which the bias is present. Bias occurrences, subject responses and reaction times are recorded. Requires 8K memory expansion. Order #145. $19.95.

Item Categorization Task—This task is designed to assess the levels of cognitive processing associated with discriminating two objects or items when three rules are used for classifying the objects. Response times, means, standard deviations, stimulus items, stimuli, and user responses are recorded and displayed. 32 items are presented per trial. Duration of this task depends on user response times. Requires 8K memory expansion. Order #146. $19.95.

Gas Mileage Calculator—Program requires as input car mileage at previous fill-up, mileage at current at previous fillup, mileage at current fillup, and the number of gallons or litres at current fillup. Subsequent calculations require only current mileage and current quantity at fillup since previous information is saved on

tape. Runs with or without memory expansion. Order #152. $7.95.

Address Label Maker—Program prints address labels and stores lists on tape for economical preparation of mailings. Great for churches and clubs. VIC 1515 or 1525 printer required. Memory expansion not required for up to 30 average length addresses. Order #180. $10.95.

All software is provided on high quality computer-grade cassettes.

Products warranted against defects in materials and workmanship for 90 days. Claims handled on an exchange basis during warranty period.

No minimum order. No charge for shipping and handling.

Programs are for Commodore VIC-20 only.

Orders accompanied with money order or cashiers check are filled within 24 hours. Otherwise allow 4 to 6 weeks.

Ohio residents please add 5½% sales tax.

VIC-20 is a registered trademark of Commodore Business Machines.

ATARI is a registered trademark of ATARI, INC.

Write for quotes concerning custom modification of products to suit your specific needs: Scientific & Educational Software, Inc., PO Box 54, Dayton, OH 45420.☐ Circle No. 66

Fun and Adventure with New Microcomputer Game

Responding to diverse customers' demands, Avalon Hill introduces a new adventure game for play on the most popular Personal Home Computers.

Space Station Zulu is an outpost of a peace-loving civilized race called Yargs. You think, just another routine tour, as you settle back into the plush comfort of the captain's chair.

ALARM! You're jolted to your senses. Suddenly, the bridge is a pandemonium. The disturbing message says: ALIENS ABOARD! Quickly you instruct your tough robots to engage the aliens. Some reports are filtering back. A robot in Landing Dock 3 reports a large concentration of what appears to be larva-stage aliens. In room B, Sgt. Olmn announces the presence of several large life forms that appear to be scattering spores. The situation looks grim. Certainly, success will depend on your daring, cunning, tactics and ability to make quick decisions.

Space Station Zulu is ready to run in diskette form on the Apple II and Atari 800 computers with 48K memory. The price is $25 each. The cassette version is available for Atari 400/800 with 32K, at $20.

We commend Avalon Hill on this new software. This gamemaker is answering its sci-fi hungry customers' demands with this new **solitare** adventure game from the microcomputer division.

The Avalon Hill Game Company, 4517 Harford Rd., Baltimore, MD 21214, (301) 254-5300.☐ Circle No. 67

3 Programs to Create and Play Music

Melody Writer—Melody Writer allows the user to type in sheet music using up to three voices and save the data on disk for future recall and editing. Melody Writer also includes a compiler program, which converts the sheet music into frequency and duration note tables, which Player utilitzes in constructing its data tables. Melody Writer allows the user a standard piano keyboard range: eight octaves, 12 notes per octave (equal tempered scale). Furthermore, the user can specify a tempo, which adjusts the duration of each note. Notes and rests can be programmed from whole to thirty-second in duration, including dotted notes. Provisions for sharps are also included. (Flats must be translated to sharps.)

The music data is stored on disk with a file name specified by the user. When compiling the data, the music data file and the voice data file must be specified. The compiled version is then saved out to disk.

Voice Programmer—This program is used to define the waveform, attack /decay, sustain/release settings of each of the three voices, and filter controls for the voices. (Also includes ring/sync/gate). The data can be saved and recalled from disk. Voice Programmer utilizes a °graphics equalizer° display to define the various on/off options. The keys f1, f3, f5, f7 select the waveform. The keys 4-7 select the voice and filter, 8 saves the data on disk and 9 recalls it. In addition, the keyboard is configured to the fifth octave, with the °a° key assigned to middle C, through the °1°key, assigned to D in the sixth octave. Sharps are also represented. (C# is assigned to the °w° key). This allows the user to sample the tonal quality of the voice and filter settings.

Player—Taking from the disk the compiled version created in Melody Maker or Sound Compuser, Player constructs a data table in memory, and executes a machine language subroutine which plays the piece. The subroutine is constructed so that any parameter of the SID can be reprogrammed at any 1/60 second interval, giving the user great flexibility in sound effect.

These programs are separated from one another so that the memory of the 64 can be used most effectively-storing data.

All three programs on disk or tape for only $24.95. Contact your local dealer or Skyles Electric Works, 231E South Whisman Road, Mountain View, CA 94041 (415) 965-1735.☐ Circle No. 68

TAXQWIK® Overview

TAXQWIK® , intended for use by **professional consultants** and **tax accountants,** has been created with the ability to process the following forms and schedules—**1040**, 1040 Estimated Tax—Lettered Schedules: **A, B, C, D, E, F, G, SE, W**—Numbered Forms: **3903**—Moving Expenses, **2106**—Employee Business Expense, **2441**—Child Care Credit, **3468**—Investment Credit **5695**—Home Energy Credit, **2210**—Underpayment of Estimated Tax, **4562**—Depreciation.

TAXQWIK® enables productivity increases in every phase of tax prepar-

ation; actual preparation of the return, checking and editing of the return, and printing and assembling of the completed return.

For example **Income Averaging**, a method of tax computation involving averaging the current year's taxable income with the taxable income for the four prior years can be computed in **less than a minute.**

The program also makes simultaneous comparisons of a client's joint return with the same client's return filed on a married, filing separately basis. Using income averaging as a base, the client can then decide which approach will better satisfy his individual tax requirements.

TAXQWIK® will also provide you with the ability to process the following **State's returns—**New York, New Jersey, Pennsylvania, California, Florida, Iowa.

TAXQWIK® prints directly onto the **Federal Forms** with the option of immediate as well as batch printing.

TAXQWIK® , The Tax Preparation System, is well documented and easy to use. This program has been in use for more than 6 years.

Specifications

Equipment Requirements—8032, 8050, 4040 Full Range of Printers.

Language—Compiled Basic.

Storage Capacity—68 Complex Returns (8050) 150 Simple Returns.

Utilities—Format a New File Disk, Back-Up Your Diskette, Validation of Date, Look-Up/Print Client File, Clear Memory.

For more information write or call Geneva Technologies Corp., 14 Commerce Drive, Cranford, NJ 07016, (201) 276-1144.□ Circle No. 70

System III Accounting System

The System III Accounting System is a fully interactive general accounting system designed for the first-time user. It is especially useful for the small business that has fairly large demands for printing, and whose files of accounts, customers, vendors, employees, and invoices are significant.

All input requests are fully prompted with complete verification of input data. Most reports may be printed either to the screen or the printer and started or stopped at any point. The user is led completely through each function by a series of highlighted prompts, fully explaining the required input at each point.

A professionally written instruction manual is included, which shows sample reports generated by the system. Further explanations of each step and prompt as it is encountered by the user, together with the reports, make the system extremely user-friendly.

The System III contains FileGuard™ I to protect irreplaceable data files, assuring the user that all data files will remain intact, even if electrical power is lost.

The System III, when used with an 8050 Disk Drive (or larger), requires only one disk for all programs and allows the user to have either a single or multiple data disks. The configuration later may be changed by the user. The system will interface all modules together, if so desired by the user.

The system may begin with less than all of the available modules and have the other modules added later. System III can be upgraded to System IV.

Computer Requirements: Commodore 8032, 8096 or SuperPET.

Suggested Configuration: 8032 Computer, 8050 Disk Drive, 4022 Printer.

Alternative Disk Versions: 4040 Disk Drive, 8250 Disk Drive, Corvus.

