

# TPUG

July 1984 \$2.95

*magazine*

The official publication for the world's largest international Commodore users group

The magazine for C-64,  
VIC 20, PET and SuperPET  
Users.

Computer aids for the disabled.

Documentation for  
(C)T9 and (C)M6



Canada Post Canada Postage paid For pay

Bulk En nombre  
third troisième  
class classe  
E12430  
Scarborough



# INTRODUCING



## THE PRO-LINE TEAM

★ **PAL 64**  
The fastest and easiest to use assembler for the Commodore 64. Pal 64 enables the user to perform assembly language programming using the standard MOS mnemonics. **\$69.95**

★ **POWER 64**  
Is an absolutely indispensable aid to the programmer using Commodore 64 BASIC. Power 64 turbo-charges resident BASIC with dozens of new super useful commands like MERGE, UNDO, TEST and DISK as well as all the old standbys such as RENUM and SEARCH & REPLACE. Includes MorePower 64. **\$69.95**

★ **TOOL BOX 64**  
Is the ultimate programmer's utility package. Includes Pal 64 assembler and Power 64 BASIC soup-up kit all together in one fully integrated and economical package. **\$129.95**

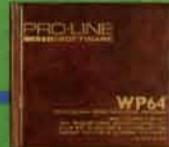
★ **SPELLPRO 64**  
Is an easy to use spelling checker with a standard dictionary expandable to 25,000 words. SpellPro 64 quickly adapts itself to your personal vocabulary and business jargon allowing you to add and delete words to/from the dictionary, edit documents to correct unrecognized words and output lists of unrecognized words to printer or screen. SpellPro 64 was designed to work with the WordPro Series and other wordprocessing programs using the WordPro file format. **\$69.95**

★ **WP64**  
This brand new offering from the originators of the WordPro Series\* brings professional wordprocessing to the Commodore 64 for the first time. Two years under development, WP64 features 100% proportional printing capability as well as 40/80 column display, automatic word wrap, two column printing, alternate paging for headers & footers, four way scrolling, extra text area and a brand new 'OOPS' buffer that magically brings back text deleted in error. All you ever dreamed of in a wordprocessor program, WP64 sets a new high standard for the software industry to meet. **\$69.95**

★ **MAILPRO 64**  
A new generation of data organizer and list manager, MailPro 64 is the easiest of all to learn and use. Handles up to 4,000 records on one disk, prints multiple labels across, does minor text editing i.e. setting up invoices. Best of all, MailPro 64 resides entirely within memory so you don't have to constantly juggle disks like you must with other data base managers for the Commodore 64. **\$69.95**

**NOW SHIPPING!!!**

**For Your Nearest Dealer  
Call  
(416) 273-6350**



†Commodore 64 and Commodore are trademarks of Commodore Business Machines Inc.

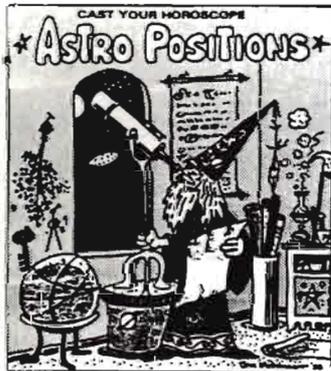
\*Presently marketed by Professional Software Inc.

Specifications subject to change without notice...

**PRO-LINE SOFTWARE**

**(416) 273-6350**

755 THE QUEENSWAY EAST, UNIT 8,  
MISSISSAUGA, ONTARIO, CANADA, L4Y 4C5



### ASTRO POSITIONS

- display and print the position of the sun and stars (one arc second accuracy) and the position of the moon and the planets (one arc minute accuracy except for Pluto) for any date in history.
- includes *User's Guide, Introduction to Positional Astronomy and Introduction to Classical Astrology.*

\$43.95



### The Banker

THE BANKER is one of the most powerful chequebook management systems available for the C-64. It is menu-driven and extremely easy to use. You can enter either cheques or deposits with full comments. All you do is fill out the cheque graphic which appears on the screen. Because THE BANKER has a category field, you can use it for your business accounts payable or home budgeting.

\$34.95



### DAISY

Database Information System

- one of the comprehensive DATA-BASE SYSTEMS for the 64
- fully bilingual on screen
- calculates; compute the contents of numeric fields - add, subtract, multiply, or divide against the defined field, using either constant value of the contents of any other field in the record.

\$39.95

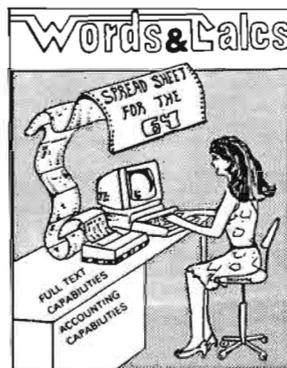


### EDIT/ASM II

- DESIGNED TO HELP YOU CREATE AND MODIFY 6502 Assembly Language Programs on the Commodore 64 Computer.

ON SCREEN HELP SYSTEM to assist you in using all commands available... add, assemble, calculate, delete, directory, disk, edit, find, help, insert, list, load, merge, new, plist, quit, and save.

\$36.95



### WORDS & CALC

Powerful household finance - Business system

- eliminate inconvenient files, confusing tables and tedious calculations.
- store all your data on WORDS & CALCS spreadsheet
- WORDS & CALCS will do your work for you in organizing, displaying, storing, printing and performing calculations

\$42.95 disk

## LOOK AT THE LANGUAGES WE HAVE

YES! We have PASCAL \$52.95

ULTRABASIC with turtle graphics and sound \$42.95

TINY BASIC COMPILER \$22.95

TINY FORTH Fig Forth implementation \$22.95

**Yes! You CAN get quality software at reasonable prices - order today**

money order  
cheque  
visa or master charge

The SMART 64 Terminal designed with Quality-Bred features, Affordable Pricing... And Service.

**\$42.95**

So why not travel the communications highways the SMART way!

Accessories included: Selective Storage of Received Data, Sends/Receives Programs and Files of ANY SIZE, Disk Wedge Built-In, Review, Rearrange, Print Files, Formatted Lines, Alarm Timer, Screen Print.

Dealer Inquiries invited.

Why not cash in on the good programs that you have written? We are actively seeking SOFTWARE AUTHORS. We are paying above-average royalties for all programs that we accept. Submit your copy on tape or disk, for VIC 20 or C-64, with detailed operating instructions. If you wish program returned, please enclose sufficient return postage.

write for free colour catalogue



Suite 210, 5950 Côte des Neiges,  
Montreal, Quebec H3S 1Z6



**THE TORONTO INTERNATIONAL  
SOFTWARE SHOW  
SEPTEMBER 20, 21, 22, 23  
1984  
INTERNATIONAL CENTRE,  
TORONTO, CANADA**



Hundreds of software applications will be featured by manufacturers,  
publishers, dealers and distributors  
from all over Canada,  
U.S.A. and around the world.

Seminars will be available to further show you  
first hand how to obtain the best software  
to meet your needs.

Also showing will be operating languages,  
hard disc drives, duplicating services, user interfaces,  
future trends, packaging and marketing services,  
peripherals and accessories.

Plan now to attend!

Mark the dates on your calendar now!

For further information contact:  
Cameron MacDonald, Debbie Bannon, Allan Stiver  
Hunter Nichols Inc.  
721 Progress Avenue,  
Scarborough, Ontario, M1H 2W7  
(416) 439-4140





# TPUG *magazine*

*Publisher*  
Chris Bennett

*Editor*  
David Williams

*Assistant Editor/Production Mgr.*  
Sandra Waugh

*Advertising Manager*  
Diane M. Hoffmann  
(416) 782-1861

*Production Assistants*  
Astrid Kumas  
Marya Miller  
Monica Bodirsky

*Graphic Design*  
Leslie Smart and Associates  
Toronto, Ont.

*Printing*  
Carswell Printing Company  
Toronto, Ont.

*Cover Photo*  
Roberto Portolese Studio  
Toronto, Ont.

TPUG Magazine is published 10 times a year by TPUG, Inc. All rights reserved. No material may be reprinted without written permission.

TPUG yearly memberships:  
Regular member (attend meetings) — \$30.00 Cdn.  
Student member (full-time, attends meetings) — \$20.00 Cdn.  
Associate (Canada) — \$20.00 Cdn.  
Associate (Overseas—sea mail) — \$30.00 U.S.  
Associate (Overseas—air mail) — \$40.00 U.S.

TPUG Inc.,  
1912-A Avenue Rd., Ste. #1  
Toronto, Ont.  
M5M 4A1

Telephone numbers:  
Business Office (416) 782-8900  
(416) 782-9252

Magazine Office (416) 782-1861

VIC 20, Commodore 64 and SuperPET are trademarks of Commodore Electronics Ltd. PET is a registered trademark of Commodore Business Machines, Inc. CBM is a registered trademark of Commodore Electronics Ltd.

TPUG Magazine is printed in Canada. Mailed at Toronto, ON and Buffalo, NY. Send change of address to: TPUG Inc., Address Changes, 1912-A Avenue Rd., Ste. 1, Toronto, ON M5M 4A1

Subscription: 15,500  
Newsstand: 8,000  
ISSN #0825-0367

*Distributed by:*

Access Computer Services	Compulit Distributors	Micron
630B Magnetic Drive	PO Box 352	409 Queen St. W.
Downsview, ON	Port Coquitlam, B.C.	Toronto, ON
M3J 2C4	V3C 4K6	M5V 2A5
(416) 736-4402	(604) 464-1221	(416) 593-9862

Dealers Inquiries ONLY  
1-800-268,1238

Subscription-related inquiries  
are handled ONLY with TPUG

## TPUG CONTACTS

### TPUG Board of Directors

President	Michael Bonnycastle	416/654-2381
Vice-President	Chris Bennett	416/782-9252
Vice-President	Gord Campbell	416/492-9518
Treasurer	Carol Shevlin	c/o 416/782-8900
Recording Sec.	John Shepherd	416/244-1487
	Bruce Beach	519/925-5376
	Rosemary Beasley	416/787-8432
	Gary Croft	416/727-8795
	Mike Donegan	416/639-0329
	John Easton	416/251-1511
	Gerry Gold	416/225-8760
	Louise Redgers	416/447-4811

Business Man.	Chris Bennett	416/782-8900
Asst. Bus. Man.	Doris Bradley	416/782-8900

### TPUG Magazine

Publisher	Chris Bennett	416/782-1861
Editor	David Williams	416/782-1861
Asst. Editor	Sandra Waugh	416/782-1861
Ad Manager	Diane Hoffmann	416/782-1861

### Meeting Co-ordinators

Brampton	Gary Ledez	c/o 416/782-8900
Central	Michael Bonnycastle	416/654-2381
Commodore 64	Louise Redgers	416/447-4811
Comal	Don Dalley	c/o 416/782-8900
	Victor Gough	c/o 416/782-8900
Communications	David Williams	416/782-1861
Eastside	Kelly Grinton	c/o 416/782-8900
	Peter Schwartz	c/o 416/782-8900
Machine Language	Jim Carswell	416/531-9909
SuperPET	Gerry Gold	416/225-8760
VIC 20	(Doris Bradley)	416/782-8900
Westside	John Easton	416/251-1511
	Al Farquharson	519/442-7000

### Librarians

Commodore 64	David Bradley	416/782-8900
	Richard Bradley	416/782-7320
French	Baudouin St-Cyr	c/o 416/782-8900
PET	Mike Donegan	416/639-0329
SuperPET	Bill Dufield	416/224-0642
VIC 20	Craig Bonner	416/663-4025
Assistant	Chris Covell	416/925-9296

### Bulletin Boards

Tom Shevlin	416/624-5431
-------------	--------------

### Conference

Gord Campbell	416/492-9518
---------------	--------------

# editorial

We sometimes have some interesting discussions in this office about the function of *TPUG Magazine*. Should it endeavour, as far as possible, to be a publication vehicle for TPUG members who have things they wish to say, or should its main concern be to cater to the needs and tastes of its readers by providing material which they will find interesting, useful and instructive?

This is not, by any means, a trivial question. Authors are often fired with a drive to tell the world about some marvellous new thing which they have discovered or done. I know this from experience; I have often felt this motive to write myself. It can lead to the production of interesting articles, but they tend to be somewhat technical and specialized. Readers, on the other hand, tend to want articles on a much more elementary and general level. At any moment, a large proportion of TPUG members are absolute beginners at computing, who are far more interested in learning how to list a disk catalog than in reading pages of listings in assembly language.

As a club publication, *TPUG Magazine* must try to strike a balance between these competing desires of club members who are authors and its readers. However, if for no other reason than that readers outnumber authors by several hundred to one, the balance must be weighted heavily in favour of the readers. This does not mean that the

magazine will cease publishing specialized articles. There are plenty of readers who appreciate them. However, I hope that future issues will contain increasing numbers of articles which have been written by experts at a "beginner" level. David Hook's excellent series on disk handling, of which the second article is in this issue, is a good example of what I mean.

While I am on the topic of experts, I am happy to announce that *TPUG Magazine* has made an agreement to publish a series of articles on the COMAL programming language by one of its Danish authors, Borge Christensen. The first of these articles should be ready to publish in this issue of the magazine. This series should provide a source of authoritative, accurate information for all readers who are interested in COMAL. In order to maintain a reasonable balance of the magazine's contents, no other articles on COMAL will appear while Mr. Christensen's series is running.

A few readers and authors who live outside Canada keep surprising us by being apparently unaware that this magazine is a Canadian publication. The overwhelming majority of the material we publish is applicable in all countries. However, readers outside Canada should check technical standards, legal requirements, etc., in their own countries. We cannot guarantee that the information that we pub-

lish is universally accurate. People can also save themselves money by being aware of our nationality. We now have enough foreign (mainly American!) postage stamps to decorate our office. They are pretty, but there is no way we can use them for return postage!

This is the last issue of *TPUG Magazine* which will appear for two months. As has been TPUG practice in the past, there will not be an August issue, as such. A combined August/September issue will appear in time for members to receive it by the beginning of September. It will contain such things as the calendar of TPUG events for the fall season.

David Williams

Important message  
to all bbs users.

The TPUG BBS has a new  
telephone number and  
new hours.

The new number is...  
(416) 429-6044.

The new operating hours are...  
24 hours per day.

7 days per week.

The new password is...

**bluejay**

MR. DRACULA

ANYONE WHO PAINTS  
MESSAGES ON WALLS  
HAS PROBLEMS.

PRINT  
HI GUYS

I TOLD YOU.

PRINT  
HI GUYS

?SYNTAX  
ERROR

quillan

# THIS & THAT

Doris Bradley  
Asst. Bus. Man.

My recent travels took me to Bermuda, where some of TPUG's international members reside. Along with the usual sightseeing activities I paid brief visits to the Canada Forces Base near Somerset, where the Disk Drivers computer club meets regularly, and to the Kellys, who originally had a photography business but now deal in Commodore computer equipment and software as well. Mrs. Kelly told me how last Christmas Eve they were open until 10:30 at night because of the interest in VIC 20's and Commodore 64's for Christmas. I am sending them some new TPUG information packages for these two machines. Perhaps there are others of you out there who would like some of these to give to friends or to have in your store. All it takes is a brief note to the office and some will be on their way.

## New Information Packages

Our new information packages, dated May 15, 1984, for the PET/SuperPET (24 pages), VIC 20 (16 pages) and Commodore 64 (24 pages) are now available. If you want a copy of the updated library for your computer(s) i.e. Commodore 64, VIC 20, PET or SuperPET, please fill in the form below and send it in to the office.

## North/South

A few months ago I mentioned that Dorothy Ann Thom of Nanisivik, on the north-west coast of Baffin Island, seemed to be the northern-most member of TPUG. To date I haven't received any challengers — and — no-one has claimed the title of southern-most member. A quick look at the globe indicates to me that Tasmania or the south island of New Zealand will provide the answer. Who is it?

## Third Annual Conference

While I can't report on the actual conference itself (it's almost two weeks away), the deadline has passed for pre-registration so I can tell you that members from 20 of the 50 United States as well as all 10 Canadian provinces will attend. In addition, Peter Fontilus of the Netherland Antilles and perhaps

one of our Australian members will attend.

## 5 and S

I'm sure you can easily distinguish the printed "5" and "S" above, but did you ever stop to think how hard it can be to tell the difference when they are handwritten. We in the office have the most difficulty with orders for (C)TS and (C)T5. Quite often we make a calculated guess and hope for the best. It would help if you added the appropriate descriptive phrase (C)TS — March 83, and (C)T5 — January 84. Thanks.

## Other Computer Clubs

TPUG does have a growing number of computer clubs which have joined as associate members of the group. We try to include information submitted by any of these groups regarding their meetings on a one-time only basis. I have recently received the following:

The **NORthern NY Commodore Home-computer Users Group (NORNY-CHUG)** meets on the 2nd Monday of each month in the Norwood Elementary School, Norwood, NY at 7 p.m. Contact *Andy Van Duyn* (315) 353-4591.

The **Yarmouth (Nova Scotia) Commodore Users' Group** meets every first and third Wednesday of the month at South End Elementary School. Contact *Reg Newell* 742-3348.

The **Commodore Computer Club of Ft. Smith, Arkansas** was formed several months ago for users in the Western Arkansas and Eastern Oklahoma area.

Monthly meetings are held the second Tuesday of each month at 9601 Rogers, Ft. Smith.

The **Commodore Reno Users Network (CRUN)** meets on the first Saturday of the month at Truckee Meadows Community College at 10 a.m. Contact *Franklin Miller* 322-7103 between 7 p.m. and 9 p.m. weekdays.

The **Mid-Atlantic Regional Commodore Association (M.A.R.C.A.)** is holding its First Annual Commodore Users' Conference and Computer Show July 28th — 29th, 1984, at the Hershey Convention Centre, Hershey, PA. Among speakers lined up for the two-day conference: **Jim Butterfield, Jim and Ellen Strasma, Len Lindsay and Lou Sander.** Cost is \$15.00 for both days, and there is a Spouse's Program. For tickets or information write to: **M.A.R.C.A., P. O. Box 76, Mt. Holly Spring, PA 17065 TPUG**



### Request for Information & Library Listing

Please send me the most recent information package(s) as checked below.

Commodore 64  VIC 20  PET/SuperPET

Name

Street Address

City/Town/P. O.

Province/State

Postal/ZIP Code



---

# CENTRAL MEETING - May 1984

---

Ian A. Wright  
Toronto, ON

I made a special effort to show up at TPUG's May Central meeting since some friends were making presentations — little did we know that this would turn out to be an evening full of surprises!

Jim Butterfield had asked if he could present first because he had to make another engagement that same night — unfortunately, unknown to him, the equipment he was using was booby-trapped! Jim wanted to show various methods of accepting input into Commodore equipment and to explain the disadvantages of each. The problems associated with the INPUT statement have been documented many times and Jim gave a short review using a program he wrote on the screen — no problem so far. Jim then wrote a small program to use the GET statement and it was at this point that his problems started. It was most unusual to see Mr. Butterfield stumped by a correctly written program that would only print “NNNNNNN...” on the screen. Jim accepted a variety of suggestions from the audience, and tried a few tricks of his own — all to no avail. By now some members of the audience were nudging each other and giggling. Jim checked the value of the variables — still “NNNNNNN...”. Finally Jim said “. . . someone has been fooling with this machine — who was it?” No response. The perpetrators were in hysterics. With a great deal of aplomb, Mr B. saved the program to disk and re-set the PET using the on/off switch (a good job since sys64790 would have only appeared to have re-set!). From then on his program worked just fine.

Jim wanted to show us how to utilize the INPUT# command with the file opened to the screen. Since the INPUT# command is directed to a specific device it allows you to press <return> without dropping out of the program. Similarly most of the other problems of the bare INPUT command can be corrected easily.

```
100 OPEN I,3      (Opens the screen as input device #1)
110 INPUT#1, N$  (Accepts the rest of this line as N$)
120 PRINT N$     (Print it — this could be to any device)
```

This statement allows you to take in a line of data without having to “GET” the first character and concatenate it with the next character until N\$ is complete. Jim says that users of BASIC 2.0 (the C-64, VIC 20, and older PET/CBM machines) will not run into problems with “garbage collecting” if they use INPUT# rather than GET routines. Jim also noted that this is only the start to a complete input routine.

After a presentation on the final arrangements for the May conference and some other reminders, Mike Bonnycastle introduced Harry Likas, a computer composer and musician from Waveform Corporation who demonstrated an amazing series of programs called **MusiCalc I, II, and III**, plus a three-octave keyboard for attachment to the C-64. Harry was responsible for composing all of the “pre-sets”, the digitized music of **MusiCalc I**, as well as the rock, new-

wave and Latin templates of **MusiCalc II** and **III**. Using the **MusiCalc** system, Harry demonstrated song playback, editing and modification, filtering, and various keyboard modes using the C-64 keyboard. You can even choose to play Blues, Arabian or Egyptian scales! You can design, save, play back and print (in notation) your own musical compositions. Another **MusiCalc** feature was described as “. . . no fault music”, since **MusiCalc** will let you play the C-64 and it will adjust your note selection to mesh with the tempo, pitch, etc., of the background template. Harry even played melody with his elbow!

The program is available from Waveform Corp., 1912 Bonita Way, Berkley California CA 94704. **MusiCalc I** is available now for \$74.95 U.S., and the extensions (**MusiCalc II**, and **III**) are around \$50.00 U.S. The keyboard is expected to be available this summer for between \$250 and \$300 U.S. and there is no availability date for a Canadian version as yet.

Tom Hutchison, an Etobicoke teacher, presented three of programs that he has written to make using Waterloo BASIC a little easier for his students. Structured languages need to follow special formats, and Tom's utility programs allow the student to check his/her work before running it. There are two identical versions of the utility program that load at different addresses (depending on the PET networking system or whether other aids are in the machine). These utilities can be loaded in on top of existing WatBASIC programs without overwriting them, and they have complete internal documentation.

The other program Tom demonstrated was a true merge that does the merge in memory. You can write a program, call in your favourite subroutines from disk or tape and merge them into the middle of the program (which was renumbered by Tom's utility). The same utility will then automatically indent and even check your program for syntax errors!

The last presenter was Jim McLaughlin, of Etobicoke, a student at Thistleton Collegiate and a member of David William's **Machine Language** program run by Etobicoke's S.E.E. school. As part of this grade 13 credit course Jim produced **Not Basic-Aid** a non-utility adding 12 commands to Commodore BASIC on the PET. Just prior to the meeting, Jim had tried to demonstrate his program to his night-school class and it had bombed, so it was hurriedly tested again before the start of the TPUG meeting. This time the program worked fine — but someone forgot to re-set the PET before Mr. Butterfield started his ill-fated demo. The effects of this omission have been described already!

**Not Basic-Aid** uses the CHREGGET routine to check for new commands “COMPACT”, “MUNCH” and “DONNEGAN”, and the results had the audience in stitches. I will not spoil the fun by describing the effects of each of these commands — but Jim's program certainly gave this meeting a surprise ending. TPUG

# Computer Aids for the Disabled



Photos by Roberto Portolese

**William Bennett**  
Scarborough, ON

The personal computer holds a promise for people whose physical disabilities have prevented them from participating in the mainstream of society. *TPUG Magazine* readers probably know how computer electronics can extend a person's nervous system, allowing him to communicate and control without much need for physical mobility or strength.

It has been said that computers can help even the most severely disabled person, but this promise cannot be wholly fulfilled today: we must wait for an evolution in computer equipment. With help from other computer users, disabled individuals can make use of personal computers now. By so doing they may be able to contribute to the development of new technology that will better suit their needs.

TASH (Technical Aids and Systems for the Handicapped) Inc., is a non-profit Canadian company that has been making available the special equipment that is needed by physically handicapped computer users. The Rehabilitation Technology Unit (RTU) of the National Research Council of Canada supports TASH through product engineering and development work. Ideas for new products have come from

disabled computer users, from teachers and clinical workers and concerned individuals. Now that a number of TASH devices can be used with Commodore computers, it is hoped that other people may be encouraged to use them and contribute to the evolution of more accessible computers.

The keyboard is the standard user interface for all personal computers. Almost without exception any software package requires use of the keys at some point, so it is not surprising that physically disabled people are prevented from using computers because they can't fully control the keyboard. Here are the ways that keyboards are being adapted or bypassed in order to meet their needs:

## Latching Keys

TASH supplies mechanical key latches for VIC 20 and Commodore computers. These allow a user of only one finger, or perhaps a headstick, poking at one key at a time, to keep the "CONTROL" or "COMMODORE" keys pressed down while another key is being selected. Like a SHIFT/LOCK key, the latch must be pressed again to unlock it: but there is enough clearance for others to use the keys without the latch.

In the future, it might be possible to change the keyboard handling software to have an optional "hold until the next key is pressed" routine in non-volatile memory, or in a particular software package.

## Keyguards

Some people have tremors in their hands which make it difficult to select one key at a time. They can use a **Keyguard**, which is a plate that covers the whole keyboard and has a chamfered hole over each key. With it, the user can support her hand and poke her finger through the hole for the key she wants. People who use keyguards tend to dislike auto-repeating keys.

Keyguards and mechanical latches are simple adaptations that could be made at home or in a school workshop. They must be designed for the particular model of computer that is being used. Another computer would likely require a different design altogether.

## Replacement Keyboards

Some computer users find that the existing computer keyboard will not suit their needs. There are aftermarket keyboards with numeric keypads; others with programmable function keys, lower profile or lighter touch. The **TASH King Keyboard**, which you can see on the front cover of this issue, is made for a person who needs to hit the keys with a fist, an elbow or a foot. It is also an aftermarket keyboard with a complete ASCII character set and it comes with a hardware interface for Apple computers.

The **King Keyboard** is two feet long by one foot wide. The keys are spaced two inches apart, so they are big targets to

*continued overleaf*



aim at. The keyguard is built into the assembly, which contains microswitches, for extremely short travel and low force.

The bright colours used for alphabetic, numeric and special keys are there for a good reason. Because a lot of the people who need to use this keyboard have perceptual difficulties, a maximum amount of user feedback was built in. Electronics inside the keyboard provide latching SHIFT and CONTROL keys, with indicator lamps to remind the user. The switch "response" time can be varied from zero to two seconds, so a person can be sure he or she will have enough time to make the correct choice. The tactile click of the micro-switch is followed by a "beep" when the keyboard has responded to a key.

Teachers using the **King Keyboard** with children suggested that the same electronic circuit could be used for a miniature keyboard. This is now being developed by RTU. For people with fine muscle control but a limited range of movement, an extremely light touch will operate the keys.

There is some difficulty in making these keyboards available for use on different personal computers. The means of getting keyboard data into the computer varies widely, so a special interface must be provided for each model of computer. In some cases, this could involve partially dismantling the computer. Also, varying keyboard layouts and the use of special keys such as the "Commodore" key on the VIC 20 and C-64 have to be accommodated.

Some people who would like to use personal computers cannot handle the physical array of switches that is a standard keyboard. If a person can operate just one or two switches (whether by moving the head, wrinkling the eyebrow or by a shaky motion of the hand) then he or she should be able to gain control of a computer. For instance, he or she might have a screen menu of alpha-numeric characters to choose from. Or he or she might use Morse code switching to generate the same characters.

Systems that enable the user of a simple controller to generate keyboard data are called keyboard emulators. Some make use of the keyboard's own memory address space:

others stand alone and connect to computer in the same way as a replacement keyboard.

#### Software-Resident Special Keyboards

If the computer's own RAM is used to hold the keyboard emulator software, there is the possibility that applications programs may intrude upon it. As well, pre-booting the special keyboard software may be awkward for the disabled user.

The new battery-powered portable personal computers offer one solution: the machine could be permanently equipped with word processing and communications software, with the special keyboard software included. Several groups are working in this area. As new portable computers arrive, the keyboard adaptations become more powerful and more refined.

Other keyboard emulators use their own memory, battery backed-up, on a separate circuit board. The **Adaptive Firmware Card** for Apple computers is one which offers a large number of user input techniques.

In either case, the special keyboard program uses keyboard address calls as a cue to process and insert its own keyboard data. The bottom row of the computer screen can be used to display a menu of characters and standard game control inputs are used to choose characters.

#### Stand-Alone Keyboard Emulators

Keeping the keyboard emulator separate from the host computer gives several advantages. The user interface is free of interactions with the programs being run on the host computer. A separate screen can display a full keyboard of characters and more. A software or hardware interface built into the host computer will accept input from the keyboard emulator: if the user wants to change over to a new computer, he need only install a keyboard interface in the new one.

The **TASH MOD Keyboard System** is a stand-alone keyboard emulator. It uses a Commodore VIC 20 to handle the video display, a variety of user interfaces and outputs to the host computer.



*continued on next page*

The **MOD Keyboard** program is contained in a special memory cartridge that has 4K bytes of battery backed-up RAM, as well as PROM. Thus, when the user has set up the system according to his needs, the RAM will retain those settings after power has been shut off and the system will reappear as soon as power is turned on again.

The upper half of the **MOD Keyboard** display is a QWERTY Keyboard layout. The lower half is one of fourteen pages of words or phrases that the user has set up for himself. VIC 20 game input signals direct a cursor (as for menu selection) over the whole screen. When a selection has been made it is displayed on the bottom row of the screen and it can be sent to the host computer at the same time via the **USER I/O port** or the **Commodore Serial Port**. The system can be used as a simple communicator, with a printer or speech synthesizer connected via an RS232 serial connection. Interfaces to several brands of personal computers are available and others are now in development.

The most severely physically disabled person can have equipment to operate a computer. It is much more difficult

and time-consuming to assess the person's capabilities, to choose appropriate interface equipment, to provide support for the person and his control switches and finally to teach him how to use the equipment.

If the effort involved in adapting computers to the disabled individual does result in an evolution of more accessible software and hardware, then it is time well spent. The clinical process of evaluation, fitting and teaching will be seen as a positive benefit and it will be easier. More people will be able to participate in a society that increasingly depends on the power of computers for creation and communication. *TPUG*

*For more information on adapted computers for the disabled you may call:*

*TASH Inc.  
70 Gibson Avenue, Unit 1  
Markham, Ontario, Canada  
L3R 2Z3  
tel: (416) 475-2212*

## THE COMPUTER IN SPECIAL EDUCATION



*Photo by Marya Miller*

**Marya Miller**  
Mount Albert, Ontario

The teacher has a PET. And an Apple. But she prefers her C-64.

"The C-64 is actually mine," says Joyce Brown, Diagnostic And Resource teacher for Mount Albert Public School. "I brought it to the school just after Christmas and it's the one we use most."

"The Apple and the PET belong to the whole school, but the children in Special Ed. use the computers so much, they think they own them. . ."

Joyce Brown has spent the last six years working with children who are classified as 'exceptional' — a term that covers the spectrum from slow learner to gifted to emotionally-disturbed to learning-disabled. She recalls that during the first year, all areas of exceptionality were lumped together, "which was difficult". More recently, she has been helping exceptional children to integrate with regular classes. She finds that it is less traumatic for the children if they are officially in a regular class and are withdrawn only for Special Education classes in their particular area of difficulty — the parents accept the idea that their child is exceptional more readily and there is less of a problem with peer acceptance, too.

Joyce Brown and her co-worker, Florence Hutchinson, are constantly re-thinking and re-working their Special Education classes. They are always on the look-out for new tools and new ideas to use with their students. "We have to be very flexible," says Brown. "We have so many different children, all of them with different problems. And very often there are varied age groups within each class."

Joyce Brown has nothing but praise for the computer as part of a 'multi-media' approach to special education. She calls it "a great motivation force", citing an increase in the ability to concentrate and greatly expanded attention spans as the most instantly noticeable results.

Both the children and Joyce Brown seem to prefer the C-64 to the PET and the Apple. They agree unanimously that the C-64 keyboard is easier to use and Brown adds; "There are too many fancy keys on the Apple."

*continued overleaf*

She considers herself strictly a "user", picking up her computer skills by repeated exposure. She notes rather ruefully that "the children often know more than I do about computers". (I am reminded of this statement a few days later, watching young Kevin and John zestfully instruct special education supply teacher Marilyn Hartley on loading in a program).

The children are certainly at home with the C-64. Some have actually gotten their own computers since the introduction of the PET into the school two years previously, and the others I spoke to say hopeful things like; "I'm getting one for Christmas." Their response is, to say the least, enthusiastic.

Joyce Brown buys most of the software herself. There are some "very good" programs available from Ontario Software "which arose from several school boards investing money in order to get a wide variety of programs put together on disk and tape." These programs are available free-of-charge to board members, but Joyce Brown finds the selection for C-64's "limited". She finds herself using commercial software such as **Typing Tutor**, **Bank Street Writer**, **Logo** and **Bristles**, to name a few.

She uses both 'educational' software *and* games. Of the educational programs, **Typing Tutor** and the **Bank Street Writer** word processor are the most heavily used (as well as math software for the PET and Apple). These programs drill the children in the basics of English and math but the procedure is apparently so much fun that the children hardly even notice that they are doing work they once found distressing and difficult. "I've had children *begging* to do math on the computer," says Brown (not quite able to keep a hint of awe from her voice). On **Bank Street Writer**, Brown says; "I've had children who have never voluntarily written a story on their own actually ask to write one." She attributes this in part to the pleasure that slow learners and learning-disabled children derive in seeing a neatly-finished product on screen, or printed out on the dot matrix printer.

"Some of these children have real difficulty in producing a finished piece of work by hand. It's a great incentive and really boosts their confidence to see the finished result from a printer." She finds both **Typing Tutor** and **Bank Street Writer** particularly helpful to children with poor motor control and spelling difficulties.

Games are also very popular with the children, though Brown admits she has had parents ask: "Why are they playing these games?" Once she explains, however, she meets with no further resistance.

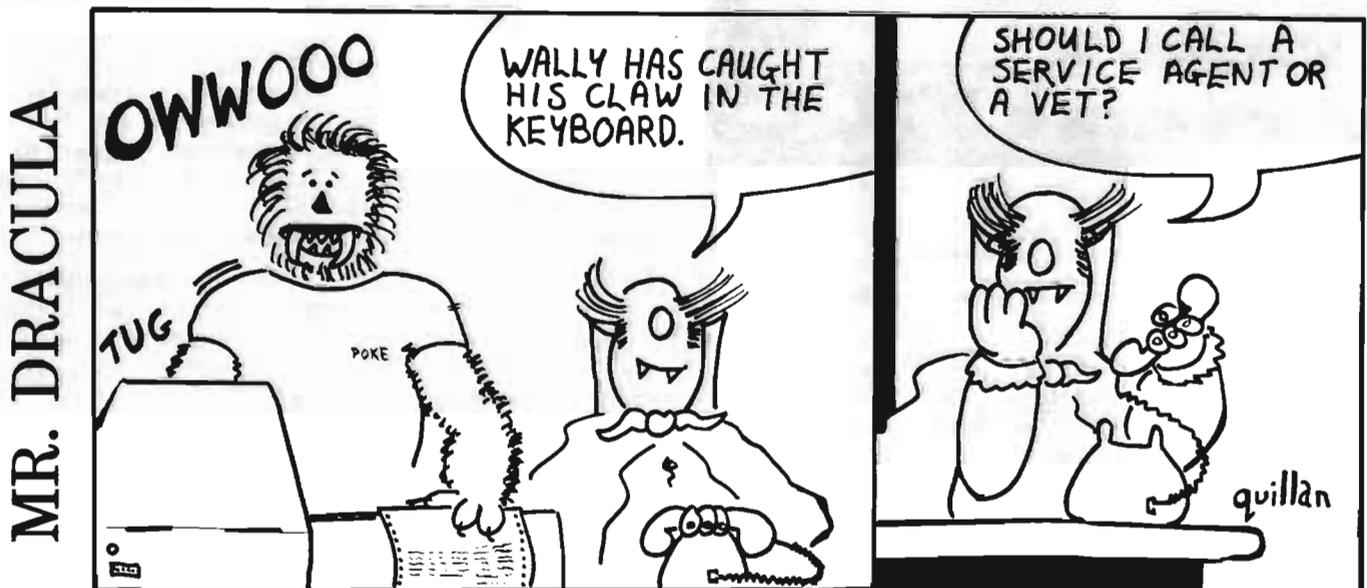
"The game cartridges are good for quick-thinking and decision-making, left and right, up and down – directionality – all the modality training, as well as concentration." Some of the favourites include **Pizza**, **Lemonade Stand** and **Bristles**. (I watched several runs of the latter and got quite a kick out of Brenda The Brat following the painter around the house, nastily messing up the work he had just done). (So, apparently, do the children!).

One aspect of the C-64 in the classroom that appears to give Joyce Brown real satisfaction is the development of positive attitudes it encourages. "I have *never* had a child sit down at the computer and say, 'I can't do that'". She is emphatic.

Another major benefit is a noticeable improvement in the childrens' behaviour and attitude towards their peers – "especially emotionally-disturbed ones". Brown marvels at groups of three or four children round the C-64, "all working *together*: co-operating, helping each other. . ." She has noticed that the desire to keep computer privileges is a great incentive towards good behaviour.

"But the computer is still more of a visual motivator," she concludes. "It doesn't help much with auditory problems. As of yet you can't speak to a computer." It is the closest she comes to a criticism. "But socially, educationally, motivationally – all I've seen are positives with the children."

"...Other than that you *can't* tear them away. . ." TPUG



# HARBOURFRONT COMPUTER CENTRE



**Janet Sherbanowski**  
*Toronto, ON*

Even though our main interest is in Commodore products, sometimes it's wise to check out the other micro-computers on the market. A perfect place to do this and have fun at the same time is **The Harbourfront Computer Centre**, located at 207 Queen's Quay in the Queen's Quay Terminal Building, in Toronto.

The available hardware includes donations from IBM, Olivetti, Atari, Osborne, the Canadian made Hyperion and a whole room set aside for Commodore.

Four years ago Stan Squires, now man-



ager of the Centre, began a Computer Centre at the Oakville Library. Funded in part by Harbourfront, Provincial and Federal Governments, the initial experiment in Oakville expanded to include the Harbourfront Centre and the Federal Government has expressed interest in setting up additional centres across the province.

The multi-user environment created at Harbourfront has benefits both for the public and the companies who have donated machines and software.

The public gains easy access to thousands of dollars of new technology that otherwise would not be easily located or used and the companies gain widespread exposure for their machines.



The main aims of the Centre are to introduce and improve computer literacy for all ages, provide a hands-on computer literacy program to the public without being a "school", and to encourage people to appreciate and apply micro-computer concepts to their daily lives.

You can arrange for individual instruction on any of the machines, with access to their manufacturer supplied software for \$18 per hour or — if you choose, to supply your own software — the price drops to \$8 an hour.

The Centre provides a varied selection of year-round programming for everyone from young initiates to businesses looking for cost effective ways of training their employees in the new technology.

Beginning with the pre-schooler, Centre staff develop the child's concentra-



tion, shape and recognition skills, math and pre-reading abilities. School-age children can get hands-on experience in BASIC, LOGO, Pilot and educational software. Business applications programs (designed to introduce the user to Multiplan, word processing, office management systems and databases) can be arranged for day or evening two-hour sessions, cost around \$48 and run for four weeks.

The programming plans go on and on including: birthday parties, special programs for the handicapped, summer Computer Camps for kids, and even whole weekends this Summer and Fall devoted to Commodore user education.

A long term goal for this Centre is to become the "library of the '80's", by providing a world-wide system of information networking.

If you live in the Toronto area the Harbourfront Computer Centre is worth a visit to find out first hand just what Apples and IBMs are all about. If you are one of the many international TPUG members, the Harbourfront area of Toronto is something to see in itself but with a micro-computer mini-world there you can't afford to pass it up when you visit Toronto. *TPUG*

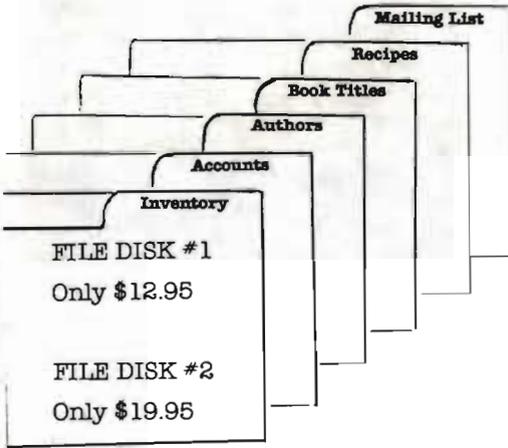


*Photos by Rob Baxter*

# FILES

## EASY TO USE FILES FOR NEW USERS

Don't wait to start using the power of your 64. . . Start using **FILES**.



FILE DISK #1  
Only \$12.95

FILE DISK #2  
Only \$19.95

FILE DISK #3  
(Both #1 & #2)  
\$29.95

(Ontario residents add 7% PST)

(Cheque or Money Order only, please)

Whatever you have in mind . . . Mailing Lists, Recipes, Book Titles & Authors, Personnel, Accounts, Inventory, etc., etc., **YOU NEED FILES!!**

**FILE DISK #1** Creates, Opens and Closes sequential files for you. It will let you write to a new file, or add (append) to an existing file . . . and enable you to read your files to the computer screen or your printer.

**FILE DISK #2** Lets you create **RELATIVE** files . . . **FAST . . . EASY!!** Almost **INSTANT ACCESS** to any record - Hundreds of records in one file - Up to 20 fields in each record - Access records by # or by 'key' field - Create - Write - Find - Read - Change - Delete or Print any record. Set your own file size and field names. Your own data base . . . **EASY . . . FRIENDLY!!**

**FILE DISK** programs are all written in **BASIC** that you can list and study. As your own programs improve, you can copy routines that error check, sound alarms, etc. **FILE DISK** programs are Menu Driven . . . and include **TUTOR** routines to help you. Also includes complete printed instructions.

**FRIENDLY SOFTWARE**  
55 Ellerslie Ave., Suite 1117  
Willowdale, Ont., M2N 1X9

# We'll back you up!

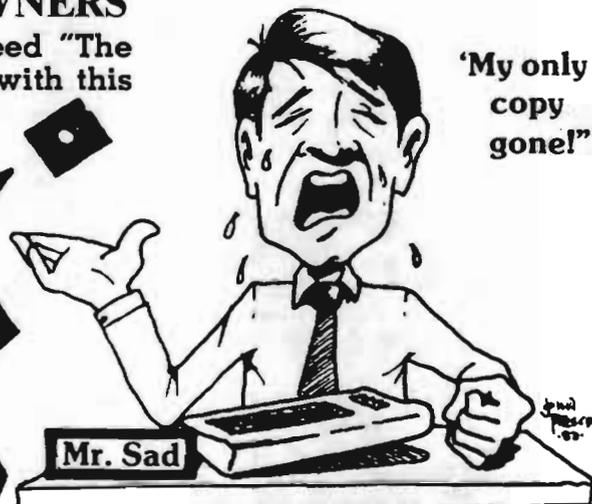


## ATTENTION COMMODORE 64 OWNERS

If you own a disk drive then you'll need "The Clone Machine". Take control of your 1541 with this package that includes:

- 1.) Complete and thorough users manual
- 2.) Copy with one or two drives
- 3.) Investigate and back-up many disks
- 4.) Copy all file types including relative types
- 5.) Edit and view track/block in Hex or ASCII
- 6.) Display full contents of directory and print
- 7.) Change program names, add, delete files with single keystroke
- 8.) Easy disk initialization
- 9.) Supports up to four drives

**Special intro \$69.95**



Mr. Sad

'My only copy gone!'

Amount enclosed \_\_\_\_\_  
 Name \_\_\_\_\_  
 Address \_\_\_\_\_  
 City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_  
 Phone Number ( \_\_\_\_\_ ) \_\_\_\_\_  
 MC or VISA # \_\_\_\_\_  
 Expiration date \_\_\_\_\_

Ont. res. add 7% PST

Shipping and handling add \$2.50

## Creative Computer Systems

106 Byron St. South  
Whitby, Ontario  
L1N 4P5  
(416) 666-2212



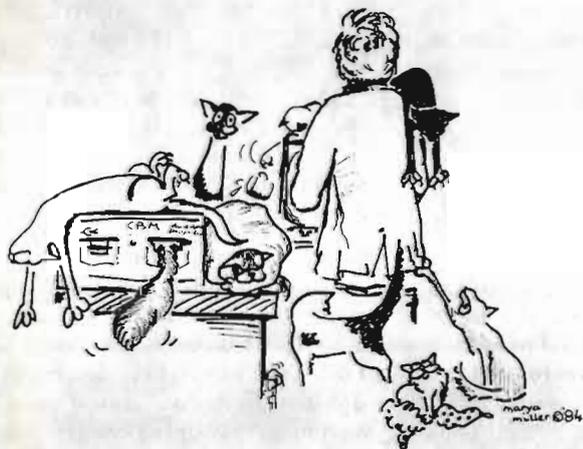
# THE GREAT 25¢ EXPERIMENT

Jim Butterfield  
Toronto, ON

It seemed like a good idea at the time. We were in the early days of home computing and the passing around of programs hadn't been refined to an advanced system. So I thought I'd try a form of computer chain letter.

You must know the sort of thing I mean. The letter which says something like:

*"This letter has brought good luck to thousands. Follow these instructions and it will bring luck to you. Bundle up your cat and mail it to the name at the top of the list. Now, make four copies of this letter. Cross out the name at the top and put your own name at the bottom of the list. Within six weeks, you will receive 65,536 cats. Do not break the chain: one man did and got his own cat back."*



*"...within six weeks, you will receive 65,536 cats..."*

Regular chain letters are bad news. They are usually an attempt to exploit the reader; most of the time, only the names near the top of the list will ever receive any goodies. But I thought: maybe this might constitute a valid way for a user to disseminate programs and make a little money from it.

I wrote a set of eighteen programs: nine were recreational in nature and nine were "serious". In each program, I put a notice: "If you copy this program, send 25¢ to the following address . . ." I passed the programs out to numerous user groups, telling them that the 25¢ message didn't apply to the groups themselves.

Here's what I had in mind: as programs got passed around, a few dozen, or hundred, quarters would make their way back to the programmer. Not big money, but it would be a sign that the program was still moving around and being used by people.

I did this in the winter of 1977. It was intended as an experiment to see what happened; I really didn't want the money. By summer of 1978, no quarters had been received.

I did get one phone call from an angry computer owner telling me that he had no intention of sending me a quarter; I told him that it was all right, I didn't really need his quarter, but it didn't seem to cool him down. At about that time, I wrote Len Lindsay (who was playing around with protection systems) and told him of my lack of success in getting a response.

A few weeks later, about eight quarters trickled in from various parts of the continent. Delayed reaction? I don't know. In late summer of 1978, I decided that the experiment had run its course and wrote any correspondents that they should remove the 25¢ notice and give the programs away free.

Then — months later — Lindsay published comments from my letter in the *PET Gazette* as part of an editorial on how dishonest people were. I didn't agree with his conclusions, by the way . . . and I wondered what the few people who had sent a quarter would think about an editorial saying I had received none.

Len's editorial provoked a flow of what looked like conscience money. I must have received three or four dollars worth of quarters in the following week. One person wrote to say, "I don't know if I have any of your programs, but in case I do, here's a dollar".

I wrote back to the donors, telling them that the experiment was long since over and that they should remove the messages from any programs they had on hand. And that seemed to be the end of it.

I still receive a quarter in the mail every six months or so. I don't recommend the method as a road to big money.

What does this all mean? Are people dishonest? Is it too much trouble to send a quarter? Was the suggested amount too high or too low? Peter Jennings suggested that I'd do much better if the amount named had been five dollars.

I think we can conclude that the system as conceived didn't work and I would not recommend it to any software house hoping to find a new distribution channel this way. I don't think you can make too many sweeping conclusions about honesty. It can be a lot of work to address an envelope, stamp it, find a quarter and tape it inside, all when you're trying something out on the computer. I'll bet that there were many well-meaning users who just never got around to it. (Please — don't send your quarter now. It's over).

I believe that the experiment shows a need for some type of formalization between a commercial supplier and end user. I'm not a big fan of protection systems, since they lessen the value of the programs they protect. I'd prefer to see programs protected by means of good documentation, support, update services and warranty. But I can understand the fears of software houses that users won't make a special effort to pay for programs.

*continued overleaf*

---

It's a pity that the experiment was a failure. It could have been very attractive for a program writer to offset some of his or her costs with a small reward, at the same time allowing and encouraging free distribution of the programs concerned. But we learn from negative results as well as positive ones.

So . . . since it looks like financial rewards are impractical for small programs, you'll have to go for fame. Donate your next program to your club's library; but be sure that it starts by printing your name prominently on the screen. That way, users will know who wrote this wonderful piece of code. Or, alternatively, they'll know who to blame. TPUG

---

## VIC 20 - C-64: SPEED DIFFERENCES

Roger D. Burge  
London, ON

What's the difference between a Commodore 64 and a VIC 20? No, it's not 44. It's usually speed.

This may come as a surprise to all the proud C-64 owners reading this, but the VIC 20 is faster. You usually won't notice the difference, but it's there, nonetheless.

You will notice it if you're writing a program, especially a game that features animation and sound; or when you type in a BASIC game from a magazine and you see the VIC 20 running at a friend's place.

Then you see the spaceships zooming along through the asteroid belt while your C-64 rocket chugs along your screen with the gusto of a VW Beetle. To be fair, though, many BASIC programmers have gotten around the speed problems. The problem is: *how do you do it and why do you have a speed problem to begin with?*

First of all, in a C-64 you have a huge amount of RAM that the computer's operating system must constantly refresh. That is to say, it is electronically refreshed, otherwise it wouldn't remember a thing you've typed in. The more RAM, the longer each pause to refresh. Even a 32K VIC 20 doesn't have as much RAM as the C-64's memory banking system.

Then there's the large video system, a 1000-cell screen memory and colour memory to constantly maintain. Again, this means more time taken each second to keep the system just running, let alone follow your program's instructions.

That large screen also means that it takes longer for a character to move from one end to another and from top to bottom.

But the bottom line is that V2 BASIC is 5 to 7 per cent slower on a C-64 than on a VIC 20. Animation, with sound, is slower still.

If you take advantage of the C-64's special features, more time is lost. Programming the SID chip is far more complex and time-consuming in BASIC than the VIC 20's simpler but effective sound system. Those wonderful sprites are also somewhat slow in movement, particularly moving across the infamous "horizontal seam".

Does this mean that the VIC 20 is a better machine? No, these two models represent the best in home computers for the mass consumer market; each has features that the others have not. You won't hear the dazzling sounds of the SID chip on a VIC 20 or the amazing high resolution and 3D graphics. The VIC 20 is a very powerful and fast computer that's about as "user friendly" as is possible in the VIC 20 and C-64 price range. The VIC 20's "simplicity" is a virtue in itself.

But getting back to our speed problem, you don't have to be a crack machine language programmer to create your own games on either machine. VIC 20 users, with their subcompact memory systems, learn many BASIC programming tricks. With the limited RAM they're forced to be both creative and concise in their thinking or else face the dreaded "out of memory" syndrome. So C-64 users should take a good look at some of the better BASIC VIC 20 programs. They just might find something that will have their aliens going into hyperspace.

If all else fails there is machine language, which is certainly not as easy to learn as BASIC but just as worthwhile. It would be interesting, in a year or so, to find out what percentage of C-64 owners have attempted to learn ML

and compare it with the percentage of VIC 20 users. I have a strong hunch that the C-64's will win.

I recently converted several of TPUG's library games to the C-64, and some of my own VIC 20 creations as well. In some, like **Slither 5** (**Slither Circus** on the VIC 20) and **Slither Duel** (**Snake** for the VIC 20) I had to write some machine language subroutines for reading the joysticks and sound effects, the original games were entirely in BASIC. It was the only way to recreate the speed of the VIC 20. If you get one of these games, I hope you may find something in it you can use in your own work.

One game I wrote for the VIC 20 which will appear in the club library shortly under the name **Riki Tiki** is almost all BASIC. This one is very, very fast but I don't have a C-64 version ready yet. I did a straight conversion once but it was embarrassingly slow. It will have to wait until I have the time to rewrite the entire core of the game in machine language.

Another new entry in the club disk library is **VIC Dodge Cars**, which I wrote in the fall of 1983. Entirely in BASIC, and the action is quick and fun. When I decided to redo it for the C-64 I was determined to do it all in BASIC, just to prove to my VIC 20 that his big brother was a good game player too. Well, it was harder than I thought and I do not feel I succeeded completely. **Dodge Cars 64** will take a little more strategy than reflex but it is comparable in entertainment value. If you have a chance to see both versions, you will find the differences in both appearance and programs quite startling.

The Commodore 64 is a wonder, but I am still thrilled with my VIC 20. I feel users of one model have something to learn from the other. TPUG

## ATTENTION ALL COMMODORE 64, VIC 20, AND PET OWNERS

A complete self-tutoring BASIC programming course is now available. This course starts with turning your computer on, to programming just anything you want! This course is currently used in both High School and Adult Evening Education classes and has also formed the basis of teacher literacy programs. Written by a teacher, who after having taught the course several times, has put together one of the finest programming courses available today. This complete 13 lesson course of over 200 pages is now available for the COMMODORE 64, VIC 20, and PET computers and takes you step by step through a discovery approach to programming and you can do it all in your leisure time! The lessons are filled with examples and easy to understand explanations as well as many programs for you to make up. At the end of each lesson is a test of the information presented. Furthermore, all answers are supplied to all the questions and programs, including the answers to the tests. Follow this course step by step, lesson by lesson, and turn yourself into a real programmer! You won't be disappointed!

We will send this COMPLETE course to you at once for just \$19.95 plus \$2.00 for shipping and handling (U.S. residents, please pay in U.S. funds). If you are not COMPLETELY satisfied, then simply return the course within 10 days of receipt for a FULL refund.

Send cheque or Money Order to:

**Brantford Educational Services**  
68 Winding Way,  
Brantford, Ontario  
Canada, N3R 3S3

Fill in the coupon or send a facsimile.

Name: \_\_\_\_\_

Address: \_\_\_\_\_

City: \_\_\_\_\_

Prov. (Zip) code: \_\_\_\_\_

Complete course: \$19.95  
Postage and handling: \$2.00  
Total: \$21.95

check desired course:

COMMODORE 64

PET

VIC 20

**Brantford Educational Services**  
68 Winding Way,  
Brantford, Ontario  
Canada, N3R 3S3

## FAMILY TREE

A software package to help you use the Commodore 64 or VIC 20 (24K) as a dynamic system to control data on your family tree.

### FEATURES

- 
- Search functions
  - Complete manual
  - Family Record sheets
  - Easy editing and updating
  - Fully indexed
  - Unlimited Genealogies
  - Output to screen or printer
  - 4, 5, 6 Generation pedigrees

## GENEALOGY SOFTWARE

phone 519-344-3990

\$49.95 CND — \$39.95 US

(Ontario and Michigan residents add sales tax)

P.O. Box 1151      1046 Parkwood Ave.  
Port Huron, MI      Sarnia, Ont.  
48061      N7V 3T9

## ★ COMMODORE USERS ★

Join the largest, active Commodore users group.

### Benefit from:

- Access to hundreds of public domain programs on tape and disk for your Commodore 64, VIC 20 and PET/CBM.
- Informative monthly club magazine

Send \$1.00 for Program Information Catalogue.  
(Free with membership).

TPUG yearly memberships:

Regular member (attends meetings)	— \$30.00 Cdn.
Student member (full-time, attends meetings)	— \$20.00 Cdn.
Associate (Canada)	— \$20.00 Cdn.
Associate (U.S.A.)	— \$20.00 U.S.
Associate (Overseas—sea mail)	— \$30.00 U.S.
Associate (Overseas—air mail)	— \$40.00 U.S.

## TPUG Inc.

DEPARTMENT "A"

1912A Avenue Road, Suite 1  
Toronto, Ontario, Canada M5M 4A1

\* LET US KNOW WHICH MACHINE YOU USE \*

# HELP IS HERE

FOR C-64 & VIC-20

## THE LITTLE BLACK BOX RESET SWITCH

- REVIVE and SAVE all CRASHED BASIC PROGRAMS even after entering command "NEW" with aid of included program listing
- Saves hours of work in retyping
- Resets computer from ANY BASIC program lock-up
- Easily installs in USER PORT
- Will not void CBM warranty
- Saves wear and tear on computer
- Diode Protected Circuitry
- Lifetime Warranty!

**Suggested Retail**  
**\$24.95 CDN (\$19.95 U.S.)**



DEALER AND DISTRIBUTOR INQUIRIES INVITED

Manufactured by:

**BESCO PRODUCTS**

#203 - 8060 Granville Ave., Richmond, B.C., Canada Ph: 278-5115

B.C. Distributor:

**LOTTUS DEVELOPMENT CORP.**

14 E. Broadway, Vancouver, B.C., Canada Ph: (604) 879-5445

## C64 CARTRIDGE PROGRAMMER MAKE YOUR OWN CARTRIDGES

Turn your favourite program(s) into a permanent cartridge.

**\$89.95** includes everything needed to program cartridges

- Complete software and documentation
- One blank 8K cartridge included
- Blank 8K cartridges **\$19.95**
- Erase in sunlight and then re-program

**Warning:**

It is illegal to duplicate copyright cartridges

Ask about our  
PET 40/80 conversion

## HIGH RES TECHNOLOGIES

16 English Ivyway

Toronto, Ontario

M2H 3M4

(416) 497-6493

Add \$2.00 shipping and handling

Ontario Res. Add 7% sales tax



Upgrade your Commodore with a high resolution graphics card — installation in our Toronto workshop — call or write for details. Kit form also available.

---

---

# TPUG COMAL COURSE

---

*TPUG Magazine is pleased to present TPUG COMAL Course by Borge Christensen, the founder of COMAL. This series will continue in several more issues. First, however, you will find Len Lindsay's corrections to Ian Wright's article, COMAL—A Structured Language, which was published on page 52 of the March April TPUG Magazine.*

**Len Lindsay**  
Madison, WI

\*PET COMAL version 0.11 is very obsolete. It was public domain.

\*PET COMAL version 0.12 is current, but the update to version 0.14 is expected from Denmark very soon. It is public domain. A complete PET COMAL 0.12 disk is enclosed as my submission to TPUG disk library.

\*CBM 8096 COMAL version 1.02 is current, soon to be replaced by CBM 8096 COMAL version 2.00. It is public domain. It is included on the PET COMAL 0.12 disk I enclose.

\*CBM 8096 COMAL version 2.00 will be available soon. It will be copyrighted.

\*C-64 version 0.12 was released with a warning that it was to be replaced very soon. It is public domain. It has some fatal bugs, and incorrectly implemented graphics.

\*C-64 COMAL version 0.14 is copyrighted—NOT public domain. It is current. Commodore Canada holds rights to it in Canada. COMAL Users Group, U.S.A., Limited holds exclusive rights to in the U.S.A., and hereby grants TPUG a

licence to reproduce and distribute unchanged copies in the U.S.A. to their members. A complete C-64 COMAL 0.14 disk is enclosed as my submission to TPUG disk library.

\*C-64 COMAL version 2.00 will be available only as a cartridge, due FALL 1984.

To convert your BASIC C-64 into COMAL put the C-64 COMAL disk in the drive and enter: LOAD "BOOT";8 and then after it loads, type: RUN.

**Please note:** there isn't a C-64 COMAL version 1.0 and never was. I believe Ian was referring to C-64 0.12 COMAL since he calls it public domain and states 11041 bytes free on boot-up, both which match that version.

C-64 COMAL includes the command BACKGROUND. However, the example BACKGROUND,0 should read BACKGROUND 0 (delete the comma).

COMAL INPUT statements use a colon (not a comma) and always have. The example shown will produce a SYNTAX error on all versions. Line 120 should read:

```
0120 INPUT "TO":T
```

Note COMAL always has 4 digit line numbers. TPUG

---

**Borge Christensen**  
Tonder, Denmark

## Little Boxes

Clear the workspace (NEW) and the screen (<CLR> key) and type the command:

```
AUTO
```

COMAL answers by displaying:

```
0010
```

and waits for you to type in your first COMAL statement. Make it this one:

```
0010 PROC SQSIDE(L)
```

As soon as you strike the <RETURN> key, COMAL answers

```
0020
```

and is ready for your next statement. Go ahead with:

```
0020 FORWARD L
```

```
0030 LEFT 90
```

```
0040 ENDPROC
```

```
0050 //
```

```
0060 PROC SQUARE(L)
```

```
0070 SQSIDE(L)
```

```
0080 SQSIDE(L)
```

```
0090 SQSIDE(L)
```

```
0100 SQSIDE(L)
```

```
0110 ENDPROC
```

```
0120 //
```

```
0130 SETGRAPHIC 0
```

Press the <RETURN> key twice after the last statement to get out of the AUTO mode. If during this short keyboard session you introduce a syntax error, COMAL reports it and places the cursor at the position where it discovers that something is wrong. Let's take it that you "stumble over the keys" in line 70 and type

```
0070 SQSIDE(L)
```

The COMAL interpreter reacts by reporting "ERROR 7" and placing the cursor right after the "L" in line 70. You only need to type in an ")" and press <RETURN>. Then COMAL removes the message and types "0080" instead, thus flagging that it is ready for your next statement. If you are using the disk version and are ready to wait two seconds for a full message in SETMSG+ mode, you get "ERROR 7: BRACKET ERROR" instead of the more laconic "ERROR 7".