Contact: Southern Solutions, PO Box P, McKinney, TX 75069.□

Circle No 69

System IV Accounting System

The System IV Accounting System is a fully interactive general accounting system designed for the larger business user, especially those who must deal with large figures, $1 million up to $1 billion. It provides more modules for a wider variety of applications. The most significant feature of the system is SuperMath™, allowing entry and use of **numbers up to one billion dollars** and providing calculation accuracy of larger numbers not available in smaller systems.

Most reports may be printed either to the screen or the printer and started or stopped at any point. The system provides to options of printer widths to accommodate most printers in use: 80 and 130 columns. All input requests are fully prompted with complete verification of input data. The user is led completely through each function by a series of highlighted prompts, fully explaining the required input at each point.

A professionally written instruction manual is included, which shows sample reports generated by the system. Further explanations of each step and prompt as it is encountered by the user, together with the reports, make the system extremely user-friendly.

The System IV contains FileGuard™ II to protect irreplaceable data files, assuring the user that not only will all data files remain intact should electrical power be lost, but also that interim posting data is retained and available as processing is restarted and continued.

The System IV, when used with an 8250 Disk Drive (or larger), requires only one disk for all programs and allows the user to have either a single or multiple data disks. The configuration later may be changed by the user. Double 8050 Disk Drive units may also be used and one 8050 with some possible disk interchange is also supported. The system will interface all modules together, if so desired by the user.

The system may begin with less than all of the available modules and have the other modules added later. Just as both the Commodore 64 BusinessWare and System III from Southern Solutions can be upgraded to System IV will be able to be upgraded to future larger systems from Southern Solutions.

Computer Requirements: 8032, 8096 or SuperPET Computer.

Suggested Configuration: 8032 Computer, 8250 Disk Drive, Mannesmann-Talley 8024-L Printer.

Alternative Disk Versions: 8050 Disk Drive and Corvus.

Contact: Southern Solutions, PO Box P, McKinney, TX 75069.□

Circle No. 71

YES! Publishes Unique New Guide

A new guide to the rapidly expanding field of microcomputer books has just been published by Yes! Inc., for the Yes! Bookshop in Washington, DC. Entitled *Computers: A Comprehensive Guide*, this 64-page annotated bibliography contains short, critical reviews of over 800 of the best microcomputer books published to date.

The books in the guide are arranged by 26 separate topics. These range from the philosophical—such as artificial intelligence, computers and society—the the practical—such as business applications, programming languages, assembly languages and microprocessors, and machine-specific hardware and software.

Each book is critically reviewed by Cris Poponoe, manager of the Yes! Bookshop and author of the highly-acclaimed Yes! Bookshop Guides, *Inner Development* and *Wellness* (both distributed by Random House). Popenoe's clear, concise style—and obvious love of computers—makes reading the guide an informative and pleasurable experience. For those who have difficulty choosing, a few books in each topic are specially recommended for their superiority in content and presentation.

All of the books reviewed are regularly stocked by the Yes! Bookshop and are available through its world-wide mail order service. Complete information, including order forms, is included in the guide. For over seven years, the Yes! Bookshop has been providing its thousands of mail order customers with fast, efficient, personal service. This includes helping them with special orders and sending them free, fully annotated updates of the new books in their fields as they are published.

Computers: A Comprehensive Guide is priced at $2, postpaid, (refundable with the first book order) and is available from: Yes! Bookshop, 1035-31st Street NW, Washington, DC 20007, (202) 338-2727.☐ Circle No 72

BusinessWare™

The BusinessWare™ group of software for the Commodore 64 computer provides full-featured, computerized record keeping that is designed for the non-professional computer user, as well as the non-professional bookkeeper or record manager. Although some packages are for home or professional use, most of these systems are perfect for the very small, or at-home, business. In addition, other businesses that have limited needs for heavy-duty processing will find these packages are ideally "in-between." These systems are an excellent choice for "step-up" systems, allowing the beginning computer-user the opportunity to start using real accounting systems and concepts that can be upgraded later to a larger system, such as the System III or System IV.

All systems have menus for the user to select various operations to be performed. The programs provide prompts, to which the user responds, to move through the operation being done. Each program provides reports and printouts that provide verification of proper entry and for use in communicating away from the computer.

Each program comes complete with a guidebook that is easy to read and understand. It provides complete information necessary for learning to use the system, for explanation of each prompt and possible alternatives, and for reference for as long as the system is used.

These programs contain professional level file structures developed for use in larger systems, but available at prices affordable by the smaller user. The user can normally configure the system to allow data to be stored on the main disk drive, a separate disk drive, or maximize file contents by trading disks in the main drive and storing data separately from the programs. Historical copies of files may be made to the tape drive, to a second 1541 Disk Drive or to drive 1 of the double drive unit. All programs contain FileGuard™ II, to protect the data files, even in the event of power failure during operation.

These systems are designed for use with the typical Commodore 64 configuration of equipment: the computer, a 1541 Disk Drive, a C2N Datasette Tape Drive, and a 1525 Printer. But some users will find they have different needs. These programs will support almost any Commodore disk drive unit or printer, plus the Corvus hard disk drives and almost any standard ASCII printer, assuming the user purchases suitable interfaces.

Some systems will communicate with each other for posting or updating of information. All files can be transmitted through a modem to another computer by using the MailBox™ package and the standard VIC Modem or any other suitably interfaced modem.

Contact: Southern Solutions, PO Box P, McKinney, TX 75069.□

Circle No. 73

LearningWare™ & FunWare™

The LearningWare™ series of programs has been designed specifically for the Commodore 64 and VIC-20 computers. All of the LearningWare™ programs will run either in the Commodore 64 or the unexpanded VIC-20, but memory expansion may be used. All programs are available on disk or tape.

The concept of the FunWare™ programs is strictly to entertain. These programs use the high quality sprite and sound synthesis capabilities of the Commodore 64 to give hours of pleasure to the whole family. Available either on disk or tape.

LearningWare™ is an educational set of programs ranging in subject and skill level, designed to supplement elementary, junior, and senior high school courses. Each program is user-friendly, in the Southern Solutions tradition, containing simple-to-read-and-understand prompts, and safeguarded against accidental program exits, yet forgiving in input spelling errors, when appropriate.

LearningWare™ is especially useful in drills, and other educational applications requiring repetition and memorization. Students are presented with the data in a manner that neither intimidates,when the respondent is incorrect, nor discourages, when the completed drill test score falls below passing.

Where appropriate, LearningWare™ scores each completed drill, with both percentage correct, and letter grade. Provision is made for altering the letter grades to fit the user's grading system, which is explained in the program documentation. All programs also display the number of questions attempted, allowing the drill supervisor to consider the percentage relative to the students chosen number of questions.

Contact: Southern Solutions, PO Box P, McKinney, TX 75069.□

Circle No. 74

Creative Software Wants You to "Save New York!"

Creative Software is pleased to announce the release of "Save New York!", a game cartridge for the Commodore 64, designed and written by Joe Jetson. This game will be available June 1.

New York, more than most towns, has mutants...in the buildings, in the sewer system, on the streets and especially in the subways. You, defender of all that is good, must destroy all of the above ground and subterranean mutants before they destroy New York.

The hardest part of "Save New York!" is trying to figure out why you would want to Save New York.

Creative Software, is located at 230 East Caribbean Drive, Sunnyvale, CA 94086. (408) 745-1655.□ Circle No. 75

Video Activated Power Switch for Remote Video Monitors

A low cost video activated power switch to turn-on and turn-off video monitors, is now available. The switch detects horizontal SYNC from the video signal to turn the monitor, or other device on. In the absence of a video signal, the monitor or other device turns off. Advantages of the remote switch are: Saves electrical

energy by using monitor display only when video signal is present; reduces monitor maintenance; eliminates distractions from free-running monitors; eliminates the need for special AC power lines or tall ladders to turn video monitors on/off in unusual locations.

Model VSP-I—$99, mounts inside the monitor; Model VSP—IA—$149, used external to the monitor. The monitor power cord plugs into the VSP—IA, and the VSP-IA plugs into a source of AC power.

Telemax, INC. 780 Lorraine Drive, Box 339, Warrington, PA 18976, (215) 343-3000.□ Circle No 76

IEEE-488 64 Digital Channel Data Acquisition Input Module

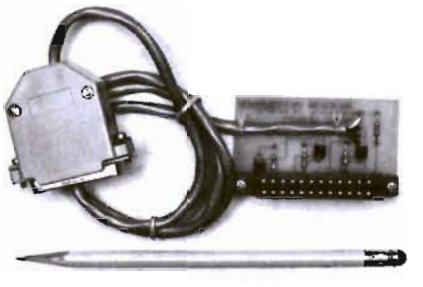
Connecticut microComputer announces a new 64 digital line input module which is a self contained IEEE 488 (GPIB) bus compatible device. It is the first product in the BUSSter series of I/O modules.

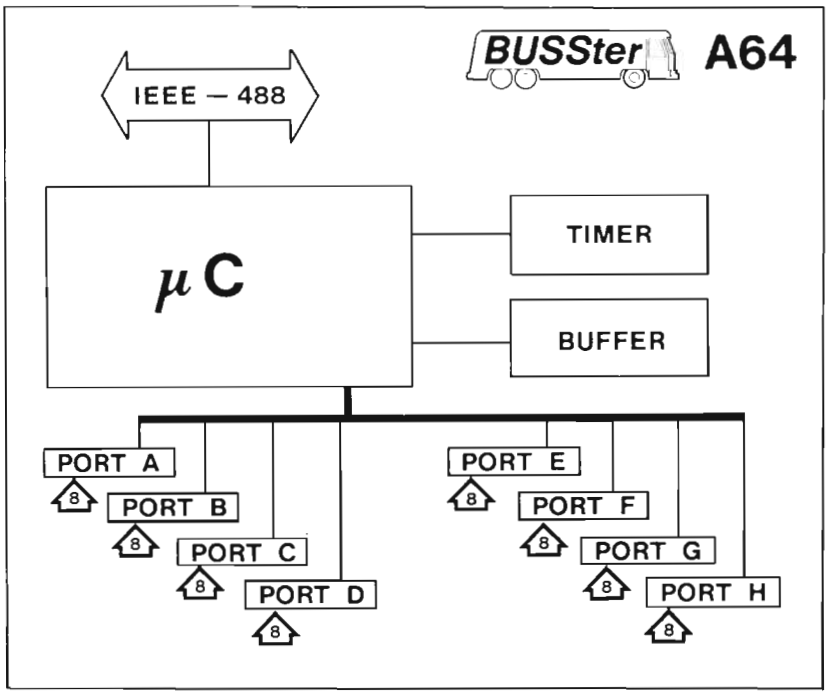
The BUSSter A64 Digital Input Module accepts commands from any host computer through its IEEE port, reads and stores data from up to 64 digital TTL level lines and then sends this information back to the computer. A BUSSter module economically increases a computer's interfacing capability while reducing its workload.

The BUSSter A64 Digital Input Module is easily programmed through BASIC commands from the controlling computer. The built-in timer and buffer allows data sampling and collection to occur, while the host computer is occupied with other tasks.

The BUSSter A64 Digital Input Module sells for $495.00 in standard version, including case and power supply, and is available from stock.

Connecticut microComputer, 36 Del Mar Drive, Brookfield, CT 06804, 203 775-4595.□ Circle No 78





IEEE-488 64 DIGITAL CHANNEL DATA ACQUISITION INPUT MODULE

- **64 Digital Inputs**
- **Built in timer**
- **Buffered**
- **2048 Byte Buffer Optional**

Modem Interface—Cost-saving convenience

The MODEM INTERFACE offers VIC-20 and C64 users the cost-saving convenience of connecting virtually any modem to their computer. The Modem Interface is compatible with the Microconnection, Smartmodem, CAT, etc. A 3-foot cable with a standard DB25 connector is supplied at no extra charge. The Modem Interface allows use of your modem's autodial/autoanswer features. Included with the Modem Interface is a free autodial terminal program. The cost is only $21.95 postpaid from Bytesize Micro Technology, PO Box 21123, Dept. GC, Seattle, WA 98111, 206 236-BYTE. Include $1.75 extra for shipping outside the US, Canada, or Mexico. VISA, Mastercard, and COD customers can call (1-800) 227-3000 toll free or (1-800) 792-0990 in CA ask for operator 225.□ Circle No 77

CeeNet

Introduction—The CeeNet is a coordinated software and hardware package which allows up to 64 Commodore VIC's and CBM 64's to be interconnected. Devices, such as disk drives, printers, and modems can be shared by all microcomputers connected to the network. In addition, library of application programs provide powerful network functions such as inter-network mail, terminal linking, file protection, and database support.

Features and Implementation—The CeeNet has been designed to be fully *extensible*. That is, the system can be easily upgraded at any time. This allows users to 'start simple' and let the system 'grow' as their needs require. The sections below describe range of systems, starting from a simple system. Each network consists of a

number of *workstations* (Computers for use by students and teachers) and *Shared Devices* (Computers which provide shared resources such as a disk drive or printer. Each shared resource requires one computer). The total number of computers can not exceed 64.

The beginning system consists of either VIC's or CBM 64's. A shared disk drive provides users with a control library of programs. In addition, an application program, ADVISE, allows teachers to 'link' their computer to the student's computer. This allows the teacher to take control over the student's computer. Users may also send messages to other users via the MESSAGE command. This configuration requires each microcomputer to have a network interface card; there is no other addition to each microcomputer. The network software is loaded into each computer at the start of each day. This system is ideal for applications which require little programming but require that many computers have access to a common set of programs (i.e. Computer aided instruction). Availability—December 1982.

The advanced beginning system includes the VIC-TREE programming cartridge. This cartridge includes all network software in addition to a large set of commands which ease BASIC programming, editing, and de-bugging. In addition, users may add a shared printer.

The intermediate system requires a CBM 64 to operate the shared disk drive. This system provides a sophisticated shared disk which supports protected individual user accounts, group accounts for file sharing, database support, faster access to disk files, and a mail facility. This system is intended for high schools which intend to teach programming, word processing, and accounting. Availability—Spring 1983.

The advanced system provides a *gateway*. A gateway is a method by which many CeeNet's can be interconnected via the telephone lines. The gateway requires one CBM 64 computer and a modem. Up to 255 CeeNet's can be interconnected. This allows users to transfer files between networks. A supplied application program provides file transfer operations, inter-network mail, and the capability to connect to large time-shared computers such as the SOURCE. Availability—Summer 1983.

Skyles Electric Works, 231E South Whisman Road, Mountain View, CA 94041, (415) 965-1735. Circle No. 79

Word Puzzles Challenge Fans of All Ages

Crypto-Cube, a new educational computer game which provides word puzzle enthusiasts with exciting, imaginative challenges is now available from DesignWare for Apple, Atari and IBM microcomputers.

Designed for one or two players, Crypto-Cube features a cube which rotates, each side exposing a grid similar to that found in a crossword puzzle. Players take turns uncovering letters to fill in the missing words.

Crypto-Cube encourages kids to expand their vocabulary as well as practice their spelling. They can also use the computer to create their own puzzles on the computer.

Prior to the start of each game, players pick one of the 50 possible puzzles included on the Crypto Disk, or create their own word list for placement on the cube by the computer's puzzle generator. These puzzles can be saved on their own disk.

"This game is a word puzzle fan's dream," said James Schuyler, president of DesignWare, Inc. "It's lively, with a lot of animation and sound features. It stimulates children's interest in words, helps them with spelling and gives them keyboard familiarity. For adults, it's a real mind-bender, on a par with the best board games available today."

Aimed at children eight years and over, Crypto-Cube will be available in quality computer and software stores nationwide in early May. Retailing for $39.95, the new software package runs on the Apple II Plus and II3, IBM PC, Atari 400, 800 and 1200 XL with 48K memory and disk drive.

Based in San Francisco, CA, DesignWare, Inc., develops and distributes a full line of educational software.