Now type LIST to check that you have typed in all the lines of the example. You will see COMAL come up with this listing:

```
0010 PROC SQSIDE(L)
0020 FORWARD L
0030 LEFT 90
0040 ENDPROC SQSIDE
0050 //
0060 PROC SQUARE(L)
0070 SQSIDE(L)
0080 SQSIDE(L)
0090 SQSIDE(L)
0100 SQSIDE(L)
0110 ENDPROC SQUARE
0120 //
0130 SETGRAPHIC 0
```

Note that the listing is structured. All the lines between a PROC and an ENDPROC statement are indented to indicate the range of the procedure. I shall return to the facility of structured listing in more detail later.

If by looking at the listing you discover that a statement is missing, say between 70 and 80, simply type it in by "manually" numbering it 75 like you would do in BASIC. After that you can use the RENUM command to make the listing look more orderly.

Now type RUN and watch COMAL switch to turtle mode (hires graphics with a sprite indicating the pen's position). The turtle is waiting at its HOME position, and the cursor is placed in the upper left corner of the screen.

Try to type

SQSIDE(40)

Press <RETURN> and watch how the turtle draws a line and then turns. Then press <CRSR UP> to scroll the command back on the top line of the screen. Press <RETURN> to have the command executed again. Another line is drawn and yet another turn is made. By doing this two times more, you should end up having a square on the screen with the turtle back in its HOME position.

Type CLEAR to clear the graphics screen. Then type

SQUARE(40)

SQUARE(50)

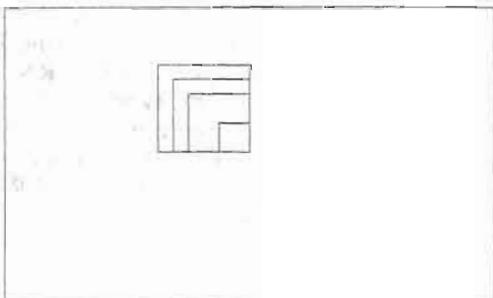
Press <RETURN> after each command.

SQUARE(60)

SQUARE(20)

Each command is a procedure call.

You should now have this picture on the screen:



**Note.** It may be that the square on your screen is not quite true because the vertical and horizontal scale are a little out of proportion. Unless you can adjust your display there is not much you can do about this in version 0.14. The big version of COMAL-80 for C-64 — version 2.00 — has a built in SCALE function to sort out possible distortions.

Return to the text screen by pressing <f1>. Clear the screen and type LIST. Now let us take a closer look at lines 10-40:

0010 PROC SQSIDE(L)	The PROCedure head with the procedure's name and the formal parameter L which is assigned a value when the procedure is called.
0020 FORWARD L	The two statements between the PROC and the ENDPROC statements are called the procedure's body.
0030 LEFT 90	
0040 ENDPROC SQSIDE	The procedure tail with the procedure's name. The name is inserted by COMAL, if not entered by the user.

Next let us see what happens, when the procedure is called with a command or a statement:

SQSIDE(40)	This is a procedure call with the actual parameter 40. It may be any numeric expression.
PROC SQSIDE(L)	The formal parameter L is set to 40.
FORWARD L	A line of length 40 is drawn, and the turtle's head is turned 90 degrees.
LEFT 90	
ENDPROC SQSIDE	Means "return" from the procedure.

The second procedure SQUARE does nothing but call SQSIDE four times. But you should notice that the actual parameter L can get its value from a call of SQUARE. Thus the command SQUARE(40) actually works like this:

SQUARE(40)	
PROC SQUARE(L)	The actual parameter is given by the numeric constant 40. This value is passed to the formal parameter L, in the head of procedure SQUARE. Thus L now has a value of 40, which is then passed to the formal parameter L in the head of SQSIDE.
SQSIDE(L)	
...	
PROC SQSIDE(L)	

This may look as though a lot of confusion could take place as to what L really means. But COMAL can cope with that. A parameter is always local to the procedure in whose head it occurs. Thus the L in SQUARE and the L in SQSIDE are really totally different seen from COMAL's point of view. But right now you should not pay too much attention to that. Just look at parameters as something that pass values from the outside world into a procedure, and let COMAL sort out the problems for you. It has been designed to do that.

# We've Got People Talking

## REVIEWERS SAY:

"This is the best typing tutor we have seen yet; ★★★★★"

INFO-64

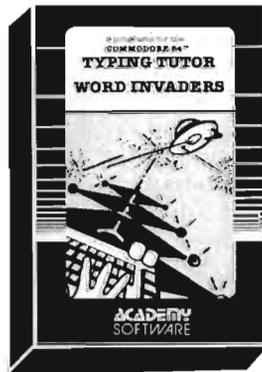
"Computer aided instruction at its best." **Commander**

"This is an excellent program that makes typing practice an enjoyable pastime instead of boring drudgery."

DILITHIUM PRESS

Rated the BEST educational program for the VIC 20

Creative Computing



## CUSTOMERS SAY:

"... delighted with my son's progress ... he is the only one in his second grade class who touch types at the computer."

"Your Typing Tutor is an excellent program ... our 4 children literally wait in line to use it."

"Thoroughly satisfied, can't believe how fast I've learned to type. I've never typed before."

In daily use by schools across the USA.

## TYPING TUTOR + WORD INVADERS

Commodore 64 ..... Tape \$21.95

Commodore 64 ..... Disk \$24.95

VIC 20 (unexpanded) ..... Tape \$21.95

## REALISTIC AIRCRAFT RESPONSE

"Has a quality of realism which sets it apart from others, even those I've tested in flight school."

Compute's Gazette

"Great program!" INFO-64

"It is tremendous fun."

Compute's Gazette

"Flight tested by an air traffic controller, two skilled pilots and an elementary school class. Highly recommended by all."

Midnite Gazette

"This is an unbelievably realistic simulation of the difficulties facing a pilot in instrument flying. I'm a 747 pilot and I think that this simulation could do a lot to improve the reactions and instrument scan habits of even very experienced pilots." **747 pilot**



## IFR (FLIGHT SIMULATOR)

Commodore 64 ..... Tape or Disk \$29.95

VIC 20 (unexpanded) ..... Cartridge \$39.95

JOYSTICK REQUIRED

Shipping and handling \$1.00 per order. CA residents add 6% tax.



# ACADEMY SOFTWARE

P.O. Box 6277 San Rafael, CA 94903 (415) 499-0850

Programmers: Write to our New Program Manager concerning any exceptional Commodore 64 program you have developed.

## compu-simple simon

COMPARE our LOW PRICES to others advertised in this issue!

FLIGHT SIMULATOR II (SubLOGIC)	\$51.99
BLUE MAX (Synapse)	35.99
NECROMANCER (Synapse)	35.99
GOTHMAG'S LAIR (PPS)	41.99
ZORK, I, II, & III (Infocom)	99.99
PAPER CLIP (Batteries)	94.99
CHESS 7.0 (Odesta)	74.99
ARCHON (Electronic Arts)	48.99
CASTLE WOLFENSTEIN (Muse)	29.99
BEACH HEAD (Access)	33.99
LODE RUNNER Broderbund)	39.99
SAMMY LIGHTFOOT (Sierra)	31.99
KMMM PASCAL	149.99
WICO Power Grip Joystick	34.99

Bank Street Writer, Paper Clip, Word Pro 3 Plus and other word processing packages all at low prices. Write for details.

We carry a complete line of software and peripherals for business and home use. Catalogues are 50¢ or free with any order.

To order send certified cheque or money order along with \$3.25 to cover postage and handling (\$8.25 for courier service) to Compu-Simple Simon, P.O. Box 583, Snowdon, Montréal, Québec H3X 3T7. Quebec residents add 9% sales tax.

DEALER INQUIRIES WELCOME

# software by mail

GAMES/ ENTERTAINMENT/BUSINESS/ EDUCATION

- One of the largest selections of computer software in Canada.
- First Class service at competitive prices.
- All orders processed within 24 hours.
- All orders delivered by United Parcel Service.
- Write today for your complimentary catalogue.

# the electric software company

The  
Electric Software  
Company  
DEPT. T

2 Bloor St. West  
Suite 100-108  
Toronto, Ontario  
M4W 3E2

MAIL ORDERS ONLY

---

---

# FORTH: A NEW LANGUAGE FOR YOUR C-64

---

Peter Rukavina  
Carlisle, ON

C64-FORTH is an alternative programming language for your Commodore 64. Programming in FORTH rather than BASIC gives you painless access to graphics, an extra measure of speed and the ability to shape a language to your own needs. This review will centre on C64-FORTH, marketed by Performance Micro Products, but there are several other versions of FORTH available, both for the C-64 and for the VIC 20 and PET/CBM series.

C64-Forth was originally developed in 1968 by Charles H. Moore because, as he says in Leo Brodie's *Starting Forth*, "The traditional languages were not providing the power, ease, or flexibility that I wanted.". Since 1968, the popularity of FORTH has grown, to the point where it is now available, in some form, for all computers.

There are two main reasons to use FORTH rather than Commodore BASIC: speed, and the ability to shape the language to your needs..

FORTH is fast. Because of the way it is set up, much of the interpreting is done at the time you write the code, rather than when you run it. A simple benchmark test, where each of the C-64's 255 characters were written into each of its 1000 screen positions, was performed: first with pure machine code, then FORTH, and finally BASIC. Machine code took about two seconds, FORTH about one minute and BASIC about fifteen minutes. The extra speed comes in very handy when several operations are to be performed at the same time (as with a high-speed arcade game or a communications program); when repetitive calculations must be made (in mathematics programs or simulations) or when a fast routine is required (a feature such as word-wrap in a word processor). FORTH provides a good medium for writing programs if BASIC is too slow, yet machine code too complicated.

FORTH is an extensible language. This means that you can use the core

commands provided with the language to define your own commands. Take, for example, an operation involving several steps, such as starting a lawnmower. The steps involved are: filling the gas tank, turning on the gas, attaching the spark plug and pulling the starter cord until the mower starts. Each of these steps can further be defined. For example: filling gas tank – unscrewing the top, placing nozzle in tank, and pouring in gas till full. In turn, each of these steps can be further reduced. The entire operation of starting the mower can be considered part of the definition of "mowing the lawn", which is, in turn, part of "spring cleaning" and so on. This is the basic premise on which FORTH operates. A more practical example might be the definition of:

**TITLESCREEN:** a routine to display the title screen for a space game.

**ALIENS:** a routine to make aliens fly over the screen (*à la* space invaders)

**YOURSHIP:** a routine to move your ship back and forth under control of the joystick

**SHOOT:** a routine to have the aliens shoot at you and to allow you to shoot at the aliens

**DEATH:** a routine to see if you, or one of the aliens, has been hit and thus should die

These words would then be used in the definition of a word SPACEGAME. Whenever SPACEGAME was executed, TITLESCREEN, ALIENS, YOURSHIP, SHOOT and DEATH would be executed. The game program is thus self-documenting. If one of the operations is altered, only that section of the program need be changed: the main structure remains intact. As you work with FORTH you are, in essence, writing your *own* language, adapted to your own needs. Tired of using a FOR...NEXT loop to clear the SID chip? Define a word SIDCLEAR! Want to have an easy way of setting up for graphics? Define a word GRAPHICS! The list goes on !

Some other nice features are access to any number base (sprites done in binary!), easy graphics (using words

such as ARC, BOX and LINE), simple editing and compatibility with any other FORTH system (you can pull out all those old 70's magazines and use their FORTH programs).

Perhaps the biggest advantage FORTH has over other languages or 'BASIC-additions' (like SIMON'S BASIC or COMAL) is the fact that you can save a "TURNKEY" operation of your programs. This means that they can be run without having to load in the FORTH system, thus allowing you to write programs for people who have not purchased the FORTH system. What they use is essentially a 'run only' subset of the language.

This is by no means an instant way to happiness. Regardless of its many features, there is still the fact that it is another language. If you've only ever programmed in BASIC it will take quite a bit of work getting used to FORTH's way of doing things. For example, FORTH uses reverse polish notation. This system (used on some calculators) adds three and five by having you enter "35+" rather than "3+5". In general, the arguments for operations precede the word. There are no line numbers in FORTH; like Pascal, it is a structured language. This way of programming demands a different kind of thinking than the free linear method you may use in BASIC. These differences, although not insurmountable, should be considered before tackling FORTH.

Performance Micro's FORTH is the version of FORTH which, according to a spokesman for the company, will soon be marketed by Commodore as **Commodore FORTH**. While not a course in FORTH, the included manual documents very well the features of the language. I found it hard to learn the basic concepts of FORTH out of the manual, but later, as a reference guide, it was excellent. Anyone considering a switch to FORTH should first pick up a copy of Leo Brodie's *Starting Forth*, an extremely simple and fun introduction to the language.

*continued on next page*

In summary, FORTH is a good language if you need easy access to the C-64's graphic functions; if you need extra speed in your programs, or if you're just looking for a new, fun language to program in. *TPUG*

*IN A NUTSHELL:*

### **STARTING FORTH**

*By Leo Brodie*

*Published by Prentice-Hall*

*Cost: \$20.00 to \$25.00*

**C64-FORTH**

*Performance Micro Products*

*770 Dedham Street -S2*

*Canton, Massachusetts*

*02021*

*Cost: \$100.00*

## PC-Documate

### *HARDWARE PROFILE*

*Name: PC - Documate Model CM-201*

*Type: Plastic template which fits over the keyboard of the VIC 20*

*Features: Printed on the template is an organized summary of the BASIC language functions, commands and statements, as well as POKE values relating to the screen (inc. color) and to music programming*

*System: VIC 20*

*Summary: A well organized summary of reference data for the VIC 20. Contains a misleading color reference*

*Price: \$12.95(U.S.)*

*Manufacturer: Systems Management Associates*

*Available from:*

*Systems Management Associates*

*3700 Computer Drive, Dept. J-1*

*P.O. Box 20025*

*Raleigh, North Carolina 27619*

### **John McEwen**

*Mississauga, ON*

When I first got involved with PETs, a few years ago, information on how the things worked was hard to come by. Hints gleaned from fellow users accumulated on scraps of paper never to be found if needed. Adam Osborne's *PET CBM Personal Computer Guide* was a welcome arrival but finding things was still hard work.

When our VIC 20 arrived I was astounded. The thing actually had a guide that the uninitiated could read, and start writing programs. Within a week of its arrival my two older boys (10 and 13 at the time) were writing their own stuff and having a grand time!

One problem remains, since none of us have wonderful memories for memory locations, etc., whoever is working on the VIC 20 usually creates a small pile of reference books, ancient but treasured *Compute!s*, *TORPETs* and *Transactors*.

Systems Management Associates offers a product which seemed to have the promise of making our search for ref-

erence material easier. Called the **PC-DOCUMATE**, it consists of a sheet of reasonably sturdy plastic sheet, a bit larger than the VIC 20, upon which information has been printed. The centre is cut out so that it will slip over the VIC 20 keyboard. A small hole has also been cut out so the power light is visible. SMA makes similar products for the C-64 and the IBM P.C.

The VIC 20 version is organized into the following subject areas:

**BASIC Functions**

**BASIC Commands and Statements**

**Colors**

**Control Keys**

**Memory Address Map**

**Music Programming**

**Screen Memory**

A binary to decimal to hexadecimal conversion chart is also tucked away in one corner.

The information is in the main quite similar to that found in references published by Commodore and others. It is clearly printed in type that is large enough to read (for me, that is). I would like to comment on the content of several of the sections.

### **1) Music Programming**

I wish I had had this when I was writing a music editor a few months back. The unique thing about it is the diagram which gives the direct conversion from the location of the note on the music staff to the value to be POKED in to produce that note.

### **2) Colors**

The color table gives the 16 colors *but* gives no indication that only the first eight can be used to POKE character colors. The last eight, when POKEd, give multicolor characters where the indicated color is the second or auxiliary color.

### **3) BASIC Commands and Statements**

In listing the statements SMA has adopted a style found in Commodore manuals (e.g. GET(var1(var2. . .varn))). Read characters from file#). I have always found this style of presentation a bit cryptic but at least in the Commodore manuals there were examples to show the way. I don't know of a better way to do it. However, as presented on this template, I will still have to look things up.

### **4) What is missing.**

I would really like to have a table that compared the character as it appears on the screen with (a) its screen codes, (b) its PETSCII values and (c) the ASCII equivalent (if any). I don't know how it could be fitted in, but it sure would be handy.

### **SUMMARY**

Is this a useful purchase? Perhaps. The misleading color section aside, having the memory maps and the music values close at hand is helpful. The beginner will appreciate the control key summary. However, I notice that as I was writing this article the usual pile of books, etc., seems to have formed around the computer. *TPUG*

---

# THE BEGINNER AND THE DISK - Part II

---

*Or: Things Mother Commodore Never Told You*

**David A. Hook**  
*Barrie, ON*

Herewith the second installment of a series devoted to the elementary aspects of dealing with a disk drive. Because of the popularity of the Commodore 64, the focus will be on the 1541 serial drive, but I'll be including references to the PET/CBM parallel (IEEE 488) disk drives as well.

Unlike the biblical commandments, the ten mentioned last issue (care and feeding of diskettes) are not complete. A couple more came to mind, so I'd like to add these to the list:

11. Never turn off (or on) the power to the disk drive when there is a diskette inside with the door closed. Although my physics training is too far back to explain the theory, it's enough to know that there can be a dandy change to the magnetic field when the current changes. When the door is locked down, the magnetic head is in direct contact with the diskette surface. A sudden change in the magnetic field could cause some pretty nasty signals to get written onto your precious diskette. Some have said that the risk is higher when you power-up. In any case, open the drive door and have a peek before power-up or shut-down. I've been lucky, but a friend of mine has lost data on three disks that way. Since my 4040 dual disk drive seems to have a grounding problem, I've got into the habit of opening the drive door and leaving it that way, except when actually loading or reading data.

12. We veterans of PETdom have had lots of chances to learn from the pros: Karl Hildon was Commodore's technical expert before he moved to BMB to continue *The Transactor* magazine. Karl's favourite recommendation is to close the disk drive door *after* the command has been issued, and *after* the disk starts to spin. The door closure brings down a hub which centres the diskette in the drive. If the door is closed gently, and maybe "wiggled" open and shut a few times, the diskette gets a better chance to be centred, or "registered". After backing-up thousands of disks for TPUG, I am confident that this procedure eliminated scads of bad copies.

Stay tuned, as there will probably be more additions to the commandments. Perhaps you'd like to share one? Pass on your comments (through the Editor) and let all of us benefit.

## The Wedge

As mentioned last time, this program is essential for disk handling when you do not have the luxury of built-in disk commands. The latter would be the case if you have a PET/CBM with Basic 4.0, a VIC 20 with the V-LINK (Richvale Telecommunications) or a C-64 with the BUSCARD (Batteries Included); or the C-64 LINK (Richvale). Other devices and software are also available to provide these commands; i.e. Basic Aid for PET/CBM and Tiny Aid for C-64 (Version 2).

Without the above, I'd venture to say that you are hand-

cuffed in dealing with the disk drive, for two important reasons:

1. Trying to look at the directory of a diskette will cause you to lose the program in memory.
2. When the disk error light flashes (or turns red on the PET/CBM drives) you have no easy way of fetching the error message to see what's gone wrong.

Those are fundamental activities, needed by even beginning programmers. The Wedge programs fill that need quite nicely and sometimes offer many other useful commands as well. If you load the appropriate Wedge when you first turn on your computer, it will be ready for action when you need it. Get into the habit!

The VIC 20 version is included on the TEST/DEMO disk that was packed with your 1541, and it is called "VIC 20 WEDGE". It works on any size VIC 20, with any amount of memory installed. When it is RUN, it automatically relocates itself to the top of BASIC memory and then adjusts the pointers so that BASIC won't clobber it later on. (There is a possible conflict with user-defined characters. But in a programming session you aren't as likely to be RUNNING such programs).

Because the VIC 20 version has been prepared in a special format, it LOADs just like it was a normal BASIC program. This means that you can easily make unlimited copies of it on your own disks. LOAD in the Wedge, swap disks and perform a normal SAVE command to the other disk. I'd strongly recommend that you place a copy as the first program on every freshly-formatted disk. It's only four blocks in size and that way it'll be close at hand for each session.

The C-64 version is in two parts on the TEST/DEMO disk. The first part is identified as "C-64 WEDGE" and the other part is "DOS 5.1". The former is in BASIC and is a so-called "boot" program. When you LOAD the C-64 WEDGE and RUN it, this program performs a LOAD of the second program. This part is the machine language part and it is parked in high memory, *not* where BASIC programs reside. The boot program ensures that the machine language goes where it belongs. Do not LOAD the "DOS 5.1" yourself, as there's nothing to be gained. If you LIST the "C-64 WEDGE" program, you will see what it's doing. The SYS command there causes the Wedge program to be activated.

To place a copy of the Wedge programs on each of your disks poses a problem. Only the "C-64 WEDGE" can be LOADED and SAVED from BASIC to another disk. Unless you are familiar with a machine language monitor (such as SUPERMON 64, by Jim Butterfield), you have no easy way to copy the "DOS 5.1" part to another disk. Bob Fairbairn, the creator of this work, has done an excellent job in adding lots of features to the C-64 version. It is an abomination that

*continued on next page*

he didn't finish the job by making it possible for others to transfer it to their own disks.

If you have a copy of "C-64 WEDGE.REL2", the above diatribe doesn't apply to you. With twenty minutes' worth of effort, I took the two pieces above and merged them. The program is unchanged but the machine language part has been tacked on to the end of BASIC. Now you can LOAD the program, swap disks and do a normal SAVE to get it on your own disk.

Because the need has become quite common, there are a number of Utility programs that allow you to copy selected programs from one disk to another. Jim Butterfield's COPY-ALL series handles the situation when you have two disk drives to work with. His recent UNICOPY program will handle copies with a single disk drive. (The data is stored in the computer's memory, so that you can get the program onto another disk). On the TEST/DEMO disk there is "1541 BACKUP" which copies the whole disk, taking twenty-eight minutes and ten disk swaps to do it with a full disk. (I know two dedicated souls who copied the whole TEST/DEMO disk, then scratched everything but the Wedge programs, just to get them onto their own disks. The benefit made it worthwhile, but what an inconvenience!)

### The Directory

With a BASIC program in memory (one that's dispensable) get a "directory" of the disk with the following command:

```
LOAD "$", 8
```

Now do a LIST and you should see the disk title, its ID and the characters "2A" showing in reverse field. The subsequent lines start with a number (the size of each file, as measured in 256-byte "blocks"). The filenames are shown between quotes and then the file "type" appears on the same line (PRG = program, SEQ = sequential file, REL = relative file andUSR = user file, which is rare). The last line of the listing has a number (from 0 to 664) and the message "BLOCKS FREE.", which indicates the amount of space remaining on the disk.

With that process of checking complete, let's return to the old program. Wait just a minute—where did it go? Look above for the answer. The LOAD of the directory has behaved just like any other LOAD command: it replaces the program in memory with the new one. When you issue the above command, the Disk Operating System (DOS) delivers the directory of the disk to you, but in the form of a BASIC program. The LOAD command obediently stores that program, destroying your old one.

Don't tell me that you haven't a copy of that program! Don't say that you were just checking available space on the disk to see where to SAVE it! That's what everybody says the first time they do it (and the second, and the third. . .)

Sure, Commodore should have provided a non-destructive disk command for the directory, but they didn't. In the military, they call that a "self-inflicted wound". If you only had put in the Wedge first. . .

The following Wedge commands give you a directory that is just displayed on the screen and not saved anywhere. The

SPACE bar will pause the display, hitting it again will resume it and the STOP key will abort the directory. Place the "at-sign" or the "greater-than" symbol in the first column of a screen row:

```
@$(or)
> $
```

Wasn't that relatively painless? LOAD in a BASIC program to repeat the test. Perform the above commands and now LIST the program. Voila! it's still intact. If the directory scrolled off the top of the screen too quickly, just repeat the command.

Those lucky folks with BASIC 4.0 disk commands built-in can just issue the command:

```
DIRECTORY (or)
CATALOG
```

Real English words are preferable, but their function is the same.

What about getting a hard-copy of the directory? I like to run these off on my printer and stick a copy of it inside the disk jacket. This is handy when you have a stack of disks, all unlabelled, as I often do. Precede the commands above with the following set-up:

```
OPEN 1,4 : CMD 1
```

Now enter the appropriate directory command. After the printer stops, finish the job with:

```
PRINT# 1 : CLOSE 1
```

Of course you could have done the 'LOAD "\$", 8' sequence before issuing the OPEN command. Then, after the CMD, you would just say LIST to send the directory to the printer. Finish it the same way.

### Disk Errors

Maybe you forgot to put the disk in the drive. Maybe you tried to LOAD in a program and mis-spelled the name or grabbed the wrong disk. In any case, that error light is blinking madly at you. What went wrong? Without the Wedge in place or a disk-error-handling routine, you're stymied.

To get the current disk status, use the "at-sign" or "greater-than" symbol all by itself. Make sure that it's in the first column of a screen line:

```
@(or)
>
```

A four-part error message will appear on the screen. Without an error present, it will look like:

```
00, OK,00,00
```

A mis-spelled filename will give the following:

```
62, FILE NOT FOUND,00,00
```

The form consists of an error number, a descriptive message, the track on the disk where an error was detected, and the offending sector. Consult the reference manual

*continued overleaf*

for full explanations of what they mean. In any case, the act of fetching the error status will clear the error and cancel the light. A repeat request will give the "OK" message since nothing remains outstanding.

Oops, I almost forgot. With BASIC 4.0, there is a "reserved" variable pair set aside for this purpose. Variable "DS" has the error number alone, while "DS\$" has all four parts as above. Say:

```
PRINT DS$
```

it matches the Wedge command above. You cannot use these variables in your own programs if BASIC 4.0 is active.

But what happens when you have forgotten to put the Wedge into effect and don't have BASIC 4.0 commands? You can turn the disk drive off and back on again (remove the disk first!) and that will get rid of the error. It's not too helpful in finding out the cause and is definitely not recommended when there could be some unCLOSEd files.

You need a BASIC program to get the error message back. My suggestion is that you type this little segment in as the first part of any program that will involve disk handling. That way, it will always be there to use. (Since your program will have to include disk error checking anyway, why not have it for debugging purposes?). I have used high line numbers so it can be appended as a subroutine.

```
62998 END
62999 REM DISK ERROR CHECK
63000 CLOSE 15 : OPEN 15,8,15
63010 INPUT# 15, E1, E2$, E3, E4
63020 IF E1 = 0 THEN 63040
63030 PRINT E1, E2$, E3, E4
63040 CLOSE 15
63050 RETURN
```

Anytime you want to query the disk, just enter "GOSUB 63000" (in immediate mode is fine) and any errors will be reported. If you want to see the "OK" message printed too, just delete line 63020.

See you next issue, as we continue to look at disk commands. Any suggestions for topics to emphasize will be most welcome. I want to make this series as useful as possible, so please let me know. Send correspondence via the Editor. TPUG

#### World's Worst Computer Pun Dept.

1. Have you heard about the nanocomputer that was so small, it didn't use BOOTS, it used BOOTIES?
2. If a dealer were to export chips to Russia, would this be called 'The PARTING of the RED I. C.'s?' TPUG

A. Ylimaki

# COMvoice



## Who says computers should be seen and not heard?

by GENESIS

COMvoice brings you the voice of the future. A speech synthesizer that extends the power of speech to Commodore 64. COMvoice features automatic English-to-Speech conversion. With unlimited vocabulary. And four levels of speech inflection. It's ideal for interaction in educational programs because COMvoice is easily programmed from BASIC.

So why not let your computer speak for itself?

**SPECIAL \$154.95**

W/EXTERNAL SPEAKER \$169.95

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



## ALSO ASK ABOUT OUR HOME SECURITY AND ENERGY MANAGEMENT PRODUCTS

### VIController

Wireless remote control system for the VIC-20 and CBM-64. Use with BSR and Leviton remote receiver modules. **\$89.95**

### COMsense

Input device for the VIC-20 and CBM-64. Provides 4 open/close and 2 analog inputs. **\$64.95**

### COMclock/AUTOboot

Clock/calendar cartridge for CBM-64 with battery backup and auto-start software in ROM. **\$89.95**

Milne's Computer Control Systems 62 Wood Crescent Regina, Saskatchewan S4S 6J7 Ph: 1-306-584-1988

# Flight Simulator II

for the Commodore 64<sup>®</sup> Computer



Put yourself in the pilot's seat of a Piper 181 Cherokee Archer and head for the skies! High-speed color 3D graphics provide a spectacular panoramic view as you practice takeoffs, landings, and aerobatics. Complete documentation gets you airborne quickly even if you've never flown before. And when you think you're ready, you can test your flying skills with the "World War I Ace" aerial battle game. Flight Simulator II features include ■ over 80 airports in four scenery areas: New York, Chicago, Seattle, and Los Angeles, with additional scenery areas available separately ■ day, dusk, and night flying modes ■ user-variable weather, from clear blue skies to grey cloudy conditions ■ full flight instrumentation ■ VOR, ILS, ADF, and DME avionics ■ navigation facilities and course plotting ■ flight handbook and information manual ■ Joystick optional.

**\$69.95**

Commodore 64 is a trademark of Commodore Electronics, Ltd.