Contact: 1901 Avenue of the Stars, Los Angeles, CA 90067, (213) 557-1331. Circle No. 80

DesignWare Makes Learning Words Fun with Spellicopter Game

Spellicopter, an action-packed computer-based spelling game for personal computers has been announced by DesignWare, Inc., a leading developer of educational software for personal computers.

In Spellicopter, the pilot must accomplish a dangerous rescue mission. The task requires keen memory, verbal, spelling and navigation skills. The goal: To rescue letters "stranded" on a distant field, organize them into a correctly spelled word and bring the cargo back to the helicopter's landing pad.

DesignWare's new game involves the player in a series of actions that interrelate. Prior to takeoff, the "pilot" is shown a list of 20 words. One by one the pilot will try to rescue these words from the field of letters.

As the flyer readies his helicopter, he is shown a sentence with the target word omitted. With all systems "Go", the pilot must fly through crowded skies and mountainous terrain. When the flyer reaches the field, he picks up the letters by focusing a laser beam on them. Correctly spelled words are carried back to the landing pad in the cargo bay of the helicopter.

The pilot must pass other aircraft and obstacles which threaten his flight path. If he does not maneuver quickly to avoid these obstacles, his helicopter will run out of fuel and crash.

"The idea behind Spellicopter is to provide a chance for both children and adults to exercise their own personal 'spelling demons' in a motivating environment in which we have combined hand-eye coordination with a pedagogically sound educational procedure," explains Jim Schuyler, president of DesignWare.

Available on the Apple II + and IIe, Atari 400, 800 and 1200 XL, and IBM PC with 48K memory and disk drive. Spellicopter retails for $39.95. The new game will be available in early May.

Contact: DesignWare, 1901 Avenue of the Stars, Los Angeles, CA 90067, (213) 557-1331.□ Circle No. 81

A Friendly Companion For You and Your VIC-20!

Hello! My name is PAL and I am here to help you, the VIC-20 programmer, with a wide assortment of quick and easy aids for programming the VIC-20. P-A-L stands for Programmer's Aids and Logs and I hope to become a real 'pal' to you as well!

Whether you are just beginning to learn programming on the VIC-20, or are a real expert, I'm sure you will find me a great help. My aids and logs are patterned after those that professional programmers use. But of course, they are designed just for you; the VIC-20 programmer!

PAL is not a how-to-program guide, nor is it a technical manual. PAL's aids and logs are merely a collection of useful VIC-20 information and worksheets in a very easy to use form.

And, PAL contains the most complete collection of VIC-20 aids and logs to be found; ranging from simplified BASIC definitions to screen graphic design worksheets. Anyone wishing to write better programs, more quickly, without the frustration of hard to use reference materials will find using PAL invaluable.

PAL is designed to help you locate that particular piece of programming information you need very quickly. PAL's aids place commonly used information in a quick and easy to use form. No more page flipping thru the manuals! The logs help you organize all your other VIC-20 information, so that you may easily locate it as needed.

PAL helps you write better programs. You'll be more organized, have quicker access to VIC information, and your programs will be better designed and documented. PAL can help you to form a mental image of what you would like your program to do, and how it should appear. This is most important for writing good programs. A program that is not well thought out will never be as good as one that is!

One look through PAL'S pages and you'll quickly see how they will benefit you and your programs. PAL is conveniently printed on a tear-out pad, is color-coded for easy reference, and has been punched for a standard 3-ring binder.

Programming the VIC-20 is a very enjoyable past-time, and very educational as well! PAL is here to help you write better programs, keep organized, and above all, have fun!

Look what PAL contains...

Border and screen full-color combination rainbow; EZ-Key quick guide to all keys and characters. Includes: pokes, reverses, CHR$, set 1 - set 2; EZ Note sound music chart and worksheets; BASIC-LY EZ condensed basic dictionary; Create-A-Character programmable characters worksheets; EZ Screen tearout screen layout and design forms; EZ Graph graphics programming aid; Doc-U-Ment program documentation worksheets; EZ Flow program flow charting worksheets; General Purpose programming worksheets; Software Listing Log Sheets; Tricks and Hints Log Sheets; Tape Cassette Log Book.

And Even More...

Basic-Aid* Quick Reference Card to basic keywords and important memory locations; Function-Aid* Templates for programmable f1-f8 keys. (*These two mount right on your VIC-20!)

95 pages of aids, worksheets and logs. Something for every VIC-20 programmer!

Data Equip. Supply Corp. Data Tape, 8315 East Firestone Blvd., Downey, CA 90241, (213) 923-9361.☐

Circle No. 82

Skyles Electric Works Presents The VicTree®

The VicTree leaves your new VIC (or CBM 64) with 42 additional commands, branches out to most BASIC 4.0 programs and roots into most printers.

The VicTree is a coordinated hardware and software package that allows your VIC to branch out in unbelievable directions and makes it easier than ever to do BASIC programming, debugging and to access your disk. And the new VicTree provides routines to interface the VIC to the powerful CeeNet local network. 8kb of ROM—4kb for the BASIC commands, 4kb for disk commands and interfacing to CeeNet—plus 4kb of RAM for miscellaneous storage. Perfect not only for the new VIC but also for the Commodore 64. Unbelievably simple to use and to install, the VicTree gives you all the additional BASIC 4.0 commands to allow most BASIC 4.0 programs to work on your new VIC or CBM 64.

Now only $89.95—or $109.95 complete with Centronics standard printer cable. (Cable alone $29.95) Available now from your local dealer or order through your Visa or MasterCard toll free: (800) 227-9998 (CA, Canada, AK, HI, please call (415) 965-1735) or send check or money order directly to: Skyles Electric Works, 231E South Whisman Road, Mountain View, CA 94041. (415) 965-1735.☐ Circle No 83

BUSIWRITER Now Available

Why word processors? Word processors allow the user to quickly and easily create letters, memos, notes, reports, term papers, manuals, poetry and any other written information using the memory of the computer as a pencil and paper. The computer display or terminal acts as a window through which the user views the information as it is entered. The outstanding advantage of using BUSIWRITER is that it acts not only as a pencil and paper but as a perfect eraser and automatic typewriter.

The Queen Bee of Word Processors

BUSIWRITER allows the user to quickly and easily make any number of alterations to the text. BUSIWRITER will instantly reformat your text and show you exactly and continuously how the final output will appear. BUSIWRITER has more functions than any other known microcomputer word processor. With BUSIWRITER assisting in the entry of text, providing a 20 page memory and performing an enormous number of editing/composing functions, the preparation of written data is far faster and outstandingly more accurate than if it were prepared by hand.

For use with Commodore CBM 64, 1515, 1525, Epson, C. Itoh, Qume, Diablo, NEC Spinwriter, Starwriter, Prowriter, Okidata, Microline, Gemini-10, and many more printers.

BUSIWRITER 64—only $99 for the CBM 64. Now available from your local dealer. Call (800) 227-9998 for the name of your nearest dealer. (CA, Canada, AK and HI, please call (415) 965-1735.) Skyles Electric Works, 231G South Whisman Road, Mountain View, CA 94041.☐ Circle No. 84

Arrow

Six times faster cassette operation. That's right! An inexpensive plug in module for the Commodore VIC-20 or CBM-64 that allows 6 to 7 times faster LOAD, SAVE and VERIFY of BASIC and machine language programs. In addition, we have shortened the "leader" preceding the program to 4 seconds as a further speed up of cassette operations.

And that's not all. It was such fun speeding up the LOAD, SAVE, and VERIFY commands that we added a new command ... APPEND to enable you to add one BASIC program to another. We also added a command to move the tape at "Fast Forward" speeds to one of nine specified 16,000 byte areas on the tape. We also added error detection to increase the reliability of cassette operations.

Easy to use. ARROW—adds 4 commands; S, L, V, and A that are used to SAVE, LOAD, VERIFY and APPEND at a **high speed** (3600 baud).

ARROW—leaves the usual messages that appear on the screen.

ARROW—adds a message telling you the length of the program that has been loaded (or saved).

ARROW—keeps all of the usual short hand features of program naming.

ARROW—starts automatically when you turn on your VIC-20 or CBM-64.