**\$49.95**

## VIC-20 SOFTWARE

Miner 2049er (K)	\$44.95
Lode Runner (K)	<b>\$49.95</b>
IFR (Flight Simulator) (K)	\$49.95
MD001 Snakman	\$19.95
CS013 Astroblitz (K)	\$49.95
CS016 Apple Panic (K)	\$49.95
CS017 Choplifter (K)	\$49.95
CS018 Black Hole (K)	\$49.95
ORM004 Crossfire	\$29.95
HESC307 Shamus (K)	\$49.95
Introduction to Basic I	\$49.95
Introduction to Basic II	\$49.95
Frantic	\$19.95
Wacky Waiters	\$19.95

\* Requires 3K exp. (K) Cartridge

## COMMODORE 64 SOFTWARE

Triad (Disk & Tape)	\$19.95
Quazar (Disk & Tape)	\$19.95
Music Machine (cart)	\$ 32.95
Music Composer (cart)	\$ 32.95
Sysres 64	\$79.95
Superbase 64 (Database)	\$129.95
Power 64	\$69.95
Pal 64	\$69.95
Beachhead (disk)	\$39.95
Neutral Zone (disk)	\$39.95
Blue Max (disk)	\$42.95
Grand Master Chess (disk & tape)	\$39.95
Fort Apocalypse (disk)	\$42.95
Survivor (disk)	\$42.95
Jumpman (disk)	\$49.95
Jumpman Junior (cart.)	\$54.95
Shamus (disk)	\$42.95
Lode Runner (disk)	\$49.95
Personality Analyzer (disk)	\$29.95

## BOOKS

Anatomy of the Commodore 64	\$24.95
Commodore 64 Exposed	\$19.95
Sprite Graphics for C-64	\$ 20.95
Machine Language for Beginners	\$18.95
64 Sound & Graphics	\$18.95
PET Basic	\$19.95
VIC 20 Prog. Reference Guide	\$29.95
Elementary Commodore 64	\$18.95
C64 Prog. Reference Guide	\$27.95
VIC Revealed	\$18.95
VIC Graphics	\$18.95
VIC Games	\$18.95
Compute's first book of PET/CBM	\$19.95
Compute's first book of Commodore 64	\$18.95

## COMPUTER SUPPLIES

C10 Cassette Tapes	\$1.00 ea.
C20 Cassette Tapes	\$1.25 ea.
Cassette Boxes	.25 ea.
Verbatim Disks M0525-01	\$44.00/10
Memorex Disks 013481	\$33.90/10
Nashua Disks SS/DD	\$ 29.90/10
K10 Disk Storage Box	\$5.95
Flip Sort Storage Box	\$39.95

## PET/CBM SOFTWARE

POWER	\$119.95
8040—80 col 4.0 sys	
4040—40 col 4.0 sys	

## JOYSTICKS

Kraft Joystick	\$19.95
WICO Power Grip	\$39.95
WICO Track Ball	\$69.95
WICO Red Ball	\$44.95
Commodore Joystick	\$9.95

\*All Joysticks will fit VIC 20 or Commodore 64 computers

## MISCELLANEOUS

### RAMAX

#### VIC-20 Memory Expansion Board

To equal the total memory of RAMAX™ you would have to buy a 16k Memory Expansion, PLUS an 8k Expansion, PLUS 3k Expansion, THEN you would need a "mother board". With RAMAX™ you buy just ONE piece . . . at ABOUT HALF THE PRICE!

\*Adds up to a full 27k bytes of additional RAM to the standard VIC-20's internal RAM of 5k.

**NOW ONLY  
\$175.00**

## ORDERING INFORMATION

Cheque, Money Order, VISA, Master Card accepted. Ontario residents add 7% sales tax. Add 5% for shipping (minimum charge \$1.00).

NOTE: All Prices in Canadian Funds.



# ELECTRONICS 2001 LTD.

5529 Yonge Street, Willowdale, Ontario M2N 5S3  
Tel: (416) 223-8400

---

# FORECASTING WITH THE 8032 - Part III

---

John Shepherd  
Islington, ON

The TBUG February 1984 PET disk, (P)T6, contains four programs which can be used to project historical data into the future. Parts I and II of this series ( *TBUG Magazine*, Issues May and June) provided an overview of time series analysis, defined some of the terminology and discussed use of the data collection program "forst data.8". These articles supplement the documentation on the disk and explain how to use these programs in practical forecasting applications.

The program "forst glm.8" has 12 linear, quadratic or trigonometric forecasting models. These equations are used for medium-range or for long-range forecasts (i.e. where the data must be projected more than three months, and even up to several years, ahead). The models are listed in Table I.

These forecasting equations are all "multiple linear regression models". They are solved by the method of "least squares" and use the historical data that we have gathered. The program fits one of these 12 equations to our data and calculates a "standard deviation of forecast error". This is a measure of how well the equation fits the data. By trying several of the models, we choose the simplest one that has an acceptable error and therefore most reasonably fits the data. We use this model to make our projections.

Let's run an example using our data file "forst demo.d". As we saw in Part II of the series, this is a record of five years' monthly brewery sales, in thousands of cases per month. We will first do the simplest extrapolation possible, that of a straight line through the data, called a "simple linear regression". This is the first equation in Table I.

First, load the program "forst glm.8" and run it. The program will ask for the name of the data file. Type "forst demo.d" and it will load the sixty months of historical data. We should next examine a screen graph of the data. Hopefully we will be able to see whether it looks like a straight line, or is curved, or shows seasonality or cycles; or worst of all, whether it looks like a random pattern of points. This graph gives us our first clue as to which forecasting model might be the most appropriate.

In this case we already know that there is seasonality and that because of this our straight line will have a poor fit, but let's "proceed" anyway. Choose the first forecasting model, the "simple linear". As we can see from Table I, the equation for this model is:

$$x = a + b*t$$

This is the equation for a straight line. The variable "x" is what we are trying to forecast; i.e. the monthly beer sales in thousands of cases. The variable "t" is the time in months, measured from the start of the sixty months of historical data. The parameter "a" is the intercept at  $t = 0$ , and "b" is

the slope of the straight line.

The program examines all sixty points of data and in 21 seconds calculates the intercept and slope of the straight line passing through them that best fits the data. At the top of the screen are two columns headed "Linear Coefficients" and "Std Error of Estimate". The first coefficient, 20.46, is the intercept "a". The second, .132, is the slope "b". The standard errors to the right of these coefficients tell us to what degree of precision we know the coefficients.

The most useful figure on the screen, however, is the "standard deviation of forecast error", 4.3 This tells us that 95% of the data points are within plus or minus twice this of our straight line. Write down this figure because we should be able to get a better fit with another model.

The program then asks if we wish to see a graph of error vs. time. As expected, the error cycles about the zero mid point, indicating that we need a better model that takes seasonality into account. Let's try the second model, called "linear plus additive seasonality".

This model is just a little more complicated. Its equation is shown in Table I and does take into account seasonality. Because the data is monthly, we have decided that the period of seasonality is 12. The equation for this model is the same as the previous simple linear model with the addition of one of eleven other possible constants (c to m), depending upon which month of the year is being calculated. The equation for each January is  $x = a + b*t$ . The equation for each February is  $x = a + b*t + c$ , and that for each March is  $x = a + b*t + d$ , and so on. The equation for each December is  $x = a + b*t + m$ .

When the program calculates a best fit for this equation, it has to solve for thirteen parameters, instead of the previous two. It therefore takes much longer to calculate (about 7 minutes). Also, since we have the same 60 data points and many more parameters to calculate, each parameter can be calculated much less precisely. However, we have a much better fit since the standard deviation of forecast error is 1.8, compared to 4.3 with the previous model.

There are other ways to handle seasonality with fewer parameters. Equations 3 to 6 and 9 to 12 use trigonometric functions. Equation 3, for example, has only four parameters, takes only about one minute to calculate, and has a standard deviation of forecast error of 2.2.

Quadratic functions can plot a curved trend line with less error than can linear functions. The simplest such quadratic function is equation 7. Note that equations 7 to 12 are quadratic versions of equations 1 to 6. For our beer sales data, the quadratic equations give a slightly better fit. Try several other models. Examine the error graphs. The model that fits our data the best is no. 8, but it is also the most

*continued on next page*

complex and takes the longest to calculate. For most purposes, a model such as no. 9 or 10 would be far more convenient and quite adequate.

Once the model has been chosen, the program will then calculate a forecast for any reasonable period in the future. These seasonal models should only be used for medium

term forecasts. For long term forecasts, use the simple linear or simple quadratic models.

In the next article we will look at the program "forest ma.8". This program produces short term forecasts using moving averages and seasonality factors. *TPUG*

## DISKALIGNER - A User's Review

Chris Bennett  
*TPUG Business Manager*

**Diskaligner** is a program designed to allow anyone to align the read/write heads of the 1541 disk drive. The program is available from G.R.Q Marketing, P.O. Box 197, Thornhill, Ontario, Canada, L3T 3N3 for \$29.95 Canadian (plus 7% in Ontario).

Using this program, one Sunday afternoon, I was able to align the read/write heads of two 1541 and two 2031 disk drives. (The 2031 is the single disk drive for the Pet/CBM line of Commodore computers).

All I needed was a medium-sized Phillips screwdriver and a very small slot screwdriver (to set the timing). First, I removed the power cord (*very important*) and the serial bus cable. Then I took out the four screws that hold the cover in place. Next, there are 6 more screws plus a small cable connector to remove before the drive assembly can be lifted out of the case. The lower casing is now used to support the drive unit while aligning the heads.

The power cord and serial bus cable are re-connected to the disk drive. The **Diskaligner** diskette is inserted and the program loaded into the Commodore 64. The first step is to test the disk rotation speed. This is very easy since there is a stroboscopic disk showing through the bottom of the drive chassis. Under normal (60hz) artificial light, the outer rings should appear stationary. A small slot screwdriver is all that is needed to adjust the speed if any problem exists.

In aligning the read/write head, the STEP MOTOR must be adjusted to meet two criteria. *One*: the head must be centered over the tracks. *Two*: the STEP MOTOR must be able to read all 35 tracks.

### Test Criteria 1

The best way to check whether or not the head is centered over the tracks is to VALIDATE a good diskette. The one I used was the '1541 TEST/DEMO' disk supplied with the drive. (NOTE: the diskette must be write-protected and cannot have been written to by your 1541.) The following program will enable you to test your drive to see if it is ok.

```
10 OPEN 15,8,15
20 PRINT#15,"V0"
30 INPUT#15,A$,B$
```

```
40 PRINT A$,B$
50 CLOSE 15
```

Enter the above program, insert the '1541 TEST/DEMO' diskette and type RUN. Now watch the red light on the front of the disk drive. It should be red and should not flicker until the message '26 WRITE PROTECT ON' appears on the screen. If the red light does flicker, then the drive needs to be aligned. If you can hear a 'tapping' noise, then it must be done as soon as possible.

### Test Criteria 2

An easy way to test whether the STEP MOTOR can read all 35 tracks, is to run the 'PERFORMANCE TEST' on the '1541 TEST/DEMO' diskette. One of the checks that it performs is to read and write to both track 1 and track 35. If your drive passes this performance test, then the second criteria for head alignment has been satisfied.

If both these procedures are repeated every week or so, you will be able to catch any head alignment problems before they get too bad.

The **Diskaligner** allows you to align the read/write head of your disk drive whenever it is needed. Since it can cost between \$35 and \$70 to have this done by a dealer, the \$30 for the program is well worth the price. One word of caution. Dismantling your disk drive will void your warranty. However, if you are still within the warranty period, you can get the disk aligned free of charge. *TPUG*



# C 64 PROVINCIAL PAYROLL

A complete Canadian Payroll System for Small Business.

- 50 Employees per disk (1541) •
- Calculate and Print Journals • Print Cheques • Calculate submissions summary for Revenue Canada •
- Accumulates data and prints T-4s • Also available for 4032 and 8032 Commodore Computers.

Available from your Commodore Dealer.

Distributed by:

**M**ICROCOMPUTER  
SOLUTIONS  
1262 DON MILLS RD. STE. 4  
DON MILLS, ONTARIO M3B 2W7  
TEL: (416) 447-4811

# BENT COMPUTER ASSOCIATES

\*

## Software

<b>SG01 Pro-Pick Basketball</b>	<small>Not a Game! Picks pro or amateur teams when odds are in your favor. List data from newspaper and generate program.</small>	<b>19.95*</b>
<b>SG02 Pro-Pick Baseball</b>	<small>Similar to SG01</small>	<b>19.95*</b>
<b>SG03 Pro-Pick Football</b>	<small>Not a Game! Similar to SG01, but uses different algorithms and data.</small>	<b>19.95*</b>
<b>SG04 Pro-Pick Hockey</b>	<small>Similar to SG03</small>	<b>19.95*</b>

\* C64 - Please specify tape or disc version - Coming soon for Vic20 (8K)

## Hardware

<b>HGR01 Video Plug</b>	<small>Improves C64 output on your 8" W TV can be used as a monitor.</small>	<b>6.95*</b>
<b>HGR02 Speaker Plug</b>	<small>External speaker for C64 or VIC20 when your monitor doesn't have one.</small>	<b>12.95*</b>
<b>HGR03 Combo Plug</b>	<small>Combines features on both of the above units.</small>	<b>18.95*</b>
<b>HGR04 Reset Button</b>	<small>Unlocks C64 or VIC20 following system crash without using your program.</small>	<b>19.95*</b>
<b>HGR05 RS232 Interface</b>	<small>Enables C64 or VIC20 access to printers, modems, etc. thru the user port. Cable not included.</small>	<b>29.95*</b>

\* Easily attaches to rear unit. Contains complete instructions.

## BENT Computer Associates

1542 Front Street

Slidell, Louisiana 70460



Check, Money Order, VISA, MasterCard (No C.G.D.)

Add 12% Shipping, Louisiana Residents add 6% Sales Tax

VIC 20 and VIC 20/50 are trademarks of Commodore Business Machines, Inc.



## The key to the system — simple but sophisticated

The problem with many computer systems is that they can be too complex, too confusing and too cumbersome. So, when we started to design this system, we decided it wouldn't fall into any of these traps.

- Once your records are set-up, the power of the computer can be used to manipulate them at the press of a single key.
- You can store, standard letters together with selected information. This letter may be sent to one, all or selected people — you choose.
- The file can be manipulated by a SEARCH feature which sorts out selected records according to your requirements.
- The CHANGE KEY feature allows a data file to be re arranged (sorted) to a different sequence on any field.
- If you can handle a manual index-card file, then you can handle this computer system.

Distributed by:



## Searching Techniques

---

Chris Bennett  
TPUG Business Manager

This month's tutorial deals with the searching of large string tables. A table is a list of data, numbers, etc., systematically arranged. For example: a list of all the available inventory items could be stored. In a program, that table could then be searched to check for valid inventory codes.

There are many ways to see if certain values are in a table. I will deal with two: a sequential search and a binary search (see Listing 1).

Both of the above assume that the table is sorted in ascending sequence. A binary search always requires this. However, a sequential search does not require that the table be sorted if line 4030 is removed from the listing. If this is done, the entire table is search from beginning to end until either the item is found or the end of table is encountered.

Let us assume the following:

KY\$(1000) is the table being searched  
K\$ is the search argument  
NV is the number of items in the table  
P is ZERO if not found or a value from 1 to NV indicating the position within the table that the item has been found

In the sample program, there are four sections of code:

120- 200 Defines the array KY\$ and fills it with 1000 values from aaa to jjj.  
1000-1140 This allows you to test the time it takes to find a value in the table using either search method.  
4000-4060 This is the sequential search routine.  
7000-7080 This is the binary search routine.

When the program is run, KY\$ is filled with 1000 values starting at 'aaa' and ending with 'jjj'. After about 30 seconds, the prompt 'S/B,KEY' shows on the screen. Enter two values separated by a comma. The first value must be either 's' for sequential or 'b' for binary search technique. The second value is the three character code to find in the table. ie 'b,dce' does a binary search of the table looking for the value 'dce'. After the search is completed, the time taken to find the key is printed (in seconds). 'not found' is printed if the value is not in the table.

Some sample timings, in seconds:

Key	Binary	Sequential
aaa	0.34	0.03

faa	0.03	6.33
jjj	0.31	12.73

As you can see, a sequential search is only fast if the item you are looking for is near the beginning of the table. The binary search, on the other hand, will find any item in the table in less than a third of a second.

In the sequential search when the table is in ascending order, the search argument K\$ is compared to each table item from 1 to 1000. If a match is found, then P is set to the subscript of the table item and the routine returns back to the main program. When a value in the table is found to be greater than the search argument, P is set to ZERO to indicate a 'not found' condition. The reason for this is that subsequent values will all be greater since the table is sorted in ascending sequence.

For the binary search, the table must be in ascending sequence (smallest first) or descending sequence (largest first). In listing 1, the table is assumed to be in ascending sequence.

The search argument is first compared to the middle item in the table. Depending on whether it is greater than or less than that item, it can be determined that the search argument belongs in either the upper or lower half of the table. That half of the table is then treated as if it were the whole table and the process is continued. Since the number of items which must be searched is reduced by one-half each time, a match (or no-match) is found very quickly. In a table of 1000 items, no more than 10 such searches or compares are needed.

In the program sample, I and J point to the bottom and top of the range of keys being compared (Line 7010). If I is ever greater than J (Line 7020), then no match has occurred and P is set to ZERO and processing has finished.

The mid-point of the table is then calculated in line 7030 and saved in K. The K'th element of KY\$ is then compared to K\$ (Lines 7040 to 7060).

If equal, the item has been found and P is set to the array position and processing is complete.

If K\$ is less than the K'th item, then we only look in the bottom half of the table. To do this, the top of table pointer (J) is set to one less than the mid-point (K) just calculated.

If K\$ is greater than the K'th item, then we only look in the top half of the table. The bottom of table pointer (I) is set to one greater than the mid-point (K). We then go back to line 7020 and repeat the whole process. TPUG

*continued overleaf*

```

120 NV=1000: L=1
130 DIM KY$(1000)
140 :
150 FOR I=65 TO 74
160 FOR J=65 TO 74
170 FOR K=65 TO 74
180 KY$(L)=CHR$(I)+CHR$(J)+CHR$(K)
190 L=L+1
200 NEXT K, J, I
1000 :
1001 REM ** MAIN LOOP OF PROGRAM **
1002 :
1010 INPUT "S/B,KEY";CD$,K$
1020 K$=LEFT$(K$+" ",3)
1050 IF CD$="S" THEN TT=TI: GOSUB 4010: SS=TI-TT
1080 IF CD$="B" THEN TT=TI: GOSUB 7010: SS=TI-TT
1090 WK=INT(SS/.6)/100+.001
1100 WK$=STR$(WK): WK$=LEFT$(WK$,LEN(WK$)-1)
1110 PRINT TAB(15-LEN(WK$));WK$;" SECS ";
1120 IF P=0 THEN PRINT "- NOT FOUND.";
1130 PRINT
1140 GOTO 1010

4000 :
4001 REM ** SEQUENTIAL SEARCH **
4002 :
4010 P=0
4020 FOR K=1 TO NV
4030 : IF KY$(K)>K$ THEN P=0: K=NV
4040 : IF KY$(K)=K$ THEN P=K: K=NV
4050 NEXT K
4060 RETURN
7000 :
7001 REM ** BINARY SEARCH **
7002 :
7010 I=1: J=NV
7020 IF I>J THEN P=0: RETURN
7030 K=INT((I+J+1)/2)
7040 IF K$<KY$(K) THEN P=K: RETURN
7050 IF K$>KY$(K) THEN J=K-1
7060 IF K$=KY$(K) THEN I=K+1
7070 GOTO 7020
7080 RETURN
READY.

```

\*\*\*\*\*

# Take A Basic BASIC Course !

\*\*\*\*\*

The following is the course outline for an introduction to BASIC programming, running this summer. These sessions will be held on alternate Thursdays, July 12 & 26, Aug. 9 & 23, and Sept 6 at the York Public Library, 1745 Eglinton Avenue West, 7:30 pm to 9:30 pm.

The course will be taught by Chris Bennett, TPUG Business Manager. It is available to all TPUG members at no cost. No previous knowledge or experience is required, just a willingness to learn. Members may attend any one session, or all sessions as they desire.

## 1. July 12th — Introduction to BASIC

- Structure of BASIC
- Data Types
- Variable Names
- DATA , READ
- INPUT, PRINT
- Arithmetic

## 2. July 26th — Arrays, Loops and IF

- DIM Statement
- Subscripts
- Loops (FOR .. NEXT, GOTO)
- IF ... THEN

## 3. Aug 9th — File Structures

- OPEN, CLOSE
- INPUT#, PRINT#, GET#
- Device Numbers
- Secondary Addresses
- Tape, Disk and Printer files

## 4. Aug 23rd — Disk Operations

- Sequential Files
- Disk Commands (ie. Format, Rename etc.)
- Directory Use
- Relative Files

## 5. Sept 6th — Strings and Logic

- Logical Operators
- Subroutines
- String Manipulation
- LEFT\$, MID\$, RIGHT\$
- CHR\$ -- ASC
- STR\$ -- VAL
- LEN TPUG

## CAPTAIN COMAL™ STRIKES BACK

### THE CAPTAIN RECOMMENDS:

- **COMAL TODAY NEWSLETTER**  
Sample Copy: \$2  
Subscription (6 issues): \$14.95
- **COMAL TODAY DISKS**  
Individual Disk: \$14.95  
Subscription (6 disks): \$59.90
- **COMAL HANDBOOK**  
Reference Book, 334 pages: \$18.95  
Book and Disk with programs from book: \$39.90
- **FOUNDATIONS IN COMPUTER STUDIES WITH COMAL**  
Tutorial Text, 313 pages: \$19.95  
Book and Disk with programs from book: \$39.90
- **STRUCTURED PROGRAMMING WITH COMAL**  
Tutorial Text, 266 pages: \$24.95  
Book and Disk with programs from book: \$39.90
- **BEGINNING COMAL**  
Tutorial Text, 333 pages: \$19.95  
Book and Disk with programs from book: \$39.90
- **CAPTAIN COMAL GETS ORGANIZED**  
Applications Tutorial, 102 pages  
Book and Disk package: \$19.95
- **STARTING WITH COMAL**  
Tutorial Text: \$18.95 (due in stock June 1984)

All items stocked for quick delivery. All Disks are 1541/4040 format. Send check or Money Order in US Dollars plus \$2 shipping/handling per book to:

**COMAL USERS GROUP, U.S.A., LIMITED**  
5501 Groveland Ter., Madison, WI 53716  
phone: (608) 222-4432

TRADEMARKS: Commodore 64 of Commodore Electronics Ltd.; Captain COMAL of COMAL Users Group, U.S.A., Ltd.

## ATTENTION PROGRAMMERS

# THE SOFTWARE LAB

A DIVISION OF FAHNKE & DUFFY INTERNATIONAL INC.

IS LOOKING FOR NEW AND INNOVATIVE SOFTWARE FOR THE COMMODORE 64!!

IF YOU HAVE AN INTERESTING GAME OR USEFUL PROGRAM NEAR COMPLETION, **COME TO US!!**

WE'LL HELP TO PUT ON THE FINISHING TOUCHES, PACKAGE, ADVERTISE AND DISTRIBUTE YOUR HARD WORK, WHETHER IT IS A NEW GAME, USEFUL UTILITY OR A BUSINESS APPLICATION.

YOU DO WHAT YOU DO BEST—PROGRAMMING.

WE DO WHAT WE DO BEST—GETTING YOUR PROGRAM ON THE MARKET, SELLING!

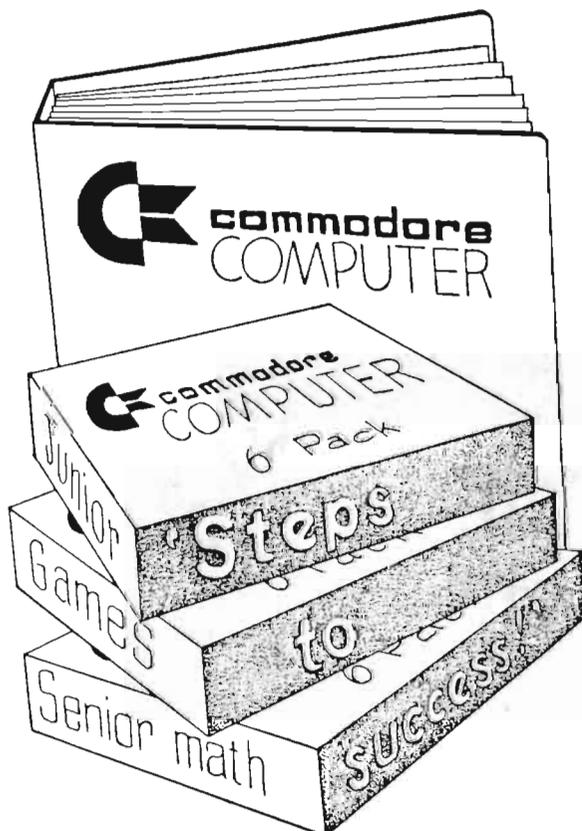
FOR MORE INFORMATION CALL US AT:

**(416) 233-2259**

**ASK FOR DEAN BROWN  
(CANADIAN MANAGER)**

**THE SOFTWARE LAB  
5170 DUNDAS STREET WEST  
ISLINGTON, ONTARIO, M9A 1C4**

## “Educational Software For Today’s Child!”



Approved and evaluated by educators

# AURORA

## SOFTWARE INC.

P.O. Box 1394  
Haileybury, Ontario P0J 1K0  
(705) 672-5517

Exclusive Distributor Of:

- \*The Commodore Educational Software Package
- \*The Ontario Software Educational Project
- \*Aurora's Own "Six Pack"

*A Canadian Company Distributing  
Canadian Produced Software*

---

---

# COMMODORE DISKETTE COMPATIBILITY

---

Jim C. Halsey  
Racine, WI

How many times have you used a diskette in a 1540/1541 disk drive, or a 2040/4040 disk drive, that seemed to have a will of its own? If you thought a 'glitch' or gremlin was responsible for the way a diskette acted, read on to find out what the reasons are likely to be.

There has been quite a bit of discussion regarding the types of diskettes to use in Commodore drives and reasons why 1540/1541/2031 and 2040/4040 formats do not seem to be read/write compatible. This article should help clear up some of these questions and problems.

The diskettes used in the 1540/1541, 2031 and 2040/4040 disk drives should be single-sided double-density, with 48 TPI (tracks per inch) structure. Some of the Commodore documentation calls for single-density diskettes. I'm sure this is an oversight on someone's part. A document from Shugart, part no. 39229-0 12/81 (the manufacturer of the mechanical drives in 2040/4040 drive systems) specifies using high resolution or double-density diskettes as well as high resolution read/write heads with Group Code Record (GCR) format operating systems. This is the format Commodore chose, which does the job admirably, even though it is a little more difficult to implement. Sometimes, it may be possible to use single-density diskettes with this format, if the diskettes are of excellent quality and the drive is designed to compensate for the more critical requirements of Group Code Record format. Some people may say "I've been using single-sided, single-density diskettes all along, and never had any trouble". This may seem to be the case, but there usually comes a time when a user starts writing or modifying programs as well as just copying programs from one diskette to another. When this happens, you start to do many reads and writes to/from the same area on a diskette. This increases the chance for diskette-induced data error, or a loss of data on a diskette. This will show up in a number

of ways, such as a program missing a line number, putting strange characters in random locations, or just not being able to read a data file.

The following pictures (fig. 1 through fig. 4), are enlarged photographs, taken with a camera mounted on a high power microscope, of formatted diskette surfaces after depositing iron powder particles on the magnetized tracks (magnetized by the read/write heads). Fig. 1 and fig. 2 are of Cromemco diskettes, fig. 1 of a single-density format, and fig. 2 of a dual-density format. Cromemco uses the IBM 3740/3741 data format for single-density recording and the IBM dual-density format. These are included so you can compare the physical differences between the standard single and dual-density data formats and Commodore formats. The lines that are perpendicular to the tracks are the actual magnetic changes in the diskette surface, or data cells. The same disk drive was used to first format a diskette in single-density format, then one in dual-density format. Using these as a guide shows us that the 1541 format (fig. 3) is comparable to dual-density, as far as the rate at which it writes data on the diskette surface; and the 4040 format (fig. 4) is even more dense than the dual-density format (the 8050 format is even more dense than the 4040 format). You'll notice the tracks are of the same center-to-center dimension on the 1541 (fig. 3) as those on the 4040 (fig. 4). The major difference is the guard band, or dark area between tracks, and the closeness of the perpendicular lines of magnetic information, or rate in which data is written. The guard band is an area that is DC voltage erased (by the read/write head) as the read/write head writes data on the surface. The purpose of the guard band is to keep track data from mixing with an adjacent track's data, or eliminating crosstalk. When a drive becomes 'out of alignment' it is either mixing track data or the read/write head is actually covering portions of two tracks at the same time, instead of one. This can happen just from

*continued on next page*

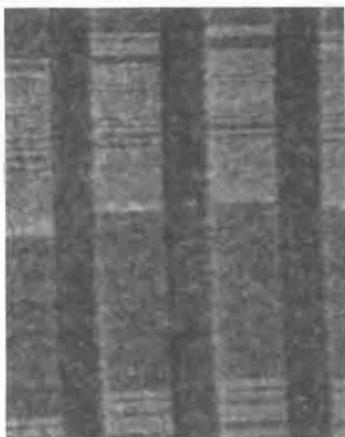


fig. 1



fig. 2



fig. 3

using a drive, let alone from carrying it about and subjecting it to a physical jarring or bouncing.

Fig. 6 is the surface of a 4040 format diskette as seen from the microscope, not enlarged by photography. This is a section of the surface showing all read/write tracks (1 through 35) as well as the residual tracks from testing the diskette from the factory. These innermost tracks that are lighter in intensity would be used if the Commodore format was 40 tracks vs. 35 tracks. Fig. 5 is the same format, with the diskette having been 'bulk erased' with a magnet before formatting. This shows how to effectively remove 'all' data on a diskette surface.

If a drive is designed to compensate for the more stringent requirements of Group Code Record format, then the chances of using single-density diskettes with fewer problems are better, which seems to be the case with the 2040/4040 drives. Now with the introduction of the 1540/1541 drives, some problems became more pronounced, the most obvious being reading and writing incompatibility, (or so it seemed for the majority of users) between 4040 format diskettes and 1540/1541 format diskettes. Some attempts to format a diskette on a 1540/1541 drive, then write on this same diskette in a 4040 drive worked, sometimes. Problems reading diskettes between types of drives were also obvious. The problem stems from the different width of the data cells on the diskette surface. If the disk operating system is told to write a block of data using 1541 format, the cell structure is wider, thus covering more area in the header data for that block. Now if you should write to that same area using a 4040 format, you will fit header data in a shorter section of that track, so when an attempt to read that data is made, some of the residual 1541 data may be interpreted as header data in addition to the normal header data. This makes the disk operating system think it has a bad block, thus generating a disk error. On some occasions there may be compatibility, but this is primarily when working with short files.

Should one start experiencing problems using one's disk drive/drives, the average user can do no more than run the performance test supplied by Commodore on the diskette that comes with one's disk drive/drives. If your disk drive

fails this test, or seems to be incompatible with other drives, it may be out of alignment and in need of care from a qualified technician. This is the major source of difficulty with the 1540/1541 drives.

In order to lessen the effect of these problems, it is best to align the 1540/1541 disk drives to the same specification used with the 2040/4040 drives. Commodore didn't have a published specification for alignment procedures on the 1540/1541 drives as of December, 1983. The alignment procedure I've used with success is the same as the procedure for the SA-390 drives, namely adjusting the physical positioning of the stepper motor, or track 17 position (radial alignment) and track 1, or outermost track, stop position. Before alignment is attempted, the mechanical positioning elements should be checked for stability, as well as the motor speed, using the stroboscope on the bottom of the spindle pulley. The stroboscopes on 1541's and 4040's are identical, as you can see from fig. 7 and fig. 8. A drive belt for a Shugart SA-400 drive is interchangeable with the Alps drives in 1540/1541's as well as the Shugart SA-390's (SA-390's are in the 2040/4040 drives). Also, the Shugart pressure pads are able to be used in the Alps drives. The hub, or pulley on the shaft of the spindle drive should be checked for stability (some of the 1540/1541 spindle drive pulleys have no retaining screw, and have come loose, causing a scraping noise when the drive motor spins). The physical track zero stop (track 1 for DOS read/write purposes) should also be checked for stability. Some of these mechanical parts can be secured in the adjusted position by using a substance such as Glyptol. Do *not* use Super-Glue or similar-type glue. Fingernail polish will also serve to keep screws from loosening due to vibration and stress. These parts are what set up the reference for track 1 (the first read/write track). Having aligned a number of 1541 drives, I've found the Shugart specification for the 'lobes' or pre-recorded data on the alignment diskette, proves to be a much more critical adjustment with the 1541's than the 2040/4040 drives. Using a factor of 90% or better for the 'Lobes' or 'Cat's Eye' pattern, versus 70% with 2040/4040 drives, gives more reliable

*continued overleaf*



fig. 4

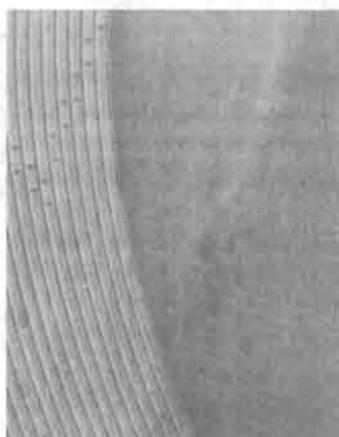


fig. 5



fig. 6

performance. If you don't know what these terms represent, please don't attempt to align your disk drive (this waveform is generated by a pattern that is recorded in an elliptical shape, on an alignment diskette only). *Disk drive alignment is no easy task, so it is well left to persons that are knowledgeable in disk drive alignment procedures, as well as having the proper test equipment and alignment diskettes.*

Should you be using diskettes that are used in both the 1540/1541 and 2040/4040 format, you should bulk erase (or erase with a DC magnet — a large speaker magnet works fine) the surface of your diskettes before formatting in either drive. If you consider the spacing of data on the two formats [refer to fig. 1-6], and consider as well, the idea of having a 1540/1541 formatted diskette with a few files stored on its surface, then erase these files (all you actually do is change the directory file type byte). This leaves all the file data on the diskette surface. This is why the two formats are not truly read/write compatible. In most cases, the two formats will read the other diskettes, but for reliable use, writing on the other format is not a good practice.

Requirements of diskettes for the 8050 and 8250 disk drives should be given the same considerations as the 4040 drives, in addition to having a structure of 96 tracks per inch vs. 48 TPI on a 1541/4040 surface. There are 77 tracks used per read/write surface on the 8050 and 8250 drives. The 8250 has two read/write heads, one on either side of the diskette. This requires having dual-sided diskettes, of which the major diskette manufacturers guarantee testing both sides. These are used in dual-sided drives with the same side facing up, and never 'flipped', as in some advertisements that say you can use both sides of the diskettes. If you think of the dust and/or smoke particles that can get inside the jacket of your diskette as 'boulders' with respect to the diskette surface (see fig. 9), imagine the grooves these boulders are wearing into the jacket liner material, let alone the diskette surface; then picture reversing the direction of travel of the diskette, or back against the grain of these boulders. That is pretty scary if you consider the time invested in diskettes from writing or editing programs and data.

What all this means is that diskettes in Commodore drives are required to work (store read/write data) at a rate that is at least 20% more demanding than the standard IBM 3740/3741 single-density format. To make more reliable recording of information, or files, the 1540/1541 drives should be aligned to 2040/4040 specifications, and it would seem that for all applications, the use of dual-density diskettes would be preferred, and certainly recommended. Now for menial, or less important, program diskettes, some users may choose to use single-density diskettes. If you should so choose, be prepared to possibly lose a file, now and then. This can happen after the diskette has been formatted and files written and verified. To realize the probability of this happening, think of a magnet you haven't used for some time that has been stuck to a metal surface. It eventually will lose some of the magnetic force. This same process can happen to diskettes over a period of time. The better the oxide surface, the longer the data will stay intact. For these reasons, I prefer to use dual-density diskettes for all applications, including VIC 20 and Commodore 64 files with the 1541 drive, and

2001/8032 files with 4040 drives. This saves problems with keeping the two types of diskettes separate, as well as giving me the confidence in my computer systems, which is necessary when working on a program for any length of time, then backing up the latest revision to this program, with no fear of having lost those hours, or sometimes days of work. Of course, everyone does keep backups while writing or revising a program, right ??? !!! *TPUG*

#### References:

*Commodore Business Machines, Inc.*  
*Disk Reference Manual, 1982*  
*5¼-inch Disk User's Manual, 1980*  
*1541 User's Manual, 1983*

*Cromemco CDOS Manual*  
*part no. 023-0036, 1980*

*Digital Equipment Corporation RK06/RK07 Disk Drive*  
*User's Manual, 1978*

*IBM Diskette General Information Manual*  
*part no. GA 21-9182-5, 1980*

*Shugart Format Manual — 5.25 inch Floppys*  
*part no. 39229-0, 1981*



fig. 7

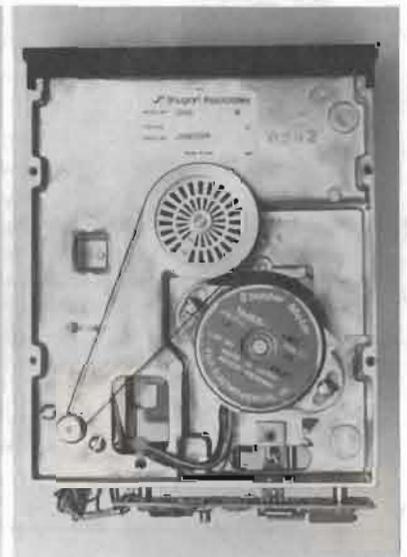


fig. 8

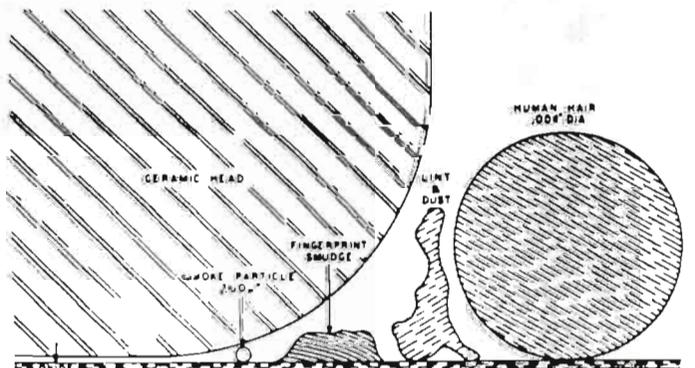


fig. 9

---

# SuperPET 6809 Assembler -- Part II

---

Brad Bjorndahl  
*Bramalea, ON*

This article continues the development of the program **PITS**, to be implemented by the Waterloo 6809 microAssembler. The previous article put forth some design questions and two of the initial modules. Some of the questions, such as the status of a bounded zombie, are not actually relevant to the program as I finally designed it, but they had to be considered. Also, when changes are made, the questions may become problems and thinking about them now will help make enlightened decisions. For example, a change which allows a zombie to move more than once per turn, or to have multiple lives, may lead to zombies which have more moves left on a turn but which are bounded by other zombies. In any case the relevant questions have been answered in the design shown at the end of this article.

The first module, called **Mainline** in the last article, is really a brief description of the entire game. The procedure 'Create Island' includes displaying island boundaries; filling it with pits, zombies and the player, and initializing several arrays. Creating the island was the single most difficult task in the game design because it was so poorly defined. It was necessary to imagine in detail how a player would want to instruct the program to position pits and zombies. The problem was the classical one of interfacing human and machine. The method I settled on is as follows: the user enters a string of characters. The program initializes a pointer to the top left position of the island and moves the pointer over the island left to right and top to bottom, like a raster scan. Each character in the input string is treated as if it is an unsigned binary number and the scanning pointer moves as many positions as indicated by that number. Therefore each character moves the pointer a different number of positions. Also, if the character happens to be a 'z', 'o' or 'y', then the program will place a zombie, pit or player respectively wherever the pointer happens to be. Once the player is positioned, the island is considered complete. This method allows the player to position as many (or as few) zombies and pits as desired and allows the positioning to be as regular or as (pseudo-) random as desired. There are two loops in the 'Create Island' procedure, one nested inside the other. The outer loop is executed once for each string of characters (commands) until the player has been placed. The inner loop is executed once for each character within the current string until the string is exhausted or the player has been placed. I presume that the comments and indentations make this clear. The inner loop contains GUESS-ENDGUESS structures which are used to select various 'cases' for action. This structure is basically a number of independent sections of code which are separated by ADMIT statements. The code is executed sequentially as normal until 1) a QUIT is encountered, in which case execution jumps to the start of the next section (i.e. a new 'case') or until 2) an ADMIT is encountered, in which case execution

skips all remaining sections and resumes after the ENDGUESS statement. Normally the programmer will start each section with a test to see if that section should be executed and if not, then QUIT that section and check the next one. 'Create Island' has a nested GUESS-ENDGUESS because I did not want the carriage return to be interpreted as a command. It is not necessary in the design step to worry about how all these loops and guesses will eventually be coded. It is sufficient that the structure is clear but, to avoid seeming high-minded, I will admit that the Waterloo assembler makes the step from design to code almost easy.

The **Mainline** module repeats each step of the game within a loop where each step consists of a player move followed by all the zombie moves. The module 'Move Y and Z' summarizes this. Note that from this point on any action must be followed by a test to see if the GAME-END flag is true. All moves, whether the player or zombies, are done by first calculating the new position, determining the result of such a move, and only then, if appropriate, making the actual move. A different order of operations would result in pits and boundaries being arbitrarily removed.

The 'Move Y' module is straightforward. The only difficulty is the sub-module 'Get new Y position', which of course requires player input. In this case only a single character is accepted, of which only the right hand nybble is used. The GUESS structure calculates the new player position according to the relative direction on the computer keypad where 5 is the current player position. If, for example, the player enters a character '3', which is ASCII \$33, the the leading nybble is replaced with 0 and the result 03 is treated as a binary number. The number 03 is satisfied after the second ADMIT, resulting in the player move down and to the right.

The 'Move Z' module is structurally similar to the 'Move Y' module but has more cases. The outer loop is repeated until all the zombies have moved and the inner loop is repeated once for each zombie. The problem that arises here is that the zombies are considered in whatever order that they were entered. Therefore it may be that a zombie cannot move until another one has moved. For each zombie there is a Z-move counter which is zero after the zombie has finished moving and the outer loop will continue until all the counters are zero.

There are a few modules which I have not included because they are machine- or programmer- dependent. One module is necessary to get a string of commands. There are so many ways of doing this that I simply leave it to the users to design their own routine. Another one is a routine to calculate an effective memory address from a row/column position. This will allow 'peeking' at a position to see if it is occupied. Other necessary routines are ones which will compare one

*continued overleaf*

---

row/column position against a list of others and one which will sum a list of numbers. The source code, which will be added to a SuperPET library disk, will contain my versions of these routines.

The design of the program **PITS** is essentially complete. The 6809 assembler coding details follow in the next articles.  
*TPUG*

---

```
Create island
  Clear screen
  Put cursor at row 24 column 1
  Display boundaries
  Initialize all Z and O position arrays to 0
  Set position pointer to top-left of island i.e. row 2 col 2
  Loop ; outer loop for each string of commands
    Put cursor at row 24 column 1
    Get a string of commands
    Loop ; inner loop for each individual command
      Guess
      If not at end of string then quit
    Admit
      Calculate new island position
      Guess
        If command is not a 'z' then quit
        Add a zombie to the island
      Admit
        If command is not a 'o' then quit
        Add a pit to the island
      Admit
        If command is not a 'y' then quit
        Add the player to the island
      Endguess
    Endguess
  Until at end of command string or the player is added
  to island
  Until the player is added to the island
  Initialize pit depths
  Initialize zombie life counter
  Clear message display area
  Return
```

```
Calculate new island position
  Save the command as a 1 byte binary counter
  Loop ; outer loop
    Get the current island position
    Loop ; inner loop
      Increment column
      If column = 80
        Set column to 2
        Increment row
      If row = 23
        Set row to 2
      Endif
    Endif
  Decrement (command) counter
  Until counter = 0
  Save the newly calculated position
  Set the counter to 1
  Until the new position is unoccupied
  Return
```

```
Move Y
  Set y-moved flag to false
  Loop
    Get new Y position
    Guess
      Compare new Y position to all O positions
      If no match then quit
      If the pit is full then quit
      Set game-end to true
    Admit
      Compare new Y position to all Z positions
      If no match then quit
      Set game-end to true
    Admit
      Compare new Y position to boundary positions
      If match then quit
      Move Y to new position
      Set Y-moved to true
    Endguess
  Until Y-moved is true or game-end is true
  Return

Get new Y position
  Get a character from the player
  Set the new position equal to the current position
  Set leading nybble to 0
  Guess
    If not 1 then quit
    Increment row
    Decrement column
  Admit
    If not 2 then quit
    Increment row
  Admit
    If not 3 then quit
    Increment row and column
  Admit
    If not 4 then quit
    Decrement column
  Admit
    If not 6 then quit
    Increment column
  Admit
    If not 7 then quit
    Decrement row and column
  Admit
    If not 8 then quit
    Decrement row
  Admit
    If not 9 then quit
    Decrement row
    Increment column
  Endguess
  Return
```

*continued on next page*

Move Z  
 Initialize Z-move counters  
 Loop  
   Set Z-pointer to 0  
   Loop  
     Calculate new Z position  
     Guess  
       If Z has not been killed then quit  
     Admit  
       If Z has not moved then quit  
     Admit  
       Compare new Z position to current Y position  
       If no match then quit  
       Set game-end to true  
     Admit  
       Compare new Z position to current O positions  
       If no match then quit  
       If O depth is zero then quit  
       Eliminate Z  
     Admit  
       Compare new Z position to other Z positions  
       If no match then quit  
     Admit  
       Move Z to new Z position  
       Decrement Z-move counter  
 Endguess

Increment Z-pointer  
 Until Z-pointer = Total-Z or game-end is true  
 Until all Z-move counters are 0 or game-end is true  
 Return  
 Calculate new Z position  
 Set new Z position to current position  
 Calculate Y-row - Z-row  
 Save row difference  
 Calculate Y-col - Z-col  
 Save column difference  
 Calculate absolute values of row and column differences  
 If abs(row diff) > or = abs(column diff)  
   If Y-row > or = Z-row  
     Increment Z-row  
   Else  
     Decrement Z-row  
 Endif  
 Else  
   If Y-col > or = Z-col  
     Increment Z-col  
   Else  
     Decrement Z-col  
 Endif  
 Endif  
 Return

**WANTED**

commodore **CBM** dual drive floppy disk  
 Model 4040

**4040 DISK DRIVES**

commodore **CBM** HARD DISK  
 9090

**HARD DISK**

for TBUG membership office  
 call **Chris Bennett,**  
 Business Manager  
 for further information.  
**(416) 782-9252**

## 6502 Assembly Language

Sheridan College is hosting two, two-day workshops on 6502 Assembly Language Programming for the PET, Apple and Atari computers. Participants will study the conceptual foundations of machine language programming, learn the most useful commands in the 6502 instruction set and write working assembly language subroutines and programs. All computer time and manuals are provided for this intensive two-day course. The only prerequisite is an elementary knowledge of BASIC programming.

The two workshops will be held at the Brampton Campus on Aug. 27 - 28 and again on Sept. 8 - 9, 1984. The fee is \$165.00 for the two days, including lunch. Further information and reservations may be obtained by calling Sheridan College Conference Centre at 845-9430, 823-9730 or 632-7081, ext. 336.

Sheridan College  
 845-9430

# ENHANCE YOUR COMMODORE-64

## RAMDISK-64

The RAMDISK-64 is a cartridge containing 64K bytes of RAM used to emulate a disk drive. No more long waits for program saves and loads. Use Ramdisk-64 as a second disk drive. Or use the 64K bytes as extra memory for large and data intensive software.

### FEATURES:

- 64 kbytes of dynamic RAM
- Includes software to emulate a disk drive
- Loads an 8 kbyte program in less than 1 second.
- Does not use COMMODORE-64 RAM space
- System reset does not erase files in RAMDISK-64.
- Small size only 3"x5.5"x0.5"
- Useable with other cartridges

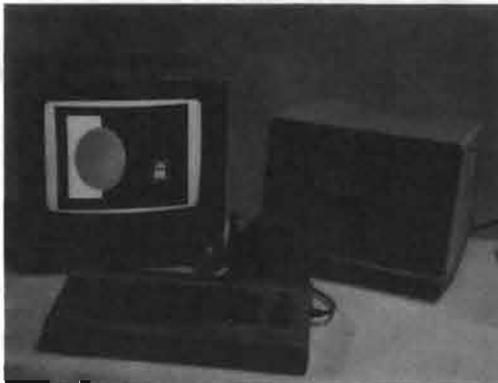
### APPLICATIONS:

- Time saver where disk usage is high:
  - Software development
  - Large programs using overlays
- Software requiring large data storage. Use with data bases or spreadsheets. Save multiple graphic screens.
- Use as a second drive. Minimize diskette swapping.



## MORE ABOUT DISK EMULATION SOFTWARE

- Allows 16 directory entries or 63.5 kbytes of storage.
- Compatible with BASIC commands OPEN, CLOSE, GET#, INPUT#, PRINT#
- Supports PRG and SEQ files
- Treat RAMDISK-64 as device 15, user changeable.
- OTHER FEATURES TOO NUMEROUS TO MENTION HERE.



## VIDEO-80

The VIDEO-80 is a high-quality 80-column cartridge.

### FEATURES:

- 80 columns x 25 lines display
- 256 character set with full Ascii and reverse letters
- Flicker-free crisp display even during scrolling
- Compatible with BASIC and the KERNAL
- Includes customizing video routines like scrolling, insert and delete line, address cursor, and screen blank
- DUAL SCREEN: Connect the normal C-64 video output to a color monitor AND connect the VIDEO-80 video output to a B/W or green screen monitor. Text will appear on the B/W or green screen monitor and the color graphics on the color monitor. A NECESSITY FOR GAME DEVELOPERS AND GRAPHICS PROGRAMMERS.
- Compact size only 3"x5.5"x0.5".

### APPLICATIONS:

- Word Processing
- Spreadsheets
- Terminal Emulation
- Software Programming. Eliminate irritating line wrap around.
- Graphics development. Use VIDEO-80 for text and C-64 video output for color graphics simultaneously.

The RAMDISK-64 and VIDEO-80 use the COMMODORE-64 expansion slot. An optional extender board (\$10) or motherboard is required.

### PRICES

RAMDISK-64 .....\$149.  
VIDEO-80 .....\$99.  
Extender Board .....\$10.

Calif. residents add 6% sales tax. Add 5% shipping and handling. VISA, MasterCard, COD accepted. Personal checks require two weeks to clear. Order by phone or mail.

PRICES QUOTED IN U.S. FUNDS

## P Technologies

1555 Riverpark Drive, Suite 206  
Sacramento, CA 95815  
(916) 920-3226  
8:00 am to 5:00 pm PST, Mon-Fri.

EXCLUSIVE CANADIAN DISTRIBUTORS - B & R ENTERPRISES & ASSOCIATES  
29 IRVING DR. - PEPPERLAW, ONT. - L0E 1N0 - CANADA - (705) 437-3187 EX. 21  
CANADIAN DEALERS INQUIRIES INVITED.

---

---

# THE NOS TRANSLATOR

---

Peter Spencer  
West Hill, ON

Picture this. There are ten different record companies. If you buy a record made by United Artists, you have to play it on a United Artists record player. If you buy a Columbia record, it will only play on a Columbia-made record player. Far-fetched? You bet. You and I take it for granted that all brands of records are playable on all brands of record players.

Now picture this. An APPLE-owning friend saves you one of his home-brew programs on tape. You take the tape home, put it in your Commodore computer cassette deck, and load the program. Sound far-fetched? That's what I used to think, but yesterday I did just that.

Not only did I use an APPLE-written tape to load a program into my Commodore machine, I also watched as an APPLE successfully loaded programs from tapes written by a TRS-80, an OSI, an Exidy Sorcerer, and several other makes of microcomputer. Later, I took the same tapes home and successfully loaded the same programs into my family's various Commodore machines!

You don't have to have been around microcomputers very long to realize that what I just described comes very close to sounding like magic. When I first heard about it, it was with a great deal of scepticism, and then excitement, that I tried it out. I found that not only does it work, it works well, it works easily and it works fast!

What makes it possible is NOS BASICODE 2, a public domain software project of the Netherlands Broadcasting Corporation (NOS for short).

Here's what you do: NOS has prepared a series of translator programs for each of the following makes of machines: Apple II, BBC (Models A and B), PET/CBM, PET 2001, C-64, VIC 20, DAI, Exidy Sorcerer, OSI, Philips P2000, Sharp MZ80 A and K, SWTPC, TRS-80, Video Genie and computers with CP/M. You load the translator program for your type of computer, it

hides itself somewhere in memory and then you can either write cassette tapes in NOS format that are readable by any of the other brands of computer or you can load NOS-format tapes written by any of the other brands. I found that on any of our family's Commodore computers, the reading and writing of NOS tapes was more than twice as fast as reading or writing an equivalent-length program in regular Commodore format.

The NOS translator programs come with a detailed, glossy, 76-page manual that explains their operation, the NOS protocol, the NOS tape format, and the gotchas.

Aha, you say; gotchas! You knew it all along, didn't you? Probably requires two extra circuit boards, 96K of memory and the technical expertise of an electronics engineer, doesn't it? The answer is definitely NO, but there are some simple gotchas. For example; if you thought you were going to get a program that makes extensive use of Commodore graphics to run on the APPLE or TRS-80, you have another thought coming. Hence, the Commodore translators are rigged to give an error message when you try to save

Commodore graphics, something your APPLE and TRS-owning friends will be thankful for. Also, because of the technical limitations of some of the micros in the project, any BASIC line that takes up more than 60 spaces when listed to the screen causes an error message when you try to save it in NOS format. My feeling is that this one is a blessing in disguise, as it forces 'spaghetti coders' to chop up their program code into shorter, more readable lines.

The BASICODE instruction manual gives a list of BASIC keywords, such as INPUT and ABS, that are freely interchangeable among the various machines in the project. It also explains the various subroutines that are automatically tacked on to any program in NOS to let you clear the screen, place the cursor at a given location, check if a key has been depressed, cause the computer to beep, simulate a print-

using statement, print to the printer, and so on. Line numbers from 0 to 999 are reserved for these subroutines, which you never have to type in once you have the translator program.

Now, for the technically minded, what actually goes onto tape? That is, what is NOS format? NOS uses two tones to record programs onto tape, 1200 Hz and 2400 Hz. A binary 0 is represented as one full cycle of 1200 Hz. A binary 1 is represented as two cycles of 2400 Hz. The binary data is sent at 1200 baud. Each byte of information is sent as eleven bits, one 0 start bit, eight data bits with the least significant first, and finally, two logic 1 stop bits. BASIC keywords are not tokenized, and all letters and ciphers are presented in ASCII. Each BASIC instruction is followed by a space. Each BASIC line is closed with a carriage return (hexadecimal 8D). All true ASCII characters in the program receive a logic 1 closing bit.

The sequence that NOS writes to tape is as follows. The leader is five seconds of stop bit (2400 Hz). The ASCII character 'START Text' (Hex 82) follows, then the BASIC information in ASCII, followed by the ASCII character 'End of Text' (hex 83). Then comes a checksum, which is derived by exclusive or'ing each byte sent with the byte previously sent. The checksum is eight bits long. The tape trailer is another five seconds of stop bit (2400 Hz).

How seriously should you take NOS BASICODE 2? Will it be just a flash in the pan? I cannot prophesy, but do you remember that there used to be several competing tape standards in the music industry, and then there were only two, namely eight-track and cassette, and now most shops seem to carry only one? I also remember that the format that now predominates was invented in the Netherlands and licensed on extremely generous terms in order to make it successful. Sounds familiar, doesn't it? And what terms could be more generous than making everything public domain?

*continued overleaf*

Weekly radio programs in the Netherlands use NOS BASICODE 2 to transmit programs of general interest to the computing public, and listeners within range of 2000 km of the transmitters have apparently reported perfect, or near perfect, copy. The Radio Netherlands transcription service carries an English-language program, "Radio-Activity", freely available to any radio station in North America, which uses NOS BASICODE 2 to transmit computer programs. The people responsible for NOS also welcome submissions of programs in NOS format, and there is a monthly prize for the best program sent in.

NOS translator programs have been or are being developed for other brands of computers than those I listed above. For instance, a loading-only program is available for the Sinclair ZX-81/Timex microcomputer, and incomplete translators already exist for the

Acorn Atom, Cosmicos 1802, NASCOM/Gemini, and TI-99/4A makes. The Sinclair SPECTRUM is also reportedly being investigated.

At the end of this article is a Commodore NOS-translator program, which I have translated from Dutch to English. You can get the NOS BASICODE 2 manual, which is professionally typeset and printed, and the NOS cassette containing the translator programs for the various machines and twenty-one sample programs in NOS format, by sending a bank money order for thirty Dutch guilders (roughly fifteen U.S. dollars) to BASICODE, Administratie Algemeen Secretariaat, NOS, P.O. Box 10, 1200 JB Hilversum, Netherlands. The bank money order should be made payable to "NOS Algemeen Secretariaat." Both the cassette and manual are bilingual, with everything presented in Dutch and in English.

Since NOS BASICODE is a non-profit public domain project being sold at cost, please don't write to NOS asking questions without including one or two International Reply Coupons, available at most post offices. Be sure to include your name and address clearly printed out. Most Dutch read English by the age of twelve, but they can't read minds!

Have fun with BASICODE. I certainly have. TPUG

*(Editor's note: Mr. Spencer provided programs for most Commodore computers. Because of lack of space, we are publishing only the C-64 version here. All the programs have been given to the TPUG librarians for inclusion in the club library.)*

```

10 goto1000
20 goto1010
100 print"3";:return
110 printleft$("50000000000000000000000000000000",abs(int(ve+1)));
112 printspc(abs(ho));:return
120 ho=peek(209)+256*peek(210)-1024+peek(211)
121 ve=int(ho/40):ho=ho-40*ve:return
200 getin$:return
210 getin$:if in$="" goto210
220 return
250 poke54296,15:poke54278,255:poke54273,95:poke54276,17
251 foroo=1to200:nextoo
252 poke54276,0:poke54277,0:poke54278,0:return
260 rv=rnd(1):return
270 fr=fre(0):return
300 sr$=str$(sr):if left$(sr$,1)=" " then sr$=mid$(sr$,2)
301 return
310 sr$="":if sr>=10*(ct-cn-sgn(cn)) then 318
311 if sr<-.5*10*(cn) then sr$="-"
312 sr$=sr$+mid$(str$(int(abs(sr)+.5*10*(cn)),2)
313 if len(sr$)<ct-cn+(cn<>0) then sr$=" "+sr$:goto313
314 if cn=0 or abs(sr)>=1e10 goto316
315 sr$=sr$+mid$(str$(1+abs(sr)-int(abs(sr))+.5*10*(cn)),3,cn+1)
316 if len(sr$)<ct then sr$=sr$+" ":goto316
317 if len(sr$)>ct then sr$=""
318 if len(sr$)<ct then sr$=sr$+"*":goto318
319 return
350 open4,4:print#4,sr$:c lose4:return
360 open4,4:print#4:c lose4:return

```

*continued on next page*

```

1000 rem"    BASICCODE TRANSLATOR
1010 rem"    for    COMMODORE-64
1020 rem
1030 rem    this program will
1040 rem    not work on any other
1050 rem    type of computer.
1060 rem
1070 rem    this version 1983 02 16
1080 gosub100 :gosub2100
1100 printchr$(14)rv$;
1110 print"  N O S - H O B B Y S C O P E - B A S I C O D E  ";
1120 printrv$"
1130 ifpeek(61731)<>85thenlist1000-1070
1140 fori=1to5:print:next
1150 print" You can choose from:
1160 print:print:print"    "rv$"1"no$;
1170 print" The LOAD programs setup;
1180 print:print"    "rv$"2"no$;
1190 print" The SAVE programs setup;
1200 print:print"    "rv$"3"no$" Stop.
1210 print:print:input" Your choice";a
1220 on a goto 3450,3220,2500
1230 run
1240 :
2000 ve=24:ho=1:gosub110
2010 printrv$"  To continue: press space bar    ";
2020 gosub210
2030 return
2040 :
2100 rv$=chr$(18) :rem reverse on
2110 no$=chr$(146) :rem reverse off
2120 ba$=rv$+"BASICCODE"+no$
2130 printchr$(14);:return
2140 :
2500 gosub100:print:print" More information on "ba$" can
2501 print" be had from N.O.S.
2502 print" The address is:
2503 print
2504 print"    "rv$"  N.O.S. - Hobbyscope    "
2505 print"    "rv$"  P.O. Box 10            "
2506 print"    "rv$"  1200 BJ HILVERSUM, Netherlands"
2507 print
2508 print" Make your bank money order payable to:
2509 print" NOS ALGEMEEN SECRETARIAAT
2510 print"
2511 print
2512 print" For only 30 dutch guilders you get
2513 print" the BASICCODE manual and the accompanying
2514 print" cassette, via airmail to North America.
2515 print:end
2516 :