Programming Aids

ARROW—Adds the following programming aids: AUTOLINE numbering, RENUMBERING of lines, DELETE a line or group of lines, SEARCH with optional replace, MOVE a block of memory, COMPARE two blocks of memory, HEXADECIMAL/DECIMAL calculator converter, XIT two machine language monitor with; Display, Go, Register, Save, Load and Exit commands.

Prices Available

AR20 (VIC-20), AR64 (CBM-64)—$49.95; ARAS20 (VIC-20), ARAS64 (CBM 64)—$89.95. Call your local dealer or contact Skyles Electric Works, 231E South Whisman Road, Mountain View, CA 94041 (415) 965-1735.☐ Circle No. 85

Computer Cases

All COMP-CASES by COMPUTER CASE COMPANY are constructed of the highest quality luggage material with hard sides, padded handles, brass hardware and key locks. Rubber pads provide furniture protection, and steel lugs on the bottom protect the case when transporting. The outside is covered in the highest quality scuff resistant textured vinyl in rich brown. The tops are easily removed so that the equipment can be operated without removal from the case. Provisions are made for cords to exit the case even when the top is on and locked, to provide convenient security without the need to disconnect electrical cords and cables. Storage space is provided for manuals, cords, working papers, and supplies within each case.

COMP-CASES are available for a wide variety of computer equipment and peripherals. Additional cases are being developed as the need arises. Special requests will be appreciated.

You can be confident that a COMP-CASE will meet your every requirement for quality, protection, security, and portability in an attractive case you can be proud of—and we guarantee it.

Computer Case Company, 5650 Indian Mound Court, Columbus, OH 43213, (614) 868-9464.☐ Circle No. 86

Commander Dealers

Given here, in zip code order, is a partial list of the Charter 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 information, hardware, or software for their Commodore Computers.

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
Pomton 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 Street, 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
Mallone, 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

Dealers, continued

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

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 Doidner

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
Street 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

Michigan

Micro Station Inc.
24484 West 10 Mile Road
Southfield, MI 48034
(313) 358-5820
Manager-Owner. Jerry Goldberg

Comm Data
320 Summit
Milford, MI 48042
(313) 685-0113
Owner: Larry Jones

Roseville Computer
25929 Gratiot
Roseville, MI 48066
(313) 772-0760
Manager-Owner. Tom Potter

Allen Park Computer Center
7000 Roosevelt, Suite 109
Allen Park, MI 48101
(313) 383-8254
Manager-Owner: Sam Noble

West Side Radio
7521 Wyoming
Dearborn, MI 48125

Ye Old Computer Shoppe
518 W. Cross St.
Ypsilanti, MI 48197
Manager/Owner: Dr. Donald Buckeye

Haney's Computer Center
15270 Gratiot
Detroit, MI 48205
(313) 839-1850
Manager-Owner: Paul M. Paul

Computer Mart
915 South Dort Highway
Flint, MI 48503
(313) 234-0161
Manager-Owner: Pat McCollem

The Computer Source
3205 S. Cedar St.
Lansing, MI 48910
Manager/Owner: Edson Borges

The Computer Connection
444 West Maumer Street
Adrian, MI 49221
(517) 265-7872
Manager-Owner: Ron Gamling

Newman A/V Communications, Inc
400-32nd Street SE
Grand Rapids, MI 49508
(616) 243-3300
Manager: Nancy Isler

Computers and More
2915 Breton SE
Grand Rapids, MI 49508.
(616) 243-3520
Manager-Owner: Bill Slaughter

Computer Tutor
502 East Front
Traverse City, MI 49684
(616) 941-5320
Manager-Owner: Caroline Garrick

Iowa

Micro Computer Applications
111 East Church Street
Marshalltown, IA 50158
(515) 752-8845
Manager-Owner: Harold Montover

Gronert Computers, Ltd.
4505 Forest Avenue
Des Moines, IA 50311
(515) 255-0618
Manager-Owner: Frank Gronert

Citizens Mutual Telephone
114 W. Jefferson
Bloomfield, IA 52537
Manager/Owner:WT Wallace

Cosmos Computers
1721 Grant Street
Bettendorf, IA 52722
(319) 355-2641
Manager-Owner: Paul Rung

Wisconsin
Starting Computers
4020 North 128th
Brookfield, WI 53005

3S Computer Center, Inc.
103 E. Silver Spring Dr.
Milwaukee, WI 53217
Manager/Owner: Alex Levitsky

Majic Business Systems
3519 West Wanda Avenue
Milwaukee, WI 53221
(414) 282-8072
Manager-Owner: Dennis Woitekaitis

Computerland of Madison
6625 Odana Road
Madison, WI 53719
(608) 833-8900
Manager-Owner: James Sullivan

Minnesota
P.M. Business Services
4400 Arden View Court
Arden Hills, MN 55112

South Dakota
Computerland Rapid City
738 Street John Street
Rapid City, SD 57701
(605) 348-5384
Manager-Owner: John Mattson

Illinois
The Software Store, Inc.
1767 Glenview Road
Glenview, IL 60025
(312)724-7730
Manager-Owner: David Pokvitis and Jeff Rayer

Digital World
711 Army Trail Road
Addison, IL 60101
(312) 628-9222
Manager-Owner: Sam Gunda

The Computer Store
11004 S. Cicero Ave.
Oak Lawn, IL 60453
Manager/Owner: Gerald Henery

B-A Computer Systems
2 North Batavia Avenue
Batavia, IL 60510
(312) 879-2350
Manager-Owner: Robert Appel

Softwareland, Inc.
420 West 75th Street
Downers Grove, IL 60516
(312) 852-6340
Manager-Owner: Maureen Quinn

Software Emporium Limited
175 North Franklin St.
Chicago, IL 60606
Manager/Owner: Reggie Miller

Rozel Industries, Inc
7360 North Lincoln Avenue
Lincolnwood, IL 60646
(312) 675-8960
Manager-Owner. Fred Whitlock and Becky Kowalsky

Fisher Scientific
4901 West Lemoyne Avenue
Chicago, IL 60651
(312) 378-7770
Manager-Owner: A.C Heidrich

Cambridge Business Systems
3345 N. Halsted
Chicago, IL 60657
Manager/Owner: Jeff Angsten & Ralph Samek

Micro Computers Plus
349 E. Main St.
Galesburg, IL 61401
Manager/Owner: Mike Henner

Kappel's Computer Store
125 East Main
Belleville, IL 62220
(618) 277-2354
Manager-Owner: Tom Kappel

Data Plus, Inc.
1706 Broadway
Quincy, IL 62301
(217) 222-65602
Manager-Owner. James Moore

Computer Tutor
1410 S. MaCarthur
Springfield, IL 62704
Manager/Owner: Bill McDannell

Missouri
Micro Age Computer
11413 Olive Blvd.
St. Louis, MO 63141
(314) 567-7644
Manager-Owner: Norm Fishel

Common Wealth Computers
5214 Blue Ridge Blvd.
Kansas City, MO 64133
(816) 356-6502
Manager-Owner: Dick York

Sigma Plus Computers
1804 Vandiver Drive
Columbia, MO 65205

Kansas
Kansas Micro Computer
1601 W. 23rd
Lawrence, KS 66044
Manager/Owner: John Ellena

Palmer News, Inc.
1050 Republican St.
Topeka, KS 66601
Manager/Owner.

Computer Business Machines
Officenter 357 South Lulu
Wichita, KS 67211
(316) 267-1150
Manager-Owner. Mrs. R. Santoscoy

Compusense
1001 South Washington
Wichita, KS 67211
Manager-Owner: J. Kendrick

Nebraska
Hobby Town
220 North 66th Street
Lincoln, NE 68505

Central Office Equipment
2020 Central Avenue
Kearney, NE 68847
(308) 234-2515
Manager-Owner. Byron Hansen

Louisiana
The Computer Center
111 C Rena Drive
Lafayette, LA 70503
(318) 988-2478
Manager-Owner: Robert Jones

Computer S.O.S.
4436 Youree Drive
Shreveport, LA 71105

Texas
Casual Computers
15340 Dallas Parkway Suite 2108
Dallas, TX 74248

Taylor Computer Systems
949 Melbourne Road
Hurst, TX 76503
(817) 284-5251
Manager-Owner. Mike Taylor

Computer Home
431 East Avenue C
San Angelo, TX 76903
(915) 653-7488
Manager-Owner: Brent DeMoville

Texas Technical Services
3115 West Loop South. #26
Houston, TX 77027
(713) 965-9977
Manager-Owner Phil Ray

I.T.S.
420 Plantation Drive
Lake Jackson, TX 77566
(713) 297-9016
Manager-Owner: I.K Kelly, Jr.