```

*continued overleaf*

```

3000 rem data conversion
3010 :
3020 ms=peek(56):mt=ms*256:ad=mt
3030 c1=1088:c2=2:dt=32:print:print:print
3040 readda$:ifda$="*"thenreturn
3050 ve=23:ho=10:gosub110:sr=dt:gosub300
3060 printsr$" one moment please... ";
3070 dt=dt-1:le=len(da$):forka=1tolestepc2
3080 by=asc(mid$(da$,ka,1))*16-c1
3090 by=by+asc(mid$(da$,ka+1,1))
3100 ifby<0thenby=by+ms+256
3110 pokead,by:ad=ad+1:next:goto3040
3120 :
3200 rem save
3210 :
3220 poke56,peek(56)-2:poke55,0:run3230
3230 gosub2100
3240 reada$:ifa$<>"*"goto3240
3250 gosub3020
3260 gosub100
3270 print" SAVE in BASICODE as follows : "
3290 print:print
3300 print" First load the program to be saved;"
3310 print:print:
3320 print" Then type : SYS"mt
3330 print:print" to save from line 1000 on;
3340 print:print
3345 print" NB: during the save in BASICODE the ";
3346 print" screen is unfortunately blank. "
3350 print
3360 print" ( Special case : SYS"mt+6
3370 print:print" will save ALL LINES.)
3380 print:print:print
3390 print" "ru$"WRITE DOWN THESE SYS COMMANDS! "
3400 gosub2000:goto1000
3410 :
3430 rem load
3440 :
3450 poke56,peek(56)-2:poke55,0:run3460
3460 gosub2100:gosub3020
3470 gosub100:print
3480 print" With the instruction:
3490 print:print" SYS"mt
3491 print:print" all existing lines from line 1000 on"
3492 print:print" will be deleted and replaced by BASICODE"
3493 print:print" code loaded from the tape."
3500 print:print:print
3510 print" Merge the current program with BASICODE
3511 print" (but keep your line numbers in order)
3520 print" from tape, with : "
3530 print
3540 print" SYS"mt+35
3541 print:print:print" NB: While loading from BASICODE the ";
3542 print" screen unfortunately is forced to go blank. "
3550 gosub2000:goto1000
3560 :

```

continued on next page

```

5000 rem data basicode load
5010 :
5020 data jbecjinhheadhfaseb@acjfji@ejhiaeo
5030 data lhiaseahji@bfeoehebmjefefi@eghebn
5040 data b@eijfb@agohj@ebiasche@anf@ejimc
5050 data j@eb@an@kjji@eh@ejdhejfhia@b@h@e
5060 data okghj@eadhjdichdikjibkm ljjm@mm lbi@e
5070 data c@ehhjbi@em@nknfjcm@nm lfik@eniji
5080 data moj@eb@an@kjmad@ecjnae@chmi@ebhn
5090 data j@ebjigdj@b@ahmad@echnae@cjii@em@em
5100 data mljikkhm@emlji@echm@emlji@ekhma@em
5110 data ehb@eko@e lhh@m@oihei lje@ehd@i@o@b@e
5120 data @eb@ig@j@n@em@e@f@n@e@cm@e@cb@b@e@k@o@b@e
5130 data fn@ab@eijfjeil@e@cdli lmad lgdj@e
5140 data @abi@em@njfjfkmc l@e@o@e@cdh@e@i l@e
5150 data ilji@em@mc l@ech@h@e@ajjh@fjfh@em@bb
5160 data m@ef@jei@o@i@o@e@i@e@i@o@e@aa@mah@m@bi
5170 data @b@e@j@h@b@e@ack@ai@e@de@ih@e@c@egh@jli@e
5180 data i@e@bb@ie@oi@e@eb lh l@e@em@ibji@e@ii@e
5190 data @bj@oo@j@e@ah@f@gh@y@k@b@b@e@y@e@k@e@cm@b@e@fk
5200 data jib@y@j@eh@l@om@ajji@e@io l@e@ai@om@e@ah
5210 data l@hdin@je@ah@m@n@e@aj@e@e@h@m@o@e@aj@e@b@mc@h
5220 data ni@b@h@e@b@k@e@b l@b@nk@io l@e@ai@e@m@h@e@oh
5230 data jekmah@e@in@h@e@bmi@e@b@n@f@b@n@f@b@n@h@nd@ch
5240 data k@e@cdlij@e@bb@f@n@ab@e@ijfd l@e@j@b@e@ic
5250 data o ldlcc@je@n@em ljm@em lje@aih l@em l@f
5260 data j@e@dj@h@f@k@f@j@k@f@f@k@f@j@j@ah@ji@e@n@e
5270 data dbfi@e@n@egbfi@e@j@j@io@e@aj@j@dm@e@dh@f
5280 data j@e@b@b@n@f@cm@e@b@ji@e@cm@e@j@dj@h@e@j@e@j@f@f
5290 data iki@e@aj@e@i@k@j@dj@ic l@e@h@e@e@b@j@e@e@hd
5300 data j@dj@b@b@h@f@j l@f@j@i@h@e@i@k@h@e@j@e@j@l@em@l
5310 data d@h@e@j@e@ded@e@e@b@e@b@e@cd@h@di@dnd@y@e@m@e@e@df
5320 data d@e@e@nd@dd@e@m@e
5330 data "*"
5340 data "*"
5350 :
50000 rem translator: p. spencer
50010 rem toronto
50020 rem author: j. haubrich
50030 rem eindhoven

```

```

5000 rem data basicode save
5010 :
5020 data jinhj@e@cm@e@cjii@ejjheadhfaseb@acjf
5030 data b@ech@oh@j@e@e@bi@e@che@e@anf@l@ejim@j@e@e@b@e
5040 data an@kjji@e@j@e@h@i@e@j@l@m@ok@j@mad@ec@j@nae
5050 data e@chmi@e@b@h@j@e@b@ji@e@k@h@ma@em@gh@j@im@ch@m
5060 data @dm lje@o@ch@dkn lhh l@em l@e@dh@e@aj@e@k@e@dm@e
5070 data o l@hdh@k@h@di@n@hd@io@j@h@b@e@b@h@e@aj@i@m@b@e@b@h
5080 data @ab@k@e@e@e@e@ag@e@e@j@h@e@b@h@e@aj@e@h@k@b@e@b@h
5090 data @e@m@e@n@j@od l@m@j@im@l@m@j@n@b@h@b@i@e@l@h
5100 data j@e@b@e@an@j@dl@e@j@e@e@k@e@e@o@j@l@h@e@e@o
5110 data h@e@f@h@f@e@om@e l@f@j@e@e@h@dkm l@h@dk@e@k@e@e@o@e
5120 data l@l@h@k@e@e@h@f@e@h@e@i l@h@di@m@k@e@e@h@e@f@b@h@e
5130 data c@j@b@e@i@i@k@b@e@m@k@j@e@e@o@e@e@b@e@f@e@al@h@ki
5140 data @e@e@m@e@o@j@b@b@e@h@e@i@e@j@o@n@e@b@b@m@e@f@j@e@k@m
5150 data @i@o@e@k@e@m@b@k@e@ak@j@mah@m@bi@e@b@o@e@dh@j
5160 data c@e@i@b@e@ack@ai@e@e@ldi@b@e@m@e@k@b@i@g@o@b@e@ack@e
5170 data i@eh@y@j@h@e@h@e@b@e@f@e@an@f@im@j@im@e@h@i@ka
5180 data e@o@j@o@e@h@e@e@l@e@b@e@k@m@e@k l@ch@n@i@g@o@j@j@e@o
5190 data l@j@o@e@h l@h@i@n@j@e@e@o@j@e@e@e l@h@i@n@j@e@e
5200 data m@e@b@e@f@e@m@e@o@e@i@h@e@j@f@i@n@dh@j@e@e@bi@e@e@e
5210 data @e@dl@y@e@e@k@e@j@e@e@e@f@h@i@e@j@e@de@h@k@e@h@k@h
5220 data h@j@b@i@e@h@e@i@n@f@e@f@e@j@i@ah@h@e@k@e@j@e@o
5230 data h@e@j@k@h@e@i@h@e@i@k@j@i@f@j@b@e@h@mad@e@ch@nae@e@e
5240 data e@h@f@e@j@m@em l@e@i@k@e@e@f@j@e@adi@e@h@h@e@e@e@f
5250 data ik@e@fi l@e@en@j@e@k@e@e@l@i@e@o@e@ai l@f@j@k@e
5260 data b@e@e@e@dj@i@e@dh@e@j@k@j@i@e@b@di l@e@o l@h@e@i@k@j@i
5270 data @h@h@e@i l@l@h@an@j@df@k@f@j@i@e@i@e@b@j@i@e@l@f@k@y
5280 data m@ng l@f@j@m@e@nc@j@f@i@o@k@e@j@e@e@e@e@e@e@e@e@e@e
5290 data n@h@e@k@f@j@i@e@i@e@j@i@e@h@h@e@k@g@h@h@j@b@i@e@g@h@e
5300 data io@j@i@e@m@e l@l@f@k@e@m@e@k@n l@f@k@e@e@k@j@n@f@k@n@e
5310 data k@b@e@i@e@o l@l@gd@j@e@o@l@e@o@d@e@l@e@o@d@e@l@e@i@h
5320 data d@e@e@e@o@di@l@l@b@n@e@cd@h@e@b@n@e@e@e@e@e@e@e@e@e@e
5330 data e@f@di@dnd@y@e@m@e
5340 data "*"
5350 :

```



**ABRAXAS**  
**SOFTWARE**  
**AUSTRALIA**

## The Video Management System

FOR THE CBM 8000 SERIES COMPUTERS

INTRODUCES:



SYSTEM INCLUDES

1. SET RATES
2. NEW MEMBER
3. NEW FILM
4. MODIFY/DELETE MEMBER
5. MODIFY/DELETE FILM
6. MEMBER ENQUIRY
7. FILM ENQUIRY
8. TAPE ENQUIRY
9. SALE/HIRE/RETURN/RENEWAL
10. OTHER TRANSACTIONS
11. ADD NEW TAPES
12. AGEING
13. SORT MEMBERS
14. SORT FILMS

## The Video Management System Menu



ONLY \$550.00 U.S.  
\$685.00 CDN.

The **FILM** file has for each film (title)

1. Film number
2. Film title
3. Category\*
4. Number of times hired this month
5. Number of times hired in total

The **TRANSACTION** file has for each entry

1. Membership number
2. Transaction date debit/credit (for hire with cash paid) account payment, invoice or credit note
3. Transaction reference number

5. Description  
6. Amount

The **MEMBERSHIP** file has for each member

1. Membership number
2. Surname, initials, address, postcode, telephone number
3. Membership expiry date
4. Membership class\*
5. Category\*
6. Type\*
7. Number of films hired by member
8. Opening balance current balance

The **TAPE** file has for each tape (copy)

1. Tape number
2. Film number
3. Status (in stock or out on hire)
4. Date last returned, if in stock or date due back if on hire.
5. Supplier from which tape was acquired
6. Type of tape\*
7. Cost of tape
8. Last two members who hired tape
9. Member who has booked tape
10. Number of times tape has been used

EXCLUSIVE NORTH AMERICAN DISTRIBUTORS



**ROMARO ENTERPRISES**  
**INTERNATIONAL**



**BEACON SOFTWARE**  
**INC.**

705 Progress Avenue, Unit 17  
Scarborough, Ontario  
M1H 2X1

(416) 431-3200

**CANADA**

P.O. BOX 227  
STRETSVILLE POSTAL STN.  
MISSISSAUGA, ONTARIO  
L5M 2B8

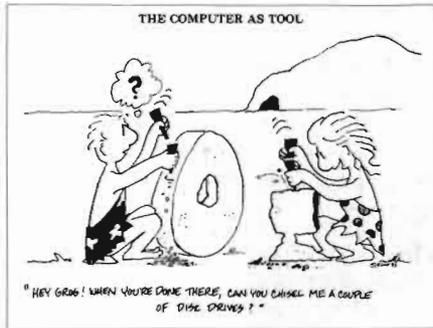
(416) 820-5235

**UNITED STATES**

DEALER INQUIRIES INVITED

**FULL FILE**  
**SECURITY**

# ICCE Publications



John David  
Islington, ON

The International Council for Computers in Education is a non-profit, tax-exempt professional organization. ICCE is dedicated to improving educational uses of computers and to helping both students and teachers become more computer literate. Although ICCE is headquartered in Eugene, Oregon, a glance at the membership reveals a truly international association — Australia, Canada and Ireland are members — that ensures a broad basis of experience from which to draw. Two of its recent publications, aimed at students and teachers, are *Computer Metaphors: Approaches to Computer Literacy for Educators* and *Computers in Composition Instruction*.

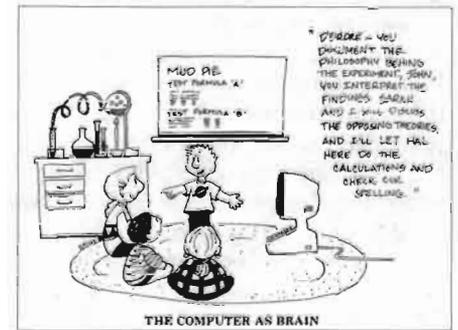
In his introduction to *Computer Metaphors: Approaches to Computer Literacy for Educators*, Howard Peelle argues that the metaphor — a comparison that is implied rather than directly expressed (e.g. "He is a fox." The comparison implies that the individual is sly or cunning without using those words) — is very useful particularly in fields such as computers in education for comprehending something new. The metaphor is a stepping-stone from the known to an area of exploration. In a series of short essays, Peelle explores the computer as person, tool, brain, map, glass box, vitamin, palette, mentor and catalyst. Each metaphor explores a different capacity of the computer. As a tool, the computer has become powerful, versatile and ubiquitous — used in industry, govern-

ment and education. As a map, the computer can assist learning (computer assisted instruction — CAI) by guiding a student through a series of problems related to one topic. As a palette, the computer is used in the artistic process for creating colourful screens or for learning and teaching music. Each of these articles is accompanied by a cartoon related to its central thesis. In "Computer as Mentor" students face a teacher on the left and a computer on the right, each seated at a desk. A student leans over to his friend and says, "The one on the right may have a lot of answers, but the one on the left a is a lot more understanding." These illustrations and statements such as, "Computer-brain comparisons get a bit gray on the matter..." make the booklet a joy to read, yet make clear the point of each metaphor. For any introductory course in computer studies, *Computer Metaphors: Approaches to Computer Literacy for Educators* would be a worthwhile addition to the course library to help focus attention on the various facets of a new educational diamond — the computer.



*Computers in Composition Instruction*, another ICCE publication, is a collection of essays written by those with experience in using computers to assist in the difficult process of writing and directed to teachers of writing. It is

divided into three major areas: the writing process, courseware design, and courseware selection. At the conclusion of each essay is a brief catalogue of programs to aid the writing process.



Included with each catalogue entry are the authors, the target grades, hardware description and cost. (No Commodore software is listed — a programming challenge? And opportunity?) These are not simply lists of word processors but programs to assist all phases of the writing process — prewriting programs to generate ideas or refine student thinking (somewhat like twenty questions), programs that collaborate with the student in developing a thesis and programs that aid in editing and revision of topics. In the latter case, Earl Woodruff, Marlene Scardamalia and Carl Bereiter — Ontario educators — examine a series of programs that assist student writing. They mention both the weaknesses and the strengths of the program called EXPLORE. Such criticisms allow potential users of the program to anticipate and compensate for perceived weakness in the courseware.

Overall, the ICCE series of booklets (there is a bibliography on the back of each; the list of topics is extensive) which offer ideas not only to instructors in computer studies, but also to teachers and students in other disciplines as well, seem to be a worthwhile addition to any school's curriculum. TPUG

# Administrator's Guide To Computers In The Classroom

David Williams  
Toronto, Ont.

Imagine you are a school administrator. Your desk is loaded with papers concerning curricula, building maintenance, proposed dates for next year's P.D. days, and evaluation reports on a thousand teachers. The police have just called to ask your advice concerning a dozen teenagers who recently robbed a drugstore. Last night, you went to a meeting which was addressed by a politician who was full of proposals to cut your budget even further. The letter you are reading is from the president of the Concerned Parents' Association asking what you are doing to improve the quality of Computer Aided Instruction—a topic which causes you twinges of apprehension because you are aware that your knowledge of it is scanty. Suddenly you notice that in your 'IN' tray is a copy of a book you ordered from the University of Oregon which covers this very topic. Realizing that it contains only about fifty pages of text, you give a sigh of relief and decide to skip lunch for the third time this week to make time to read it.

Would the book help you? Well, maybe. You would soon discover that it contains an extensive bibliography of other books and articles you could read (just what you always wanted!). You would also soon find that most of the rest of the book consists of an abstract of these other works. It reads somewhat like a university thesis, in which every significant point is supported with references to other authors. And, in good academic style, it starts with generalities and works its way towards specifics. The titles of the four chapters illustrate this. In order, they are 'The Computer Revolution in Education', 'Using Computers in the Classroom', 'Guidelines for Bringing Computers into the Schools' and, finally, 'Case Studies'. The facts tend to be left until the end.

I do not want to give the impression that I think this is a bad book. On the contrary, as I read it through I found

myself frequently nodding my head in agreement. It contains useful summaries of ways in which computers are being used in American schools, both for teaching and for administration, and it has a lot of information which should be valuable to administrators—if they can find it.

My main criticism of the book concerns not its contents, but its presentation. This is, after all, supposed to be a book which teaches something, and educators are supposed to know effective ways of presenting material so that it can be easily learned. Every would-be teacher learns to argue FROM specific examples TO abstract principles. Effective textbooks do the same thing. This book does not. Within their own community, it seems, educators do not always practice what they preach! Most

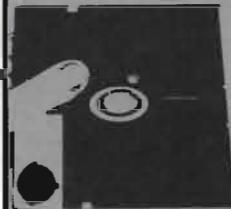
teachers are also well accustomed to using visual aids wherever possible. A well-presented illustration or graph can convey more information than thousands of words. This book contains no illustrations, no graphs, nothing but text.

This isn't a bad book. Many school administrators would probably be well advised to buy it. But it could have been so much better! TPUG

## Administrator's Guide to Computers in the Classroom

Published by:  
Clearinghouse on Educational Management  
College of Education  
University of Oregon  
Price: \$5.50 (U.S.)  
ix + 53 pages

## REDUCES DISKETTE COST 50%!



Make use of the back of your 5 1/4" Diskettes and SAVE

- **NIBBLE NOTCH** tools make it easy.
- Adds the precise notch where you need it.
- Doubles diskette space or MONEY BACK!

### NIBBLE NOTCH I

Cuts square notch for users of  
Apple, II, II+, IIe, III, Franklin and Commodore.

only **\$14.95** each  
add \$1.50 each Postage & Handling (\$4.50 each foreign P&H)

### NIBBLE NOTCH II

Cuts square notch and 1/4 inch round "index hole." For use  
with computers other than those shown above.

only **\$21.90** each  
add \$2.50 each Postage & Handling (\$6.50 each foreign P&H)

\*Florida Residents Add 5% Sales Tax\*

SATISFACTION GUARANTEED OR YOUR MONEY BACK!

## ORDER TODAY!

SEND CHECK OR MONEY ORDER TO:

### NIBBLE NOTCH COMPUTER PRODUCTS

DIVISION OF CORTRAN INTERNATIONAL

4211 NW 75th TERRACE • DEPT. 15 • LAUDERHILL, FL. 33319

PAT. PEND.

ALL TRADEMARKS ARE ACKNOWLEDGED

## hey diddle diddle

Rich Westerman  
St. Anne, ILL

*HEY DIDDLE DIDDLE* from SPINNAKER. Disk (protected) for C-64. Recommended for ages 3-10. \$39.95 (U. S.) (much cheaper at discount outlets).

**Hey Diddle Diddle** is a package which introduces children to poetry and verse by way of hi-res graphics and music.

Upon loading the program you are presented with a three-option menu inviting you to choose either **Storytime**, **Storybook** or **Rhyme-Time**. It being my practice to never read the directions first, I chose the option which caught my fancy, **Rhyme-Time**. I was also offered a choice of two levels of difficulty.

Okay. How about level 1?

The program paused for a moment while the C-64 loaded some files from the disk and suddenly I was confronted with a four-line poem on the screen. The four lines of the poem, however, were arranged incorrectly. Hmmm... Using the joystick, I managed to rearrange the lines of the poem to what I thought was the correct order. I then invited the computer to 'check' my work. (I must confess that my 8-year-old was looking over my shoulder and he,

being a poetry fan, checked my work first). Guess what? I got it right! (Lots of flashing lights, congratulations, etc.) My poetic scholarship was then rewarded with the poem being displayed at the bottom of the screen while a hi-res picture of events described in the poem was drawn above the script. While the illustration was being created, 3-voice music poured out of my T.V.'s speaker. Emboldened by my success in Level 1, I bravely attempted to play a round in Level 2.

This time, eight discrete lines of poetry were displayed on the screen for me to arrange in a sensible order. This did indeed increase the difficulty of the problem, but after several rounds at Level 1, I believe that children in the upper recommended age ranges will welcome this challenge.

**Rhyme-Time** has a 'clock' factor in it to add still more complication to the solving of the rhyme, but the clock can easily be disabled, allowing the child as much time as he or she wants to work on it.

Altogether, a lot of fun, according to the assorted kids who have run the program in our household.

I eventually ran the other options on

the menu, **Storytime** and **Storybook**. **Storytime** is a routine which brings up each of the thirty poems in the 'library'; drawing the picture, playing the music and moving onto the next poem. Sort of a 'demo' mode.

**Storybook** seems to be a routine aimed at beginning readers—the first four lines of the poem are written slowly while the accompanying illustration is drawn. The child has a longer time to read the poem. As in **Storytime**, this is repeated until all thirty poems have been displayed.

The poems chosen for **Hey Diddle Diddle** are all old favorites, taken from such sources as 'Mother Goose' and 'A Child's Garden Of Verse'. I'm heartened at Spinnaker's decision to use 'classic' poems rather than contemporary ones. Kids see little enough real literature in daily life and it certainly can't hurt them to see and speak their own language correctly. I'm curmudgeon enough to be thoroughly disgusted by a generation of kids whose language model is T.V.

All in all, **Hey Diddle Diddle** is one of the most likeable software packages I've seen in *any* category. TPUG

## pitstop

Rich Westerman  
St. Anne, ILL

*PITSTOP* from Epyx. Cartridge. For C-64. \$34.95 (U. S.)

**Pitstop** is a Grand Prix racing simulation/arcade game with a twist.

There are plenty of racing simulations out by now for both the VIC 20 and the C-64. Some good, some not. All involve steering your car over a course (some fancy, some simple) while avoiding various hazards and pitfalls.

**Pitstop** is a bit different. First of all, it is the most 'polished' of the racing sim-

ulations I have seen. By 'polished', I mean that it has taken the best of the graphics routines and combined them with the most imaginative and innovative sound effects. This has resulted in a well-thought-out, very impressive game.

The game powers up with the usual title card and soon prompts you to input the *number of players*—1 to 4. You are then asked which of 3 *levels* of difficulty you wish to play at. Next, the program wants to know which of 6 *different courses* you wish to race on. (The only course names that I recognized

were 'LeMans' and 'Monaco', but I confess to be largely ignorant and totally uninterested in motor-racing). Finally, you are asked *how many laps* you wish the race to consist of.

Having made all your choices, the racing begins. Your car appears on the starting line and you use the joystick in the normal way to steer, accelerate or brake. A separate graphic on the screen gives an 'overview' of the course. You can see a small dot moving about this little course as you're driving— that

*continued overleaf*

# nes...games...games...games...games...games...ga

little dot is *you*. This helps you to plan your driving strategy as well as letting you know how far you have to go to get to the pits.

And here we get to that 'twist' I mentioned before. While racing around the course, you will overtake other cars on the course. These cars have a strange way of preventing you from passing them — they bump into you! Every bump takes its toll on your tires. Eventually your tires will go through a series of colour changes, indicating that they are worn out and must be replaced. (Hitting the side walls of the course will cause tire wear, as well). At a certain point on the course you may enter the pits for tire replacement and

refuelling.

When you enter the pits, the screen display becomes that of a race car in the pits, surrounded by members of the pit crew. Using the joystick, you must manipulate each crew member in changing tires and refuelling your car. While you're doing this, you see the race continuing in the background (a nice touch of realism). And speaking of realism, the sound effects on this game are quite good. You even hear your car 'go through the gears' as you accelerate and decelerate around the course.

After completing all the work necessary in the pits you appear back out on

the track and continue on with the race until the designated number of laps have been completed or you crash, due to ignoring tire wear. If more than one person is racing, the next racer will now be prompted to 'get ready' and he or she will try to beat the other racer's time. A scoreboard appears at the end of the game giving the order of the finishers, their times and the money won.

I can really find little fault with this game, except that I seemed to tire of it quickly, as did my kids. Even though it is, in my opinion, the best of the racing simulations, it still doesn't have the 'staying power' of classics like *Jumpman*, or *Miner 2049'r*. *TPUG*

## lode runner

Ian Wright  
Toronto, Ont.

*"You are a galactic commando deep in enemy territory ... Your task? To infiltrate each of 150 different treasury rooms, evade the deadly Bungeling guards, and recover each chest of Bungeling booty."*

This is from the opening description of a terrific new game from Broderbund Software who brought you such favourites as *David's Midnight Magic*, and *Choplifter*. The program "Lode Runner" is another award winner. You don't shoot everything that moves, and although sharp reflexes are an asset, you also need to plan ahead to complete more than a few screens.

You do not use your laser pistol to shoot the guards, but to drill pits and passageways into the floor. If a guard falls in, you may run over him, but he will either climb out or be replaced by a new guard. If you fall in you lose one of your five lives — but you can collect extras by completing each level.

The graphics are marvelous — especially the moving "humans". The sounds and music are nice and not at all overpowering; unlike so many games.

This program comes with a booklet of instructions which explains the various features of the game. Although

best with a joystick, the game can be played from the keyboard, and there are an additional twelve keyboard commands that allow you to freeze, toggle on/off, display and even cheat!

So far there is little to suggest that this is any different from "Jumpman" or "Frogger", but wait — what's an EDIT MODE? You can design and construct your own rooms! If the 150 rooms on the Broderbund disk are not enough, then construct your own disk of 150

more screens. With this game, "gaming" can move into another skill level in which you can compete to design the toughest, most challenging screen — and then go for high score on each others screens.

Lode Runner was designed by Doug Smith, and the C-64 disk version was prepared by Dane Bigham. These gentlemen should be complimented on their talent and skills. *TPUG*



# de...product parade...product parade...product p



We are introducing a new feature in *TPUG Magazine* — *New Products Announcements*. As a source of information we use news releases sent to TPUG Magazine by various soft and hardware manufacturers and distributors. We hope that our readers will find this new feature useful. Please, write to us if a particular product arouses your interest and you would like to see a more thorough review of it in subsequent issues of our magazine.



**OMNIWRITER/OMNISPELLER** introduced by Micron Distributing

**OMNIWRITER/OMNISPELLER** is a software program that offers a full line of word processing features as an integrated spelling checker for the Commodore 64.

Designed with advanced colour graphics, **OMNIWRITER/OMNISPELLER** is a page-based word processing program, providing instant and clearly defined tabs, indentation and centering. Other features include: multi-line headings and footings; mail merge for names, addresses and labels; underlining and bold-facing; full scrolling capabilities and

function keys to page forward and backward; global search and replace; and it supports a wide variety of printers.

**OMNIWRITER** is fully integrated with **OMNISPELLER**, a 30,000 word spelling checker, which highlights misspelled words and can be expanded to include the "jargon" needs of any professional.

Each program includes a quick reference card, a keyboard overlay and a complete reference manual.

**OMNIWRITER/OMNISPELLER** is distributed through Micron Distributing or is available at your local computer store at a price of \$85.95



**SADI Communications Interface and Printer Adapter** from Connecticut microComputer Inc.

The CmC **SADI** is a microprocessor based interface designed to allow communication between Commodore PET and CBM computers and a wide range of devices including serial and parallel printers, CRTs, modems, acoustic couplers, hardcopy terminals and other computers. **SADI**'s two independent ports (one serial in/out and one parallel out) give the Commodore computers tremendous flexibility as controllers and as dumb or smart terminals. Data can travel between the computer and one or both ports or between ports.

General features include true ASCII conversion, cursor move conversions for program listings, and automatic insertion or deletion of linefeeds. The **SADI** can also issue a form feed or

any number of blank lines. The device address is switch selectable (0-15). Serial features include 11 baud rates (75 to 9600), selectable parity and a 32 character input buffer with x-on/x-off feature. For the parallel device the 'busy', 'ready' and 'data' polarities are selectable.

The **SADI** is easily programmed using BASIC commands, and is compatible with Wordpro, VISICALC and other software. It comes assembled and tested with case, PET IEEE cable and power supply. Thirty-day money back trial period.

Retail price in U.S.A. \$295.00, optional 230 V power supply \$30.00. Contact Connecticut microComputer, 36 Del Mar Drive, Brookfield CT 06804



**Pascal Compiler** from Lymbic Systems, Inc.

Lymbic Systems, Inc., a California-based firm specializing in computer peripherals and software for the Commodore Computer line, introduces the **PASCAL COMPILER** for the Commodore 64. The **PASCAL COMPILER** was developed by Oxford Computer Systems (Software) Ltd.

Pascal, a programming language, simplifies development of software for customized application. It is also the preferred language for teaching programming skills. The **PASCAL COMPILER** therefore should prove useful to both computer students and those who find that off-the-shelf software does not meet their needs or is not easily adaptable.

The **PASCAL COMPILER** generates efficient native code for the C-64, resulting in fast executing programs demanded by the business/scientific environment. The following utilities are provided as an aid to programming development: a linker (allowing for combining assembler code into the program), debugging facilities, and an editor.

The documentation provides a tutorial on Pascal program development through the compilation process, as well as offering the user an easy access reference. Retail price is approximately \$50 U.S. Inquiries should be directed to Lymbic Systems, 560 San Antonio Rd., Suite 202, Palo Alto, California 94306; (415)424-0168.

# arade...product parade...product parade...produc

ESP>Calc from New Leaf Inc.

ESP>Calc was designed for those people who are not experienced users of spreadsheets. New Leaf Inc. has simplified the entry data and arithmetic operations which makes ESP>Calc easy-to-use. However, it also has functions necessary to do complicated financial models. ESP>Calc will run on both VIC 20 and Commodore 64.

ESP>Calc's maximum spreadsheet size is only limited by the computer's memory. This means that on a VIC 20, you

can get approximately 1000 cells and on a Commodore 64, about 2000 cells.

The two-colour manual is a simple, step-by-step guide. It includes some practical examples of spreadsheets for a household budget, heating and electric usage, stock portfolio, rental property analysis and more.

Price: \$43.50 U.S. for a cassette version and \$47.50 U.S. for a disk version. Contact New Leaf Inc. 120 Lynnhaven, Belleville, IL 62223



Prompter from Transierra Technology

The Prompter is a die-cut overlay for the Commodore 64 and VIC 20 computers. With more and more softwares available that utilize the 4 functions keys on these computers, it may become difficult to remember what commands or information has been assigned to these keys. The Prompter solves the problem. The overlay is cut from a special plastic that is flexible enough to pass through a typewriter without damage. It stores with disk or in the cassette case.

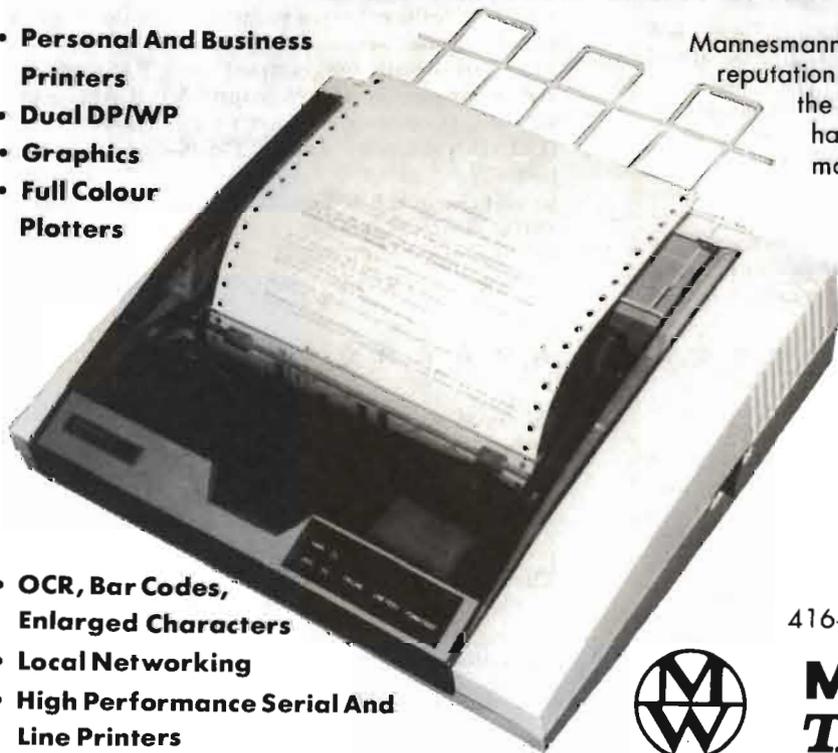
Together with the overlays come preprinted labels that are shaded to indicate the shifted function keys as well as unshifted ones.

The whole package includes 10 overlays, 20 preprinted labels, 20 title labels and instructions. Price \$5.95 U.S. plus \$.25 for postage/handling (and if a California resident \$.36 tax). Send cheque or m.o. to: Transierra Technology P.O. Box 1028, Madera, CA 93639

## GREAT IMPRESSIONS

- Personal And Business Printers
- Dual DP/WP
- Graphics
- Full Colour Plotters

- OCR, Bar Codes, Enlarged Characters
- Local Networking
- High Performance Serial And Line Printers



Mannesmann Tally Corp., has an international reputation for designing and manufacturing the highest quality printers. Reliable, hard-wearing printers designed for maximum efficiency and versatility.

From the Spirit, micro printer at \$499.00 to the MT 660, line printer at \$12,000,

Mannesmann Tally offers the most extensive range of models, all priced aggressively.

Mannesmann Tally makes a great impression.

Available at computer dealers in your area.

Mannesmann Tally Corporation  
703 Petrolia Rd. Downsview,  
Ontario M3J 2N6

416-661-9783 - Telex: 06 522 873



**MANNESMANN  
TALLY**

# COMSPEC

866 Wilson Ave., Downsview  
633-5605

- HARDWARE
- SOFTWARE
- SERVICE
- BOOKS
- ACCESSORIES

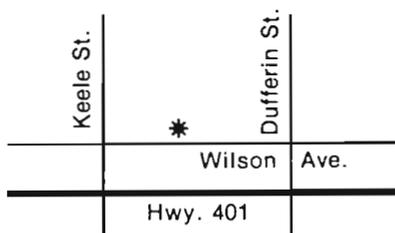
 **commodore**  
Off the Shelf Delivery  
Trade-Ins Accepted

C-64 COMPUTER	348.88
WICO 'BOSS'	19.95
WICO '3-WAY'	32.95
WICO 'BAT HANDLE'	29.95
EASY SCRIPT	49.00
COMMODORE VIC CARTRIDGES	13.95
COMMODORE C-64 CARTRIDGES	22.95
BOX OF 10 DISKS WITH THIS AD	19.95

## 10% Discount to TPUG Members

ON SOFTWARE, BOOKS AND ACCESSORIES.  
(NOT INCLUDING SALE ITEMS)

**TPUG Library  
Available For Copy**



# ULTRA COPY 64

**DISK DUPLICATION SYSTEM  
FOR COMMODORE 64**

- Analyze disk tracks for data & errors
- Skip empty tracks to speed copying
- Copy everything

• Fast, reliable copying with 1 or 2 drives  
\$39.95 plus \$3 shipping. Mastercard and Visa

## C-64 ULTRA RESET SWITCH

- Built into new 6 foot disk drive cable
  - Nothing to solder - no connections
  - Eliminate voltage spikes & switch wear
- \$16.95 plus \$3 shipping. Mastercard and Visa

ULTRABYTE Call (313) 562-9855

23400 Michigan, Suite 502, Dearborn, MI 48124  
Satisfaction guaranteed, 10 day return privilege  
DEALER INQUIRIES INVITED

**NEW FOR HIGH SCHOOL  
ADMINISTRATORS**

**IF A STUDENT WANTS TO  
CHANGE HIS TIMETABLE...  
CAN YOU DO IT?**

**YES YOU CAN!**



The WYCOR Class Scheduling  
System lets you:

- Add new students, change existing timetables.
- Checks all available classes, class sizes.
- Displays 32 best timetables that minimize class sizes.
- Lets YOU pick the timetable.
- Prints the new timetable.
- Updates class size.
- Prepares a teachers change list.
- Other reports available.



**WYCOR BUSINESS SYSTEMS**

170 The Donway West, Suite 401

Toronto, Canada M3C 2G3

Tel. (416) 444-3492

# New Additions To The TPUG Library

(Access to library available to TPUG members only)

NOTE: Each List-Me File includes the following notation:  
 "Copyright ©1983 by Toronto PET Users Group Inc."  
 "OK to copy but is not to be sold or published for profit"

The **LIBRARY** to which a disk/tape belongs is indicated by the library code in brackets. This code appears as the first character in the three-character identification code:

(C) Commodore 64      (P) PET/CBM      (V) VIC 20      (S) SuperPET

If you wish to order disks or tapes from our library, please make sure that the programs you order are compatible with the computer you have.

## (V)T9-

### MAY 1984

LIST-ME(V)T9.L	LIST this file for description of programs on (V)T9.
HAM LOG DEMO.V	Demo for the HAM operator programs which follow.
LOG MASTER.V	Use these programs to create a master log of HAM operators.
LOG MASTER8K.V	The second is for an expander.
T&R STATS8K.V	Calculates statistics for a person or a student.
T&R STATS.V	As above, but no memory expander is required.
TERMINAL8K.V	Working upload/download terminal program for the VIC 20. Full documentation can be found on BBS's using a 'Punter' PET-type BBS program.
VISITING HRS8K.V	Mystery word game—wander through the hospital.
JEDI16K.V	Word game based on 'Star Wars'. Try to find the troopers and destroy the bases.

SLITHER DUEL3.V	'Snake' game with three enemies—keyboard.
SOUND DEMO.V	A few sound effects for you to enjoy.
GALACTIC CODE.V	Code making/breaking game to play with.
RATRAP16K.V	3D maze where you try to find the cheese in the least time—very good.
RESUCE 16K.V	Manoeuvre the ship down to earth and back up to the mothership without crashing into anything.
STAR TREK16K.V	Excellent 'Trek' game with many functions and features.
GALAX/BOOT.V	Keyboard 'Galaxian'-type game with custom characters.
GALAXIAN.V	Main part of the game. First load the above.
TERM.VIC	Machine language for the terminal program. Loaded automatically by the main program above. <b>You must have a disk drive to download!</b>

## (P)T9-

### MAY 1984

LIST-ME(P)T9.L	Documentation for disk (P)T9.
EVALUATOR P1.W	Education—Instructions for "THE EVALUATOR.P" program. "THE EVALUATOR.P" appeared on (P)T7—Use to evaluate student records. Use WordPro or "WP LIST/SCREEN.Z" from (P)T6. From D. Trankner.
EVALUATOR P2.W	Instructions—"THE EVALUATOR.P".
EVALUATOR P3.W	" " "
EVALUATOR P4.W	" " "
EVALUATOR P5.W	" " "
EVALUATOR P6.W	" " "
EVALUATOR P7.W	" " "
EVALUATOR P8.W	" " "
EVALUATOR P9.W	" " "
EVALUATOR P10.W	" " "
AID/LIST TLKIT.P	Utility—A Basic Aid-Toolkit type program.
AID/LIST TLKIT.W	Utility—Instructions for "AID/LIST TLKIT.P". Use WordPro or "WP LIST/SCREEN.Z" from (P)T6. From D. Trankner.
INTEGRATION.8	Math—Simple integration program from H. Anderson Electronics.
HARMONIC 3.8	Math—Fourrier analysis of any waveform.
ABIDE WITH ME.Z	Music.
BELLOWS EXT 1.4	Education-Photography tutorial 1 From Sheridan College & Commodore

BELLOWS EXT 2.4	Education-Photography tutorial	2
HYPERFCL DIST1.4	" " "	3
HYPERFCL DIST2.4	" " "	4
DEPTH FOCUS1.4	" " "	5
DEPTH FOCUS2.4	" " "	6
PHTPROC CNTRL1.4	" " "	7
PHTPROC CNTRL2.4	" " "	9
LIGHT 1.4	" " "	10
LIGHT 2.4	" " "	11
LIGHT 3.4	" " "	12
NEWMON-INST.P	Utility—Information on Newmon. Source is available in the C-64 library. From G. Fontaine.	
NEWMON-HELP E.P	Utility—English instructions for Newmon.	
NEWMON-HELP F.P	Utility—French instructions for Newmon.	
NEWMON\$1000.P	Utility—Monitor for the PET at Hex 1000.	
NEWMON\$7000.P	Utility—Monitor for the PET at Hex 7000.	
KEYPRINT@027A.F	Utility—Screen print F40. Use to execute. From R. Illes.	
KEYPRINT@027A.8	Utility—Screen print 8032. Use to execute.	
KEYPRINT@7F6E.F	Utility—Screen print F40. Use to execute.	
KEYPRINT@7F6E.8	Utility—Screen print 8032. Use to execute.	
ENCLOSURE SI.Z	Miscellaneous—Hi-Fi speaker enclosure calculations.	
WATERCONSUM SI.Z	Miscellaneous—How much water do you use? (surprising!!!)	

**TORONTO PET USERS GROUP**

1912A Avenue Rd., Ste. 1  
 Toronto, Ontario  
 M5M 4A1

416-782-9252

416-782-8900

**ORDER FORM**

Name .....

Membership # .....

Street Address .....

Telephone .....

City/Town/P.O. ....

Province/State .....

Postal/ZIP Code .....

The prices indicated include postage and handling as well as Ontario Provincial Sales Tax (if applicable).

Make cheque or money order payable to: TPUG.

**DISKS**

To order club disks by mail, send \$10.00 for each 4040/2031/1540/1541 disk (4040 format), and \$12.00 for each 8050/8250 disk (8050 format). We do honour purchase orders from school boards.

If you wish to order the total library to date for a specific computer (PET, SuperPET, VIC 20 or Commodore 64), contact the club office to find out how many disks there are currently. The cost is \$8.00 per disk (4040 format) and \$10.00 per disk (8050 format).

These disks are for use with a \_\_\_\_\_ computer and a \_\_\_\_\_ disk drive.

Please send me the following:

3 Letter/No. Code	Description	4040 or 8050 Format	Price
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____
Total			_____ .00

**TAPES**

To order VIC 20 or Commodore 64 library tapes, send \$6.00 for each tape. If you wish to order the total library to date for the VIC 20 or Commodore 64, contact the club office to find out how many tapes there are currently. The cost is \$5.00 per tape.

To order PET/CBM or Commodore Educational Software tapes, send \$10.00 for each tape.

These tapes are for use with a \_\_\_\_\_ computer and a datasette.

If for a PET computer – what model \_\_\_\_\_ – Basic – 1.0 ( ), 2.0 ( ), 4.0 ( )?

3 Letter/No. Code	Description	Price
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____
		_____ .00

# C-64 LIBRARY NOTES

David Bradley  
Toronto, ON

## The "M" Disks

Some of you may be wondering about the "M" disks that were listed in the June issue of the *TPUG Magazine*. To put it simply they are monthly disks that we didn't have months for.

I got such a flood of stuff in that if I didn't do something it would be (literally) Christmas before all of the stuff I had in February would be released. And then I would still have everything else that came in since February to take care of.

So just treat the "M" disks as you would (C)T6 or any other monthly disk.

## Star Trek V2

Any of you that have played Dave Neale's *Star Trek* game know how good it is. Well, Dave has been working hard since February when V1 was released and he is almost finished V2. It has many new features. For example you can now take prisoners and interrogate them to find out where other Klingon ships are hiding, the stars have been changed to planets, the Klingons will fire at you if you don't fire at them within a random period of time, and much, much more.

If you want to see V2 in the *TPUG* library be sure and send Dave the \$2 that he requested. Otherwise he alone will have the best *Star Trek* game in the galaxy...

## Strange Printers

Some people have written and called me to complain that some of the programs won't print to their printer. Some programs won't work with all printers. Since I have an MPS-801 I test the programs with that and if they work they go into the library. If a certain program doesn't like your non-Commodore printer I'm sorry, but I didn't volunteer to change programs so that they will work with strange equipment. Like I have said many times *'The only way to guarantee your peripherals will be*

*100% compatible is to stick to Commodore equipment.'*

## Fix for MOVING SIGN.C

Many of you may have noticed that the program *MOVING SIGN.C* crashes very soon into execution. The fix is simple. *LOAD* the program. *Delete line 5.* And re-save the program. That's it!

## Thanks, All Contributors

At this time I would like to say *thank you* to everyone who has sent in programs for the Commodore 64 library. I don't just mean the people here in Toronto, I mean the people all across Canada, New Zealand, Australia, The Netherlands, The United States of America, and everyone else that has submitted programs.

Remember, *one man's junk is another man's treasure.* In other words, I am still looking for more 'junk' (or treasure if you rather).

## Disk Tip

Here is one that they don't display in any of the disk envelopes that I have ever seen. When you mark your disks be sure to *never use pen* (or any other writing utensil with a hard tip). If you are not *extremely* careful you will damage your disk. The damage, as a rule, will not show up until you are saving the most *crucial file of your life*. If all you have is a pen, write on the sticker *before* you put it on the disk. The best thing you can do is run (or walk) out and get a felt tip marker.

## Staying Cool

One of my biggest worries when I first got my C-64 was heat. It didn't take long for me to realize that a small restaurant could operate over the C-64's power supply and the 1541 disk drive. So, I went out to a local hardware store and spent about \$40 on a variable speed oscillating fan. I set it up so it would blow on both my 1541 and the power supply. So far, after a year and a half, I have had no trouble with either unit. So far...

## Is It Worth It

I get many calls from people who are not yet members of the club because they don't have a computer yet but they will be going to 'the States' in a couple of weeks to get one.

The way I see it, going to 'the States' was worth it up until about September of 1983. At that time you could save anywhere from \$100 to \$200 on the C-64 alone. But now with the C-64 available here, in Canada for about \$350 I don't see the point. Sure, the C-64 in 'the States' is \$199. But after the *exchange rate and the duty* I think you save about \$20. And that is without taking the *gas money* into account.

The way I see it, even if it costs you a few bucks more, it is better to find a good dealer and buy from him. Then you know someone 'in' the business and you don't have to travel to 'the States' to have your warranty honoured. (In case you missed the implied point there, Commodore Canada *will not honour* the warranty of a machine purchased in 'the States'.) And with a flat rate of \$70 to fix the C-64 regardless of the problem, the money you thought you saved by going to 'the States' doesn't last long. *TPUG*





## (C)T9

By David Bradley

The following is a slightly abridged version of the documentation that was available at the TPUG Commodore 64 meeting. The only items which have been excluded are the start and finish addresses in both hexadecimal and decimal. If you need these, they are included in the complete version from the TPUG office for \$1.00.

**Note:** Programs with a '\*' in front of the name in this file will not work in tape therefore they are not included in the tape version of this disk.

### ABBREVIATIONS USED IN THIS LIST-ME

BL — Block Length  
 PL — Program Length  
 FT — File Type  
 CL — Classification

### New MENU System

The first 3 programs on this disk are menu programs. Simply LOAD "M\*",8 and RUN. The names of the first 14 programs on the disk will be displayed. If what you want is not visible, select #15. This will LOAD the second menu program. If you still don't see what you want, again pick #15 to LOAD the third menu. From menu 3 you can LOAD the remaining programs or the prior menu programs. If response to this type of menu system is good I will do all of the disks like this.

### EQUIPMENT ASSUMPTION

In the program description you should see a line that begins 'EQUIPMENT REQUIRED'. On this line will appear all the necessary peripherals that you will need to operate the program in question. Please note that it is assumed that you have a Commodore 64 and some sort of monitor or television set. Therefore these items will not be listed on this line.

### LIST-ME (C)T9.L

This file contains one-line documentation for the programs on (C)T9. If you have a printer, print it out and keep it close by.

### \*DIR SORT V1.C

*LOAD as follows: LOAD "DIR SORT V1.C";8  
 BL: 53, FT: PRG, CL: UTILITY  
 Equipment required: 1541 disk drive.*

This program will let you read the directory of any diskette into your Commodore 64, look at it, and then sort it and write it back out to the original diskette. Handy if you like everything alphabetized.

### \*BAM.C

*LOAD as follows: LOAD "BAM.C";8  
 BL: 15, FT: PRG, CL: UTILITY  
 Equipment required: 1541 disk drive.*

This program allows you to see where your material is stored on disk.

The purpose of the BAM is to protect allocated blocks so they are not written over and therefore destroyed. The map resides on the directory track (track 18, sector 0). The BAM turns off bits when it allocates a sector. This program looks at these bits and if on (meaning it is free) will print either a red or a green square. By looking at the map you can determine how full or empty the disk is.

### \*BLOCK FREE.C

*LOAD as follows: LOAD "BLOCK FREE.C";8  
 BL: 2, FT: PRG, CL: UTILITY  
 Equipment required: 1541 disk drive.*

This little program will tell you how many blocks free there are on your diskette. A handy little subroutine to have around.

### \*BOOT DOS 5.2.C

*LOAD as follows: LOAD "BOOT DOS 5.2.C";8  
 BL: 1, FT: PRG, CL: UTILITY  
 Equipment required: 1541 disk drive.*

This program LOADs and executes DOS5.2.D.

### \*DOS5.2.D

*Do NOT attempt to LOAD this file.  
 BL: 4, FT: PRG, CL: DATA*

A machine language program that is LOADED and used by BOOT DOS 5.2.C.

### \*COCKROACH 1D.C

*LOAD as follows: LOAD "COCKROACH 1D.C";8  
 BL: 6, FT: PRG, CL: UTILITY  
 Equipment required: 1541 disk drive.*

This program will copy the entire contents of one 1541 diskette to another using one 1541 disk drive. Just follow the instructions that the program gives you and you should have no troubles.

**Note:** Just to be safe, put a write protect sticker over the notch of your ORIGINAL diskette.

### \*COCKROACH 1.C

*LOAD as follows: LOAD "COCKROACH 1.C";8  
 BL: 6, FT: PRG, CL: UTILITY  
 Equipment required: 1541 disk drive.*

This program will copy the entire contents of one 1541 diskette to another using one 1541 disk drive. Just follow the instructions that the program gives you and you should have no troubles.

**Note:** Just to be safe, put a write protect sticker over the notch of your ORIGINAL diskette.

### \*COCKROACH 2D.C

*LOAD as follows: LOAD "COCKROACH 2D.C";8  
 L: 7, FT: PRG, CL: UTILITY  
 Equipment required: 2 1541 disk drives.*

This program will copy the entire contents of one 1541 diskette to another using two 1541 disk drives. Just follow the instructions that the program gives you and you should have no troubles.

**Note:** Just to be safe, put a write protect sticker over the notch of your ORIGINAL diskette.

### \*COCKROACH E.C

*LOAD as follows: LOAD "COCKROACH E.C";8  
 BL: 5, FT: PRG, CL: UTILITY  
 Equipment required: 1541 disk drive.*

This program will check every track and sector of your diskette for errors.

### \*DIR READ.C

*LOAD as follows: LOAD "DIR READ.C";8  
 BL: 8, FT: PRG, CL: UTILITY  
 Equipment required: 1541 disk drive.*

This is a handy little subroutine that reads the directory of a diskette into memory. It can be easily added to your programs where you need such a feature.

### \*FAST BAM.C

*LOAD as follows: LOAD "FAST BAM.C";8  
 BL: 10, FT: PRG, CL: UTILITY  
 Equipment required: 1541 disk drive.*

This program will show you much faster than previous similar programs what areas of your disk are used.

### BAMGET.D

*Do NOT attempt to LOAD this file.  
 BL: 1, FT: PRG, CL: DATA*

A machine language program that is LOADED and used by FAST BAM.C.

## FUNCTION KEYS.C

*LOAD as follows: LOAD "FUNCTION KEYS.C";8*

*BL: 4, FT: PRG, CL: UTILITY*

This program, as is, will assign the following keywords to your Commodore 64's function keys.

F1 - LOAD "\$",8  
F2 - LIST  
F3 - RUN  
F4 - LOAD  
F5 - GOSUB  
F6 - GOTO  
F7 - RETURN  
F8 - SAVE

If you want to change these, simply list the program and change the values of F\$ in lines 2 through 9 before you RUN it. Good luck. . .

## \*LOCK UNLOCK.C

*LOAD as follows: LOAD "LOCK UNLOCK.C";8  
BL: 5, FT: PRG, CL: UTILITY*

*Equipment required: 1541 disk drive.*

This program allows you to 'lock' a file on your disk so that it can not be erased before you 'unlock' it. A handy little safety device.

## \*RECOVER 1541.C

*LOAD as follows: LOAD "RECOVER 1541.C";8  
BL: 15, FT: PRG, CL: UTILITY*

*Equipment required: 1541 disk drive.*

This program will go through a disk and show you the names of any files that have been scratched. If you see one that you didn't want to scratch and would like to recover it, input the number beside the name of the program and if it can be recovered, it will be.

## \*REL FILE BOOT.C

*LOAD as follows: LOAD "REL FILE BOOT.C";8*

*BL: 3, FT: PRG, CL: UTILITY*

*Equipment required: 1541 disk drive, MPS-801 or 1526 printer.*

This program will LOAD and execute a very good RELative file handling program. Once you set up your file and put a freshly formatted disk in your 1541 this program will create your RELative file and let you write to the file, read from the file, update the file, modify the file, search the file, sort the file, delete parts of the file, restart the program from the beginning, print the file, add more records to the file, copy the contents of one record to another and look at the directory of your DATA disk.

This program looks like it was very well thought out and put together. Good luck. . .

## \*ULTRASORT/MC.D

*Do NOT attempt to LOAD this file.*

*BL: 5, FT: PRG, CL: DATA*

This is a DATA file that is LOADED and used by REL FILE BOOT.C.

## \*FILE MANAGER.D

*Do NOT attempt to LOAD this file.*

*BL: 82, FT: PRG, CL: DATA*

This is a program LOADED and executed by REL FILE BOOT.C

## SPRITE DATA.C

*LOAD as follows: LOAD "SPRITE DATA.C";8  
BL: 14, FT: PRG, CL: UTILITY*

This program will allow you to design a sprite on your monitor and then when you are finished it provides you with the DATA statements for your sprite. That way you can create your sprites, save them to your disk or tape as programs and later merge them into a program that uses them. Handy, eh?

## SUPER MIND.C

*LOAD as follows: LOAD "SUPERMIND.C";8  
BL: 30, FT: PRG, CL: GAME*

This program is just like the game MASTERMIND. You have to try and figure out what combination of colours the computer is hiding.

When you guess the computer will tell you how many of the colours that you have picked are in the solution. If the colour is OK, but it is in the wrong position you will get a white circle. If the colour is OK and it is in the right position you get a black circle. From the white and black circles you have to figure out which colours are in the solution. You have eight guesses to find the solution so think every move out VERY carefully. Good luck. . .

## LABEL MAKER.C

*LOAD as follows: LOAD "LABEL MAKER.C";8  
BL: 64, FT: PRG, CL: BUSINESS*

*Equipment required: MPS-801 or 1526 printer.*

This program allows you to make up labels for disks or whatever else you want to label. It allows you to keep, within the program, various labels that you use often. (If you do add to the program, be sure and save it on disk that was formatted in a 1541.)

This program was written to be used with the Gemini 10X but it will (except for the condensed modes) work with a 1525 or MPS-801. Good luck. . .

## \*COPY-ALL V2.C

*LOAD as follows: LOAD "COPY-ALL V2.C";8  
BL: 10, FT: PRG, CL: UTILITY*

*Equipment required: 2 1541 disk drives.*

This is an updated version of COPY-ALL. The only difference between this and the original is that this will copy RELative files.

This program will copy any or all of a 1541 diskette to another using two 1541 disk drives. Be sure that you change one of the 1541's device number. The program that does this is on the disk that Commodore includes with the 1541 so you should have it. If I am not mistaken it is called 'DISK ADDR CHNGE' or something similar.

One thing that stumps a lot of people is the question 'Pattern?'. Suppose you were looking at (CJT6 (TPUG February C-64 disk) and you wanted to copy Dave Neale's STAR TREK game. When the 'Pattern?' prompt come up if you were to enter 'STAR TREK\*' the computer would know that it is to search for all files starting with 'STAR TREK' and not bother with any of the others.

If you just press RETURN on the '\*' it will show you all the file names. If you knew that the only program you wanted to copy was called 'BBBBS' all you would do is type in BBBBS and the program will only look for that program.

One nice feature of this program is that if you have picked everything you want to copy but still a lot of disks to go through once you answer 'N' to one of the programs, if you press RETURN it will skip past the rest of the programs on the disk automatically. Nice, eh?

I hope this clears things up for some of you. Good luck. . .

## BIG SET.C

*LOAD as follows: LOAD "BIG SET.C";8  
BL: 40, FT: PRG, CL: UTILITY*

This program will alter the Commodore 64's character set in such a way that you can make characters 4 times the size of the standard characters.

'How?' you ask. Well, suppose you wanted to print out the word BALD. To make the big characters work you have to put the standard character followed by the capital character on the first line. On the second you have to do the same except put them in reverse field.

Sound hard? Don't worry, it really isn't! When you LOAD and RUN this program just read the instructions carefully and look at what they have done. Good luck. . .

## SINE IN.C

*LOAD as follows: LOAD "SINE IN.C";8  
BL: 4, FT: PRG, CL: DEMO*

*After you input a word this program will display your word in a sine wave pattern down your monitor.*

## \*TINY DIRECTORY.C

*LOAD as follows: LOAD "TINY DIRECTORY.C";8*

*BL: 8, FT: PRG, CL: UTILITY*

*Equipment required: 1541 disk drive, MPS-801 or 1526 printer.*

This program will print out the directory of your disk in a 2 column format.

#### \*DISK MAP.C

*LOAD as follows: LOAD "DISK MAP.C";8  
BL: 44, FT: PRG, CL: UTILITY  
Equipment required: 1541 disk drive, MPS-801  
or 1526 printer.*

After you choose which output device you want (3 is the screen, 4 is the printer) this program will read the directory of the disk in drive 0 of device 8. It will print out a directory listing, including the initial track and sector of each file.

Then it reads every block on the disk to get the first two bytes (i.e. file system links). Following this it summarizes the directory. Then it looks for inconsistencies in disk file structure, describing any errors it finds. Then a complete map of the disk can be printed.

Once all this is done the program switches to interactive mode. For more information on interactive mode and all the other features of this program be sure and LOAD and LIST 'LIST-ME MAP.L' to your printer.

#### \*LIST-ME MAP.L

*LOAD as follows: LOAD "LIST-ME MAP.L";8  
BL: 23, FT: PRG, CL: LIST-ME*

This file contains detailed information about what DISK MAP.C does. Before you attempt to use DISK MAP.C be sure and LOAD and LIST this file to your printer.

In case you don't know how to list a program to a printer, do the following after you have loaded the program you want to LIST.

OPEN4,4 : CMD4 : LIST (RETURN)  
When the program has finished LISTing, enter the following:  
PRINT#4 : CLOSE4 (RETURN)  
Good luck. . .

#### \*DIR LIST SORT.C

*LOAD as follows: LOAD "DIR LIST SORT.C";8  
BL: 38, FT: PRG, CL: UTILITY  
Equipment required: 1541 disk drive, MPS-801  
or 1526 printer.*

This program will help you keep track of what disks have which programs. After you feed in the disks you can print out a list of what is where. You'll never have to go rummaging through a stack of disks again!

#### \*CHOPPER.C

*LOAD as follows: LOAD "CHOPPER.C";8  
BL: 8, FT: PRG, CL: DEMO  
Equipment required: joystick.*

A sprite demo with a helicopter that you can control using a joystick plugged into control port #2.

Some things you will need to know is that f1 increases your speed, f7 decreases the speed, up arrow ends the demonstration, left arrow halts your copter, and d toggles the auto pilot. That should get you going. Good luck. . .

#### \*CHOPPER.D

*Do NOT attempt to LOAD this file.  
BL: 2, FT: PRG, CL: DATA*

A machine language DATA file that is LOADED and used by CHOPPER.C.

#### STOCK.C

*LOAD as follows: LOAD "STOCK.C";8  
BL: 39, FT: PRG, CL: GAME*

This program is a simulation of the operations of the stock market. There are 12 company stocks available for trade providing you have the money.

There is a 6% or \$50.00 per stock transaction charge. Transactions are collected at the end of each week.

You have 4 main options. You can BUY stock, SELL stock, PASS (leave things as they are), or you can end the game. When you end the game you are presented with a summary of how well you did.

**Note:** When you are buying or selling be sure and enter at least the first 3 letters of the name of the stock. For instance if you wanted to sell 25 shares of Exxon Oil all you have to enter when the program asks you for the name of the stock is 'EXX'. A nice feature that saves a lot of typing. Good luck. . .

#### \*CLASS PROGRAM.C

*LOAD as follows: LOAD "CLASS PROGRAM.C";8  
BL: 24, FT: PRG, CL: EDUCATION  
Equipment required: 1541 disk drive, MPS-801  
or 1526 printer.*

Use this program to keep track of your classes. This program also has a printer option so even if the 'kids' get your data disk you still have their marks.

#### MULTI-AID.C

*LOAD as follows: LOAD "MULTI-AID.C";8,1  
BL: 25, FT: PRG, CL: UTILITY*

Multi-Aid is an integrated package of programming utilities put together by Steven Darnold.

Once the program has LOADED, activate it entering SYS 12\*4096. You will then see a summary of instructions. These can be recalled at anytime by entering SYS 12\*4096. This will not affect the BASIC program in memory.