The Computer Experience
125 Southbridge
San Antonio, TX 78217
(512) 340-2901
Manager-Owner: Carolyn Roberts

Valley Computer Systems, Inc.
1101 North Cage, Suite A1
Pharr, TX 78577

Software-N-Things
2141 West Anderson Lane
Austin, TX 78757
(512) 451-4347
Manager-Owner: John Krieg

Dealers, continued

Professional Computer Associates
5326 Cameron
Austin, TX 78723
(512) 459-1220
Manager-Owner: Steve Derosa

Computerland of Amarillo
2300 Bell Street
Amarillo, TX 79106
(806) 353-7482
Manager-Owner: Mark Trowbridge

Colorado

Whole Life Distributors
965 Washington Street #6
Denver, CO 80203
(303) 861-2825
Manager-Owner: Tom Tarbart

Zero Page, Inc.
2380 Naegele Road
Colorado Springs, CO 80904
(303) 633-0211
Manager-Owner: David C. Cooper

Idaho

Electronic Specialties, Inc.
8411 Fairview Avenue
Boise, ID 83704
(208) 376-5040
Manager-Owner: Terry Romero

Utah

Computer Plus
1078 East Fort Union Blvd
Midvale, UT 84047
(801) 566-3902
Manager-Owner: Steve Whitzelor and Allen Vincent

The Hi-Fi Shop
2236 Washington Blvd.
Ogden, UT 84401
(801) 621-5244
Manager-Owner: Brent Richardson

Central Utah Electronics Supply
735 South State
Provo, UT 84601
(801) 373-7522
Manager-Owner: George S Moore

Arizona

Computer Super Store
4001 East Thomas Road
Phoenix, AZ 85108
Manager-Owner: Richard Sarhan

Personal Computer Place
1840 West Southern Avenue
Mesa, AZ 85202
(602) 833-8949
Manager-Owner: Roger Smith

Computer Depot
1201 Iron Springs Road
Prescott, AZ 86301
(602) 778-7473
Manager-Owner. Brice Eldridge

New Mexico

Computer Super Store
1660 Eubank NE
Albuquerque, NM 87112
Manager-Owner: Richard Saham

Nevada

PCS Computer
3900 West Charleston, Ste R
Las Vegas, NV 89102
(702) 870-4138
Manager-Owner: Mickey Cole

Harry's Business Machines, Inc.
323 West Street
Reno, NV 89501
(702) 322-4559
Manager-Owner: Gordon Foote

California

Opamp Tech Books
1033 North Sycamore
Los Angeles, CA 90038
(213) 464-4322
Manager-Owner: Alicion

Data Equipment Supply Corp.
8315 Firestone Blvd.
Downey, CA 90241
(213) 923-9361
Manager: Robert Johnson

Computer Place
23914 Crenshaw Blvd.
Torrance, CA 90505
(213) 325-4754
Manager-Owner: Wen T. Huang

Fyrst Byte
10053 Whittwood Drive
Whittier, CA 90603
(213) 947-9411
Manager-Owner: Darrell Miller

Game Room
5675 Kanan Road
Agoura, CA 91301
(213) 707-0142
Manager-Owner: Jean Collier

General Computer Store
22323 Sherman Way Unit #7
Canoga Park, CA 91303
(213) 704-6600
Manager-Owner: Anita Broadway

HW Electronics
19511 Business Center Drive
North Ridge, CA 91324
(213) 886-9200
Manager-Owner. Ronda

Levity Distributers
6857½ Ben Avenue
North Hollywood, CA 91605
(213) 982-2514
Manager-Owner: Melinda Plesha

Data Systems West
421 West Las Tunas Drive
San Gabriel, CA 91776
(213) 289-3791
Owner: Frank J. Mogavero

Consumer Computers
8314 Parkway Drive
La Mesa, CA 92041
(714) 465-8888
Manager: Steve Scott

Calco Digital Equpiment Inc.
1919 Aple Street
Oceanside, CA 92054
(714) 433-4119
Vice President: Ronald N. Paperno

20-64 Software Center
9829 Mira-Mesa Blvd.
San Diego, CA 92131
(619) 695-0214
Manager-Owner: Larry Skaggs

Computer Nook, Inc.
965 South E St
San Bernadino, CA 92408
Manager/Owner: Frank Bayless
James J. Schmidt

Inland Electro Mart
8624 California Avenue
Riverside, CA 92504
(714) 687-3776
Manager-Owner: Jack

PEC
1440 South State College, Suite 6F
Anaheim, CA 92806

Quality Computer Center
801 South Victoria Street, #104
Ventura, CA 93003
(805) 642-1979
Manager-Owner: David Stewart

Jay-Kern Electronics
1135 Columbus
Bakersfield, CA 93305
871-5800
Manager-Owner: Don Taylor

Micro Pacific Computer Center
5148 North Palm
Fresno, CA 93704
(209) 229-0101
Manager-Owner: Mike Reinhold

J. Snell & Co., Inc.
657 Mission Street
San Francisco, CA 94105
(415) 421-5898
Manager-Owner: James Snell

Staceys Bookstore
219 University Avenue
Palo Alto, CA 94301

PC Computers
10166 San Pablo Avenue
El Cerrito, CA 94503
(415) 527-6044
Manager-Owner: Gary Guttebo

Computer Ideas, Inc.
1029 Tennessee Street
Vallejo, CA 94590
(707) 552-5076
Manager-Owner: J. Gavin

Fox Computer Co.
2678 North Main Street
Walnut Creek, CA 94596
(415) 944-9277
Manager-Owner: Stan Nielson

Albany Typewriter & Computer
923 San Pablo Avenue
Albany, CA 94706
(415) 526-1959
Manager-Owner: Bill Tichy

Idea Computers
301 North Santa Cruz Avenue
Los Gatos, CA 95030
(408) 354-1210
Manager-Owner: Tom Wolf

The Computer Room
230 Mt. Herman Road
Scotts Valley, CA 95066
(408) 438-5001
Manager-Owner: Gary Guttebo

The Computer Center Stores
930 Town & Country Village
San Jose, CA 95128
(408) 246-5710
Manager-Owner: R. Reid,
J. Barlow and N. Kinney

Inland Electro Mart
8624 California Drive
Riverside, CA 95204
(714) 687-3776
Manager-Owner: Jack

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

Software Plus
6201 "C" Greenback Lane
Citrus Heights, CA 95610
(916) 726-4979
Manager Owner: Carolyn Webster

The Radio Place
2964 Freeport Blvd
Sacramento, CA 95818
(916) 441-7388
Manager-Owner: Gary Stilwell

Integral Computer Solutions
2947 Fulton Ave.
Sacramento, CA 95821
Manager/Owner: Mike Kouri

Ray Morgan Co.
554 Rio Lindo Avenue
Chico, CA 95926
(916) 343-6065
Manager: Dave Wegner

Computer Place
1698 Market Street
Redding, CA 96001
(916) 221-1312
Manager-Owner: John Fredricks

Radio Mart
1075 Cypress
Redding, CA 96001
(916) 241-3000
Manager-Owner: John Cokeley

Oregon

SW Computers
1125 NE 82nd
Portland, OR 97220
Manager-Owner: Jerry

Edu-Tech
1575 NW 9th
Corvallis, OR 97330
(503) 758-5577
Manager-Owner: L. Clark and
W. Brown