You can save a copy of Multi-Aid to disk by entering the machine language monitor and typing .S"MULTI-AID.C",08,B000,

C800. If you want a tape copy, change the 08 to 01. Note, however, that tape copies always load with a ?LOAD ERROR. This occurs because part of Multi-Aid goes under BASIC and this invalidates the checksum. If ST equals 32 on a cassette LOAD, just ignore the ?LOAD ERROR.

Once you have LOADED and activated Multi-Aid the function keys now do the following.

f1 will delete everything from the position of the cursor to the end of a line.

f1 plus the CONTROL key provides an escape from quote mode.

f1 plus the SHIFT key freezes the computer until the SHIFT key is released.

f3 performs a disk LOAD.

f4 puts you into the machine language monitor. (To exit this mode type X and press return.)

f5 displays the disk drive's error channel.

f6 puts the command OPEN 1,8,15," on the screen. You can use this to undertake disk operations, such as scratching files or formatting disks.

f7 changes the screen background colour.

f8 changes the screen border colour.

You also have 7 new BASIC commands. They are RENUM, DEL, FIND, CHANGE, OLD, APPEND, and OFF.

As well Multi-Aid adds a machine language monitor. The commands used by it are as follows:

.S Save range as PRG to device.

.L Load PRG from device.

.T Transfer range to new address.

.F Find bytes in range (' for ASCII).

.C Change entire range specified byte.

.A Assemble machine language at address.

.D Disassemble 22 line from address.

.P Disassemble over range.

.M Display memory contents over a range.

.R Display registers.

.G Begin execution at address.

.X Return to BASIC.

Yes, sir. Quite a little utility. . .

#### LIST ASCII \$C0.C

*LOAD as follows: LOAD "LIST ASCII \$C0.C";8  
BL: 12, FT: PRG, CL: UTILITY  
Equipment required: printer.*

This program will list the program in memory to an ASCII printer. All you have to do is enter SYS 12\*4096.

All of the cursor movements will be translated into something that you and your printer can understand. For example:

Cursor up will be printed as <up>.

Cursor down will be printed as <dn>.

Cursor left will be printed as <le>.

Cursor right will be printed as <rt>.

A handy utility to have if you have an ASCII printer.

#### LIST ASCII \$9D.C

LOAD as follows: LOAD "LIST ASCII \$9D.C";8  
BL: 12, FT: PRG, CL: UTILITY  
Equipment required: printer.

## (C)M6

#### New MENU System

The first 2 programs on this disk are menu programs. Simply LOAD "M\*",8 and RUN. The names of the first 14 programs on the disk will be displayed. If what you want is not visible, select #15. This will LOAD the second menu program. From menu 2 you can LOAD the remaining programs or the prior menu program. If response to this type of menu system is good I will do all of the disks like this.

#### EQUIPMENT ASSUMPTION

In the program description you should see a line that begins 'EQUIPMENT REQUIRED'. On this line will appear all the necessary peripherals that you will need to operate the program in question. Please note that it is assumed that you have a Commodore 64 and some sort of monitor or television set. Therefore these items will not be listed on this line.

#### LIST-ME (C)M6.L

This file contains one-line documentation for the programs on (C)M6. If you have a printer, print it out and keep it close by.

#### STAR TREK INST.C

LOAD as follows: LOAD "STAR TREK INST.C";8  
BL: 27, FT: PRG, CL: INSTRUCTIONS

This file contains a lot of information that you will need to know BEFORE you attempt to play STAR TREK 84.C.

So, please LOAD and RUN this program first!

#### STAR TREK 84.C

LOAD as follows: LOAD "STAR TREK 84.C";8  
BL: 131, FT: PRG, CL: GAME

This is another version of STAR TREK. It was written by Mike Omotani. Mike decided to add a lot of new commands to the game to make life for you, the captain of the Starship Enterprise, a bit easier. Many of the commands may be unfamiliar to many of you captains that are used to the Enterprise before Mike got his hands on it so the following is a list of the commands you will need to know to operate this 'new' Enterprise.

This program will list the program in memory to an ASCII printer. All you have to do is enter SYS 40192.

All of the cursor movements will be translated into something that you and your printer can understand. For example:

- NAV - To set course.
- SRS - For *short range sensor scan*.
- LRS - For *long range sensor scan*.
- PHA - To fire *phasers*.
- TOR - To fire *photon torpedoes*.
- SHE - Status of *shields*.
- DAM - For *damage control reports*.
- COM - To call on library- *computer*.
- 0 - Cumulative Galactic Record.
- 1 - Status Report.
- 2 - Photon Torpedo Data.
- 3 - Starbase Nav Data.
- 4 - Direction/Distance Calculator.
- 5 - Galaxy 'Region Name' Map.
- 6 - Inventory Of Devices On Board WRP  
- To change warp speed.
- SHD - To *discharge shields*.
- RES - To *remain stationary*.
- SHT - To *send out shuttle craft*.
- SPM - To *send subspace message*.
- SPJ - To *use space jump*.
- DRD - To *send out droids*.
- ROB - To *use damage repair robot*.
- CLK - To *use cloaking device*.
- LRP - To *use long range probe*.
- MIN - To *detonate space mines*.
- XXX - To *resign your command*.

This list of commands will mean a lot more to you if you LOAD and RUN STAR TREK INST.C.

Good luck. . .

#### ADDING QUIZ.C

LOAD as follows: LOAD "ADDING QUIZ.C";8  
BL: 60, FT: PRG, CL: EDUCATION

This program will display adding questions of varying degrees of difficulty. Depending on the level called for, one or more of the digits in question will be blanked out.

The student's task will be to enter the correct digit(s).

Any number of questions at any level can be called for.

#### REPEAT.C

LOAD as follows: LOAD "REPEAT.C";8  
BL: 17, FT: PRG, CL: GAME

When you are ready, a pattern of 4 numbers will be displayed. Immediately one will quickly flash. You must repeat that same number by pressing the same key as that number. If you get the correct number, it will be displayed again, followed by another number (watch closely because the second number could be the same as the

Cursor up will be printed as <up>.  
Cursor down will be printed as <dn>.  
Cursor left will be printed as <le>.  
Cursor right will be printed as <rt>.  
A handy utility to have if you have an ASCII printer.

first). Once again you must repeat the sequence—the first and the second etc. . . This continues until you've made 20 correct numbers in a row. The computer plays the sequence only once a turn.  
Good luck. . .

#### \*DOS COMMANDS.C

LOAD as follows: LOAD "DOS COMMANDS.C";8  
BL: 26, FT: PRG, CL: UTILITY  
Equipment required: 1541 disk drive, MPS-801 or 1526 printer.

This program will read the directory of you disk into memory and allow you to do any of the following to or with the diskette in your drive.

You can LOAD any program, you can LOAD and RUN any program, format a new disk, validate the disk, rename a file on the disk, scratch a file from the disk, list the directory to your printer, or LOAD and LIST a program to your printer.  
Good luck. . .

#### MATH MAGIC.C

LOAD as follows: LOAD "MATH MAGIC.C";8  
BL: 63, FT: PRG, CL: EDUCATION

This program will test you on addition, subtraction, multiplication, or division at any of four levels (simple, easy, hard, and hardest of all). When you answer a question correctly you get some treasure. The Treasure will be one of a number of precious metals and gems. If you answer incorrectly you lose some of your treasure. If you lose all your treasure you will perish!  
Good luck. . .

#### AUTOLINE PRINT.C

LOAD as follows: LOAD "AUTOLINE PRINT.C";8  
BL: 6, FT: PRG, CL: UTILITY

You can use this program to print consecutive specified intervals. For example if you were writing a music program and you knew that you were going to need DATA statements from line 250 to line 550 numbered by 10's all you have to do is tell this program where to start, where to end and what the interval is and it will do it for you.  
Handy, eh?

### HAPPY NEW YEAR.C

*LOAD as follows: LOAD "HAPPYNEWYEAR.C",8*

*BL: 12, FT: PRG, CL: MUSIC SOUND*

*Equipment required: a monitor or TV with audio output.*

This program displays (in big letters) 'Happy New Year' while flashing the screen by changing colours rapidly and playing that old time favourite 'Auld Lang Syne'. Great for New Year's Eve parties and get togethers.

### \*HARMONIZER.C

*LOAD as follows: LOAD "HARMONIZER.C",8*

*BL: 18, FT: PRG, CL: MUSIC SOUND*

*Equipment required: a monitor or TV with audio output.*

This program allows you to make up your own songs and save them to disk as SEQuential files.

After you have saved your symphonies you can LOAD them back in and impress your Apple friends with the superior sound capabilities of your Commodore 64. Good luck. . .

### \*RED RIVER.D

*Do NOT attempt to LOAD this file.*

*BL: 26, FT: SEQ, CL: DATA*

This is a sample song that can be LOAded into HARMONIZER.C. This file is the DATA for that old favourite 'Red River Valley'. Great to impress your relatives with when they ask you what your computer can do. Good luck. . .

### STORYTELLER.C

*LOAD as follows: LOAD "STORYTELLER.C",8*

*BL: 32, FT: PRG, CL: EDUCATION*

This program will allow you to read about the varied adventures of a mouse, what he does, who he meets, and who his friends are.

### JOYSTICK DRAW.C

*LOAD as follows: LOAD "JOYSTICK DRAW.C",8*

*BL: 7, FT: PRG, CL: DEMO*

*Equipment required: joystick.*

This program will allow you to draw on the hi-res page using a joystick plugged into control port #2.

To change screen colours, press the fire button. To change the border colours press fire while pulling back on the joystick. To change the colour of your lines press the

space bar.  
Have fun. . .

### KEYBOARD BEEP.C

*LOAD as follows: LOAD "KEYBOARD BEEP.C",8*

*BL: 2, FT: PRG, CL: MUSIC SOUND*

*Equipment required: a monitor or TV with audio output.*

This small program will make your Commodore 64 beep every time a key is pressed. Just what you've been waiting for, eh?

### DOUBLE SPACER.C

*LOAD as follows: LOAD "DOUBLE SPACER.C",8*

*BL: 1, FT: PRG, CL: UTILITY*

This program modifies your Commodore 64 so that it double spaces everything that it prints on your monitor.

LOAD it into a friends Commodore 64 and have a laugh telling them that it is broken. Heh heh.

### 1526HIRESDUMP.C

*LOAD as follows: LOAD "1526 HIRES DUMP.C",8*

*BL: 4, FT: PRG, CL: UTILITY*

*Equipment required: 1526 printer.*

This program will dump hires pictures (like those found on TPUG disks (C)G1 & (C)G2) to your 1526 printer.

Now you can decorate your computer room with pictures of Winston Churchill, Donald Duck, and/or all the gang from Sesame Street.

### \*DISK FIDDLER.C

*LOAD as follows: LOAD "DISK FIDDLER.C",8*

*BL: 7, FT: PRG, CL: UTILITY*

*Equipment required: 1541 disk drive.*

With this program you can read any sector from any track into the memory of your Commodore 64. Once you have it in memory you can alter it and then re-save it back to your diskette.

### FRACTAL.C

*LOAD as follows: LOAD "FRACTAL.C",8*

*BL: 3, FT: PRG, CL: DEMO*

This program draws an interesting shape on the high resolution page of your Commodore 64.

### ARCADE.C

*LOAD as follows: LOAD "ARCADE.C",8*

*BL: 18, FT: PRG, CL: GAME*

This is a collection of several little games.

You are presented with a menu that looks like this:

1. . . Three liner (use shift)
2. . . Six line wonder
3. . . Car drive
4. . . Russian roulette
5. . . Stars
6. . . Target
7. . . Quit

To play the game press the number that corresponds to the name of the game. Once you lose at one of the games you are zipped back to the menu and you are free to try another. Have fun. . .

### LANDER.C

*LOAD as follows: LOAD "LANDER.C",8*

*BL: 12, FT: PRG, CL: GAME*

Your job is to pilot your landing craft from its launch platform to the base in the mountains. You must land on the red strip very gently and carefully.

You have 3 different thrust controls. f5 is the strongest and is handy to get you out of somewhere in a big hurry. f3 is not quite as strong and should be used most of the time. f1 is for very delicate navigation. You'll see once you start playing the game where you'll need it.

You also have lateral jet control. The '>' controls the right jet on your landing craft and the '<' controls the left jet.

You have a choice of how much fuel you want to take with you depending on how good a pilot you are. Good luck. . .

### COMPUTERS.C

*LOAD as follows: LOAD "COMPUTERS.C",8*

*BL: 14, FT: PRG, CL: DEMO*

This program will tell you all of the things that computers can't do. (Note: if you think of other things besides what is already included in this program you can put them in by adding more DATA statements within the program and re-saving the program to a 1541 formatted diskette.)

### THE BUS BARNS.C

*LOAD as follows: LOAD "THE BUS BARNS.C",8*

*BL: 42, FT: PRG, CL: EDUCATION*

You have been hired to park 12 buses. But before you can park any of them you have to answer the question displayed in the barn that you are to park in.

You can pick the type of question when you apply for the job. Have fun. . .

### THE WALRUS.C

*LOAD as follows: LOAD "THE WALRUS.C",8*

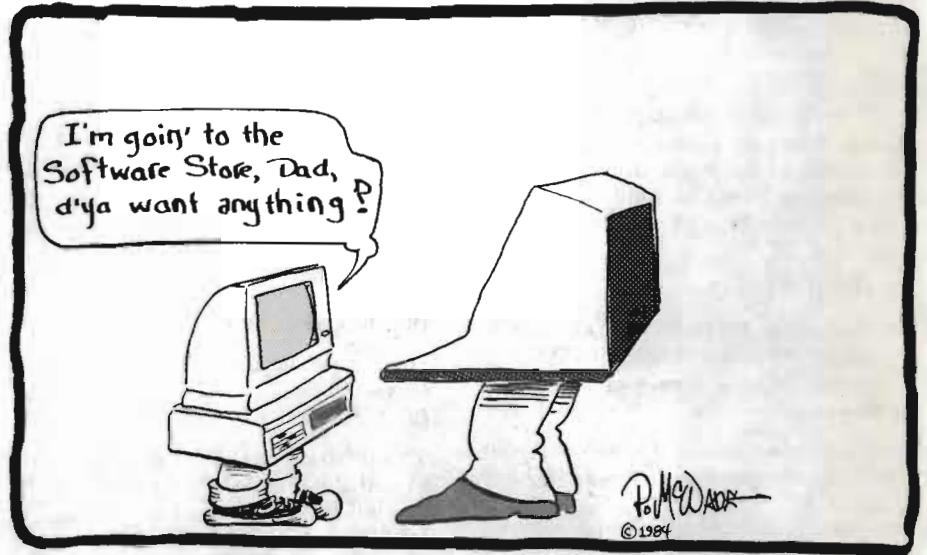
BL: 19, FT: PRG, CL: SOUND MUSIC  
 Equipment required: a monitor or TV with audio output.

This program will play display the lyrics and play a song about a family that has lost their mother to a computer.  
 A sad tale indeed. . .

**ENCLOSURE.C**

LOAD as follows: LOAD "ENCLOSURE.C";8  
 BL: 26, FT: PRG, CL: UTILITY

This program is designed to compute the optimum enclosure volume for a specific bass driver (woofer) provided that three pieces of information are given. These are:  
 Free-air resonance (HZ)  
 Compliance volume (Cubic Metres)  
 Mechanical 'Q' of driver  
 The free-air resonance is the frequency at which the impedance of the driver is the greatest.  
 The compliance volume of the driver is the enclosure volume that will give a 0-db



# dealer's MARKETPLACE

<p>This space could be advertising <b>YOUR</b> product</p>	<p>20 &amp; 64 computer  <b>Game Rentals</b>      by the week      3 LOCATIONS    <b>COMPURENT</b>      416-823-2368</p>	<p><b>Bradley Brothers</b> <i>Bulletin Board System</i>      c/o Richard and David Bradley      147 Roe Avenue      Toronto, Ontario      M5M 2H8  <b>ONLY \$10.00 per year</b>        (All cheques should be payable to David Bradley) or call <b>B.B.B.B.S.</b>      (416) 487-5838        as seen on The Journal</p>
<p><b>Electronics 2001</b>      5529 Yonge St, Toronto, Ontario      M2N 5S3 (416) 223-8400        Syntax offer for C-64      Cyclons — \$ 9.95      Bubbles — \$ 9.95      Scribbler — \$14.95        Ridiculously low prices!!!      Watch this space for monthly specials</p>	<p><b>COMMODORE SOFTWARE AFICIONADOS</b>      You'd like free software, we'd like reviews. The Book Company seeks additional reviewers for The Book of Commodore Software. For details, write and send a sample review to: The Book Company, 11223 S. Hindry Ave., Los Angeles, CA 90045</p>	<p><b>MARKET PLACE</b></p>

---

---

# CALENDAR OF TPUG EVENTS

---

## Summer-1984-Schedule

Generally, there are no monthly meetings in July and August, though there will be a **Basic BASIC Course** at York Public Library, 1745 Eglinton Ave. W., on alternate **Thursdays**, July 12 & 26, Aug. 9 & 23, and **Sept. 6** from 7:30 p.m. to 9:30 p.m. No previous knowledge or experience is required, just a willingness to learn. You may attend whichever sessions

you want. Details are available in this issue of *TPUG Magazine* or from the TPUG office.

Are you interested in organizing some other interest group in the Greater Toronto area? Please let the club office know, by mail, phone, or TPUG bulletin board.

A full calendar of events for the coming fall season will appear in the next (August/September) issue of *TPUG Magazine*.

---

## TPUG Associate Club Chapter Meetings

---

### CANADA

#### Edmonton Commodore Users Group

– meets at Archbishop Jordan High School, Sherwood Park, on the last Friday of each month at 7:00 p.m.

Contact **Bob Kadylo** 403-465-3523

#### Guelph Computer Club

– meets at Co-operators Insurance Assoc. on the 2nd Wednesday of each month at 7:30 p.m.

Contact **Brian Grime** 519-822-4992

#### London Commodore Users Club

– meets at the University of Western Ontario, in Room 40 of the School of Business Administration, on the last Monday of each month at 7:00 p.m.

Contact **Dennis Trankner** 519-681-5059

#### Sarnia C-64 Users Group

– meets at Lambton College on the first Sunday of each month at 7:30 p.m.

Contact **J. C. Hollemans** 519-542-4710

#### Commodore Users Club of Sudbury

– meets at Lasalle High School in the cafeteria on the last Thursday of each month at 7:00 p.m.

Contact **Tim Miner** 705-566-9632

#### PET Educators Group (Windsor)

– meets at Faculty of Education Building, 600 3rd Concession, Windsor, on the 3rd Wednesday of each month (not July & August) at 7:00 p.m.

Contact **John Moore** 519-253-8658

#### Associate Club Chapter Meetings

### UNITED STATES

#### Commodore Houston Users Group (Texas)

Meetings – Nassau Bay City Hall, Clear Lake City, on the 1st Wednesday of each month at 7:00 p.m.

– Bleyl Jr. High School, 10,000 Mills Rd. (Cypress-Fairbanks District), on the 3rd Thursday of each month at 6:30 p.m.

– Hildebrandt Middle School, 22,800 Hildebrandt Rd. (Klein ISD), on the 3rd Tuesday of each month at 6:30 p.m.

Contact **Mary F. Howe** 713-376-7000

#### Genesee County Area Pet Users Group (Michigan)

– meets at Bentley High School on Belsay Rd. on the 3rd Thursday of each month at 7:00 p.m.

Contact **Gordon Hale** 313-239-1366

#### Michigan's Commodore 64 Users Group

– meets at Warren Woods High School in Warren on the 3rd Tuesday of each month at 7:00 p.m.

Contact **Chuck Ciesliga** 313-773-6302

#### Russellville Commodore Users Group, Inc.

– meets at Oakland Heights Elementary School on the 3rd Thursday of each month at 7:00 p.m.

Contact **Bob Brazeal** 501-967-1868

#### Sacramento Commodore Computer Club (California)

– meets at Hiram Johnson High School, 65th St. & 14th Ave. in the cafeteria on the 4th Monday of each month at 7:00 p.m.

Contact **Geoff Worstell** 916-961-8699

#### Southern Minnesota Commodore Users Group

– meets at Mankato State University on the first Thursday of each month at 7:30 p.m.

Contact **Dean Otto** 507-625-6942

#### Westmoreland Commodore User's Club

– meets at Westmoreland County Community College (Youngwood PA) on the 3rd Friday evening of each month

Contact **Bob McKinley** 412-863-3930

### INTERNATIONAL

#### Baden Computer Club (Germany)

– meets at CFB Baden-Soellingen on the 2nd Sunday of each month at 7:00 p.m.

Contact **Kevin Rowe**

# SCS PAYROLL C-64

- Easy to use. No special training required
- SCS Payrolls have been used by Canadians since 1980
- Calculates & Prints Gross Pay, Overtime and Net Pay
- Prints Cheques or Pay Statements – User Choice
- Prints Employee Files with Cumulative Totals
- Monthly Report Gives Journal Entry and Receiver General Remittance Data
- Prints T4s at Year End
- Uses Formulae Recommended by Revenue Canada
- Complete Audit Trail
- Up to 50 Employees on 1 Data Disk

## \$139.

Dealer Inquiries Welcome

### SCS VICPAY

Calculates & Displays  
Gross Pay, Deductions  
and Net Pay  
\$24.95



Skeena Computer Services Ltd.  
Box 437 KITIMAT BC  
V8C 2R9 (604) 632-4212

#### Advertisers' Index

Academy Software	21
Aurora Software Inc.	33
Bradley Brothers Bulletin Board	62
Beacon Software Inc.	30
Bent Computer Associates	30
Besco Products	18
Book Company (The)	62
Brantford Educational Services	17
Comal Users Group U.S.A. Ltd.	33
Compuent	62
Compu-Simple Simon	21
Comspec	53
Creative Computer Systems	14
Electric Software Company	21
Electronics 2001	27&62
Friendly Software	14
Genealogy Software	17
High Resolutions Technologies	18
Hunter Nichols	IBC
King Microware	1
Mannesmann Tally	52
Microcomputer Solutions	30
Milne's Computer Control Systems	26
Nibble Notch	48
Precision Software	2
Pro-Line Software	IFC
P Technologies	40
Richvale Telecommunications	OBC
Romaro Enterprises International	46
Sheridan College	39
Skeena Computer Services Ltd	64
Software Lab	33
Toronto PET Users' Group	17&39
Ultrabyte	53
Wycor Business Systems	53

## classified

This space is limited to TPUG member wanted or for sale items only.

Space cost is 25 cents per word. NO DEALER ADS ACCEPTED

\*I have one Silicon Office Data Base Program for the Commodore 8096 with the Commodore D9090 hard disk drive. \$900.00 U.S. or best offer. Larry Rood, Gryphon House, P.O. Box 275 Mount Rainier, MD 20712 USA, tel. 301/779-6200

\*For Sale: CBM 2001, 4040, 4022, 8010, A D cnvtr, Software and 16 PET books. \$1500.00 U.S. Call Dave Gailius 213 321-4350 days, 714 870-9919 evenings

\*Commodore PET 32K, Dual Disk Drive, TX80 Printer & lots of business, educational and games software \$2,200 or offer. 446-1035

\*8250 Disk Drive for sale, or trade for 8050. Call Darlene 416 624-6526

\*For Sale: CBM 8032, 8050 and 8023P. \$2,400. Call Chris 782-8402 evenings.

\*Centronics P1 Thermo printer, manual, 20 rolls paper, \$120.00 or best. Call 247-7714.

\*Sell: 24K board with 3 expansion slots for VIC 20 \$55.00, Super expander cartridge \$25.00, Gorf game cartridge \$8.00, VIC 20 programmers' ref. guide \$5.00. All cartridges are like new, they have not been used very much. Michael Ulik, 211 1/2 S. Minn. Ave., St. Peter, MN 56082 USA

# Strengthen your hand

# with Superbase 64™

The complete information control system for the Commodore 64.

The World Famous Commodore 64.



No matter what your business or interest, with Superbase 64 you have a totally flexible 'record' system, as big as you want it, as fast as you need it.

Create your own formats, enter your records, change layouts and datafields.

Superbase gives you unrivalled control in home or office, business or professional practice, with a range of features including:

### YOUR OWN RECORDS

- Design your layout using text, numeric, linking and key fields.
- Record size up to 100 characters spread over up to 127 records.
- Number of records limited only by your equipment.
- As many databases as you want, each with up to 15 files.
- Learn fast through built-in HELP screens - then add your own notes.

### FAST ACCESS

- Keyed records for high speed retrieval - 3 secs to any file size.
- Select and search for any name, description, number, date, etc. in multiple combinations.
- Sort records into any order.
- Display selections or generate printed reports.
- Browse through records matching on any criteria.

### TOTAL CONTROL

- Links to other programs and EASY SCRIPT for personalised mailings, high-quality letters, quotes, tables, etc.
- Effective management of invoices, addresses, stock, membership, appointments - any and every kind of record.
- English like commands for easy conversational programming, plus built-in BASIC.

### DATABASE MANAGEMENT

- Easy to understand menus or alter length - no file rebuilding needed.
- Update files with automatic batch processing option.
- Calendar arithmetic for effective time management.
- Display quantities, values, totals, as you enter them.
- Formulae for on-screen result calculation.



Superbase 64™

Precision Software (USA), Inc.  
Suite 11D  
1675 York Avenue  
NEW YORK  
N.Y. 10128  
(212) 410 3418



# Developing a mind for the Future. RTC



**SUPER BASIC \$46.95** CDN.

- Gives you 3 different versions of Commodore Basic Programming Language Version 4 PLUS!  
 A Built in Machine Language Monitor!  
 Disk & File Maintenance Commands  
 Data Handling Commands  
 Graphics Plus Basic  
 Compatible with Commodore's "B" Series & Much Much More!

**RECREATE PROGRAM \$39.95** CDN.

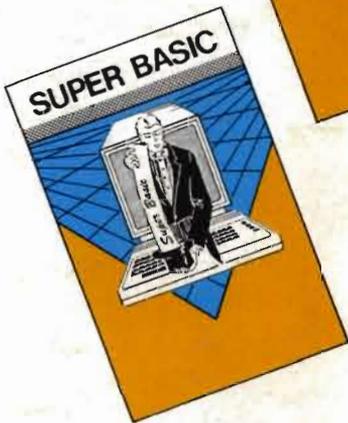
RTC's Answers to Program Recreation  
 Converts Printer's File to SCRIPT 64's Files

**SUPER COPY \$39.95** CDN.

Super Fast Disk Copies on a 1541  
 Copy Entire Disk in 7 Minutes or Less  
 Copy Selected Files  
 Complete Pattern Matching  
 Full Prompts

**BASIC AID \$49.95** CDN.

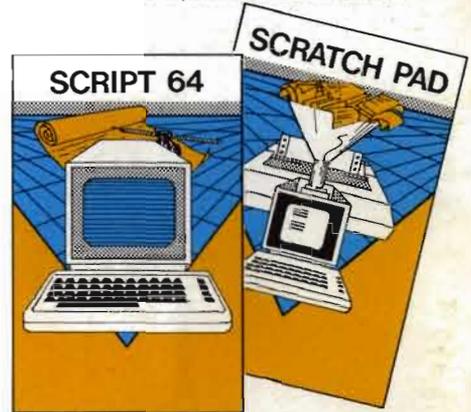
Your Aid to Writing Programs  
 Allows Scrolling Through Programs  
 Adds 33 more Commands to the Basic Language  
 Has Find, Change, Merge, Move Commands  
 Convert Hex, Binary and Decimal Numbers and More!



**NEW** Combined PACKAGE

**SCRIPT 64 & SCRATCH PAD 64 \$129.00** CDN.

Script 64.  
 Word Processor in French and English  
 80 Columns  
 Global Search and Replace  
 User Created Dictionary  
 Spelling Check  
 Scratch Pad 64  
 The Database/Mail List in One!  
 Merges with Script 64 Word Processor  
 Print out Labels, Envelopes, Mail List & More!  
 Suitable with both Single and Dual Disk Drives  
 Fully C64 Link Compatible



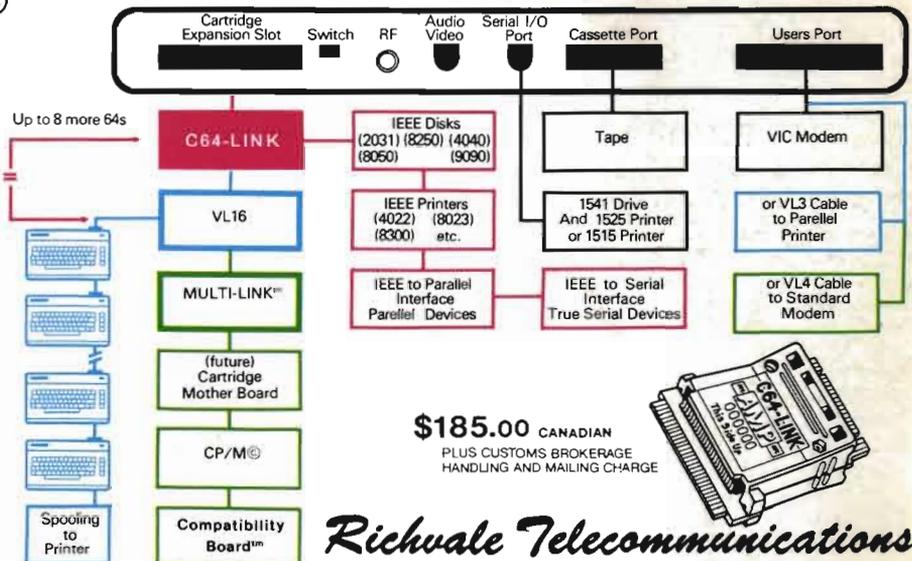
## C64 LINK <sup>®</sup> The Smart 64

Give These Expanded Capabilities to Your 64 and VIC 20

- The ability to transfer data from any type of device to another (IEEE, Serial, Parallel)
- BASIC 4.0 which allows you to run more PET BASIC programs and gives you extended disk and I/O commands.
- The ability to have several 64s on line together - sharing common IEEE devices such as disks or printers with Spooling Capability.
- Built-in machine language monitor.
- A built-in terminal or modem program which allows the system to communicate through a modem to many bulletin board systems and other computer mainframes.
- Compatibility with CP/M.

Contact your local Commodore dealer or RTC.

Payments by VISA, MASTERCARD or BANK TRANSFER.  
 Mail orders also by certified cheque, etc.



**\$185.00** CANADIAN  
 PLUS CUSTOMS BROKERAGE  
 HANDLING AND MAILING CHARGE



*Richvale Telecommunications*

10610 Bayview Avenue (Bayview Plaza) Richmond Hill, Ontario, Canada L4C 3N8 (416) 884-4165