Libra Book Inc.
856 Olive Street
Eugene, OR 97401
Manager-Owner: Larry West

M-S-G
537 Willamette Court
Eugene, OR 97401
Manager/Owner: Bill Kennish

Midland Cascade
61 West 8th Avenue
Eugene, OR 97401
Manager-Owner: Ron Isaacs

Ace-Tec
Highway 101 South Computer Center
Bandon, OR 97411
(503) 347-9322
Manager-Owner: Ace Egnew

John's Newsland
130 E. Main
Medford, OR 97501
Manager/Owner: John Wikoff

Washington

Compu-Play
1320 South 324th, Suite A-3
Federal Way, WA 98003
(206) 839-4453
Owner: Kevin Mitchell

Electronic Supermart
7040 South 180th Street
Kent, WA 98032
(206) 251-8484
Manager-Owner: Richard Thorp

Data Borne Computers
641 SW 152nd
Seattle, WA 98166
Manager-Owner: Richard Kantak

Programs Plus
16874 Southcenter Parkway
Seattle, WA 98188
(206) 575-1375
Manager-Owner: Nick Smith

Computer World
909 SE Everett Mall Way
Everett, WA 98204
Manager/Owner: William Lauby

Conti Electronics Ltd.
c/o Afcon
140-14th
Blaine, WA 98230
Manager-Owner: G.W. Harder

Computer Corner
1610 North Laventure
Mt. Vernon, WA 98273
(206) 428-1840
Manager-Owner: Kirk D. Shroyer

Technique Data Systems Inc
3306 Hwy 101 East
Port Angeles, WA 98362
Manager-Owner: Loren Larson

Computer +
2504 Jefferson Avenue
Tacoma, WA 98402
(206) 272-2329
Manager-Owner: David Dodd

Computer Town
1215 Center
Tacoma, WA 98409
(206) 272-2271

Nibbles & Bytes
4020 South Steel Street, Suite 105
Tacoma, WA 98409
(206) 475-5938
Manager-Owner: John Clark

Reed Graphics Inc.
964 East 99th Street
Tacoma, WA 98445
Manager-Owner: Don Reed

The Electronics Shop
131 North Decator
Olympia, WA 98502
(206) 357-6304
Manager-Owner: Frank and
Tim Linaham

Bits, Bytes and Nibbles, Inc.
209 Northtown Shopping Center
Spokane, WA 99207
(509) 487-1601
Manager-Owner: Richard Shulman

Alpha Computer Center
1341 D. George Washington Way
Richland, WA 99352
Manager-Owner: John Freer

Alaska

BG Systems Co.
204 East International
Anchorage, AK 99502
(907) 276-2986
Manager-Owner: Robert DeLoach

Micro Age Computer Store
2440 Seward Highway
Anchorage, AK 99503
(907) 279-6688
Manager-Owner: Jay Wisthoff

Canada

British Columbia

Book Shoppe
33-10th Avenue South
Cranbrook, BC V1C 2M9
(604) 426-3415
Manager-Owner: Tommy Wheeler

Dynamic Computer Systems
6-1960 Springfield Road
Kelowna, BC V1Y 5V7
(604) 860-7795
Manager-Owner: Wayne Henricksor

Sight & Sound Computer Center
657 Central
Prince George, BC V2M 1C6
(604) 562-0356
Manager-Owner: Jim Condon

Vulcan Computer Systems
20571 Fraser Highway
Langley, BC V3A 4G4
(604) 530-8572
Manager-Owner: Gord Tremain

Datum Electronics Ltd
11-3000 Lougheed Highway
Port Coquitlam, BC V3B 1C5
(604) 464-4424
Manager-Owner: John Lim

Minitronics Coquitlam Center
1114-2929 Barnet Highway
Coquitlam, BC V3B 3R5
(604) 464-0633
Manager-Owner: Ross Watson

Village Computer Systems
17704-56th Avenue, #8
Surrey, BC V3S 1C7
(604) 576-8112
Manager-Owner: Ron Jasper

Minitronics
1440 Kingsway
Vancouver, BC V5N 2R5
(604) 872-3225
Manager-Owner: Bud Lam-Lim

Castle Data Systems Ltd.
670 East Broadway
Vancouver, BC V5T 1X6
(604) 872-2829
Manager-Owner: Ken Poole

Softcorner Developments
683 East Broadway
Vancouver, BC V5T 1X7
(604) 872-6601
Manager-Owner: Howard Schaefer

Conti Electronics
7204 Main Street
Vancouver, BC V5X 3J4
(604) 324-0505
Manager-Owner: Greg Harder

621 Reading Center
621 West Pender
Vancouver, BC V6B 1W7
(604) 684-3814
Manager-Owner: Wilson Yau

Dealers, continued

Conti Computer Systems
1216 West Broadway
Vancouver, BC V6H 1G3
(604) 734-0606
Manager-Owner: Duncan Fraser

Mall Book Bazaar
850 Granville Street
Vancouver, BC V6Z 1K3
(604) 687-2213
Manager-Owner: Mike McCord

Owens & Sons Cash Register
2101 Government Street
Victoria, BC V8T 4P2
(604) 383-7134
Manager-Owner: Robert Owen and Glen Gray

Owens & Sons Cash Register
245 Fraser Street
Nanaimo, BC V9R 5C2
(604) 753-4178
Manager-Owner: Wayne Wrigley

Dynamic Computer Center
3105-31st Avenue
Vernon, BC
(604) 860-7795
Manager-Owner: Chris Chambers

Minitronics Guildford Center
Surrey, BC
(604) 584-1033

Quebec

Systems Ormic Ltd.
909 DeBourgone
Ste-Foy, Quebec G1W 4S6
(418) 659-3858

Systems Ormic Ltd.
999 De Bouragogue
Ste-Foy, Quebec G1W 4S6
Manager-Owner: Yvon Labbee

Systems Ormic Ltd
133 Ste-Germain
Rimouski, Quebec G5L 4B6

Systems Ormic Ltd.
1500 Norte Dame
Trois-Rivieres, Quebec G9A 4X5

Caleq, Inc.
331 Sir Wilfred Laurier
St. Lambert, Quebec J4R 2L1
(514) 465-6055
Manager-Owner: Marcel Bourcier

Le Camelot
1191 Place Phillips Square
Montreal, Quebec H3B 3C9
(514) 861-5019
Manager-Owner: Francine Roseenthol

Ontario

Questar International, Inc.
7270 Woodbine Avenue
Markam, Ontario L3R 4B9
(416) 475-8044

Electronics 2001
5529 Yonge Street
Willowdale, Ontario M2N 5C3
(416) 223-8400

Lichtman's News Agency
34 Adelaide Street West
Toronto, Ontario M5H 1P6

House of Computers
368 Eglinton Avenue West
Toronto, Ontario M5N 1A2
(416) 482-4336

Arkon Electronics
409 Queens Street West
Toronto, Ontario M5V 2A5
(416) 593-9653
Manager-Owner: Paul Perrin

Computer Circuit Ltd.
733 Richmond Street
London, Ontario N6A 3H2
(519) 672-9370
Manager-Owner: Wally Soloton

Lichtman's News Agency
Bayview Village Shopping Mall

Lichtman's News Agency
1430 Yonge Street
Toronto, Ontario

Alberta

Soft Option
Box 112, Station T
Calgary, Alberta T2H 2G7
Manager-Owner: David Evans

Computer World
4921-49th Street
Red Deer, Atlanta T4N 1V2
(403) 347-4280
Manager-Owner: David Todd

Kelley Software Dist. Ltd.
PO Box 11932
Edmonton, Alberta T5J 3L1
Manager-Owner: Tom Kelly

T.J.B. Microsystems, Ltd.
10991-124th Street
Edmonton, Alberta T5M 0H9
(403) 451-6262
Manager-Owner: John Atwood and Doug Poon

T.J.B. Microsystems, Ltd.
10458-82nd Street
Edmonton, Alberta
(403) 433-3161
Manager-Owner: John Atwood

Electro-T's
8770-170th Street
Edmonton, Alberta T5T 3J7
(403) 487-0144
Manager-Owner: Gerry Rush

Dico Post #9
103-9709 Franklin Avenue
Fort McMurray, Alberta
(403) 791-6664
Manager-Owner: Pat Erler

Johnnies Appliances
Box 1010
Le Duc, Alberta T9E 2Y6
(403) 986-2738
Manager-Owner: Ron and Lee

Chinook Electronics
Box 585
Lethbridge, Alberta
(403) 320-1424
Manager-Owner: Wayne

Saskatchawan

Micro Shack, Ltd.
333 Park Street
Regina, Saskatchawan S4P 5B2
(306) 543-4079
Manager-Owner: Dave Claypool

Computer Time
240-22 Street East
Saskatoon, Saskatchawan S7K 0E8
(306) 664-3456
Manager-Owner: Paul and Russ

Micro Shack, Ltd.
607-45th Street West
Saskatoon, Saskatchawan S7L 5W5
(306) 244-6909
Manager-Owner: Ray Real

Nova Scotia

Atlantic News
5560 Morris Street West
Halifax, Nova Scotia B3J 1C2
(902) 429-5468
Manager-Owner: Pat Doherty

Prince Edward Island

Jeta Systems
33 Belmont Street
Box 2857
Charlottetown, PEI C1A 8C4
(902) 566-1059

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

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—All Commodore Users under 20 years of age.

SW Ohio VIC Users Club
659 Carthage Avenue
Cincinnati, OH 45215
Contact—Tom E. Harris
761-7510

Public Doman Inc.
5025 So. Rangeline Road
West Milton, OH
Contact—Bill Munch

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 (published monthly)
Interests—All uses of Commodore 64 Computer

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 Ave. SE
Auburn, Washington 98002
Contact—Ben Dunnington
(206) 939-0582
(206) 924-6992
Newsletter—Commodore 64 Magazine

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

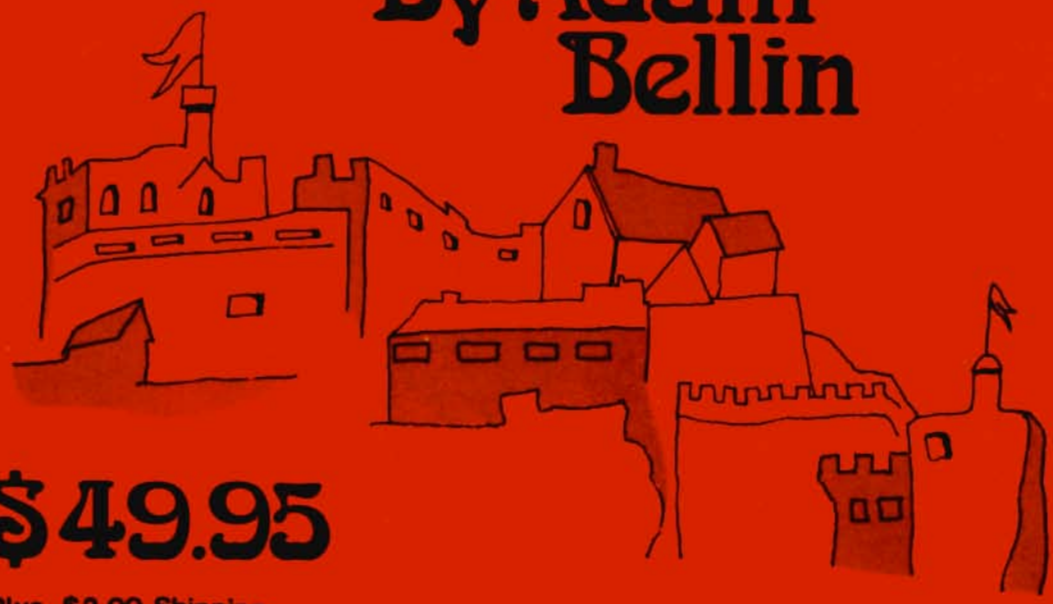
Central Washington Commodore User's CLub
1222 South 1st Street
Yakima, WA 98902
Contact—Bob Wood or Tim McElroy

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

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

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**

VICTORY SOFTWARE®

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

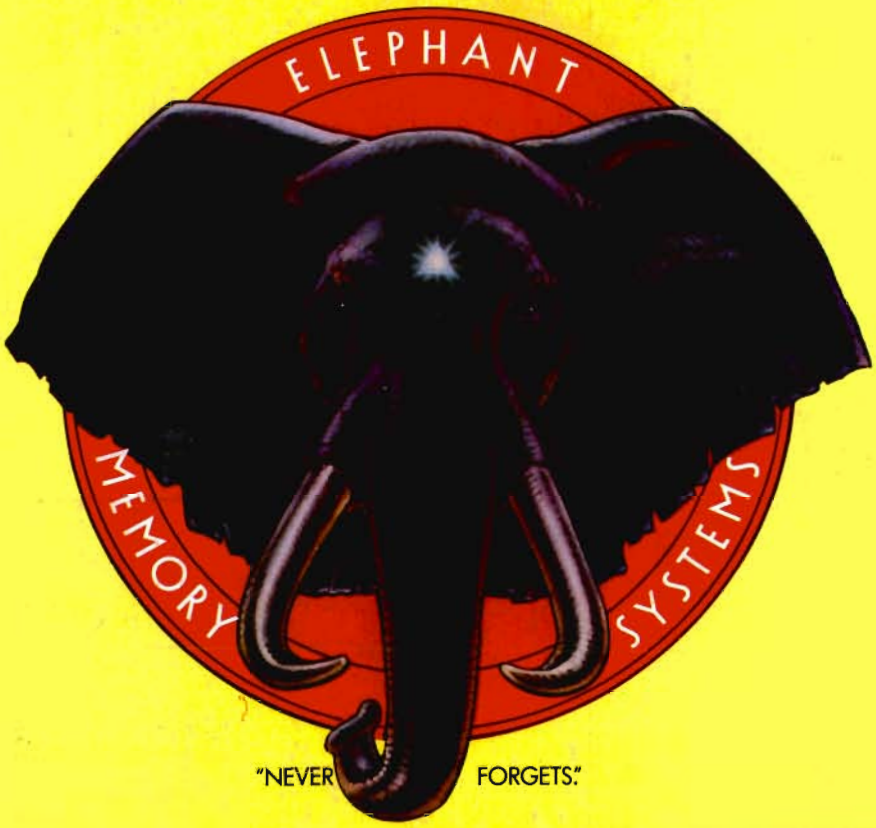
All programs fit in standard memory. All programs are on cassette tape.

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

REMEMBER:



MORE THAN JUST ANOTHER PRETTY FACE.

Says who? Says ANSI.

Specifically, subcommittee X3B8 of the American National Standards Institute (ANSI) says so. The fact is all Elephant™ floppies meet or exceed the specs required to meet or exceed all their standards.

But just who is "subcommittee X3B8" to issue such pronouncements?

They're a group of people representing a large, well-balanced cross section of disciplines—from academia, government agencies, and the computer industry. People from places like IBM, Hewlett-Packard, 3M, Lawrence Livermore Labs, The U.S. Department of Defense, Honeywell and The Association of Computer Programmers and Analysts. In short, it's a bunch of high-caliber nitpickers whose mission, it seems, in order to make better disks for consumers, is also to make life miserable for everyone in the disk-making business.

How? By gathering together periodically (often, one suspects, under the full moon) to concoct more and more rules to increase the quality of flexible disks. Their most recent rule book runs over 20 single-spaced pages—listing, and insisting upon—hundreds upon hundreds of standards a disk must meet in order to be blessed by ANSI. (And thereby be taken seriously by people who take disks seriously.)

In fact, if you'd like a copy of this formidable document, for free, just let us know and we'll send you one. Because once you know what it takes to make an Elephant for ANSI . . .

We think you'll want us to make some Elephants for you.

ELEPHANT.™ HEAVY DUTY DISKS.

For a free poster-size portrait of our powerful pachyderm, please write us.
Distributed Exclusively by Leading Edge Products, Inc., 225 Turnpike Street, Canton, Massachusetts 02021
Call: toll-free 1-800-343-6833; or in Massachusetts call collect (617) 828-8150. Telex 951-624.