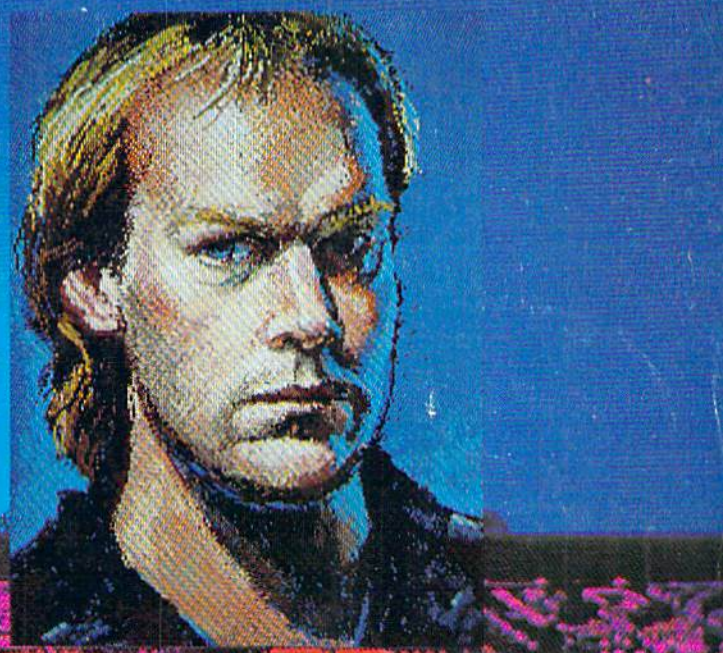
January/February 1986
$2.50 U.S.
$3.50 Canada ISSN 0744-8724

Buyers Guide: BASIC Compilers

commodore microcomputers

C-128 MEMORY MAPS



Software Reviews:
HACKER
PAPERBACK WRITER
PROJECT: SPACE STATION
MINDWHEEL
AND MORE

Amiga Update
HOT GRAPHICS
From Jack Haeger

Free Type-in Programs:
SCRAMBLER
CONTROL YOUR DATA
CONTROL Q

Control Your Home With a Commodore

0 43467 00122 01

YOU'VE ALWAYS HAD
NOW YOU CAN HAVE AN

Amiga's 4,096 colors give your business graphics a visible advantage.

Amiga's 4 channels of stereo give you a sound advantage.

Learning on Amiga is higher education.

A LOT OF COMPETITION. UNFAIR ADVANTAGE.

Amiga makes telecommunications fast, easy and colorful.

Nobody ever said it was going to be easy. But it just got easier. Now, there's Amiga.™ The first and only computer to give you a creative edge.

Amiga makes you look better, sound better, work faster and more productively. It can be your number cruncher, filing system, audio-visual department, graphic designer, print shop and faithful workhorse.

You can't buy a personal computer at any price that has all of Amiga's features. Nor can you find one that's easier to use. Amiga lets you point at symbols instead of learning complicated commands.

Amiga is friendly, but it's a powerhouse, too. It has twice the memory of Macintosh™ or IBM® PC. It costs less than either of them and can do everything they can do, better, because Amiga is more creative.

No other personal computer gives you over 4,000 colors, stereo sound and incredible dimension. Imagine the advantage of preparing business presentations with color graphics and sophisticated animation right on your computer.

Need to make creative use of your time? Amiga can do as many as four or five things at once in separate windows on the screen. Not just display them. Work on them. No other personal computer can.

Amiga will print the cover memo while you're working on a spreadsheet. And there's probably enough power left over to receive a phone message or a stock quote over a modem at the same time.

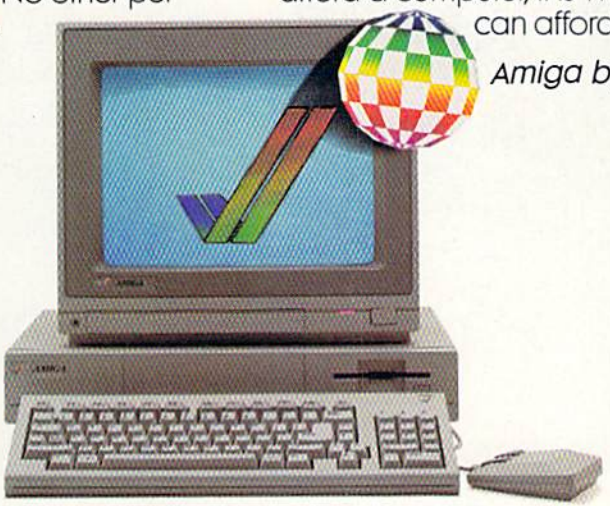
Amiga is IBM-compatible, too. A simple piece of software teaches Amiga to emulate the IBM operating system, so you can run most IBM programs. You'll have instant access to the largest library of business software in the world, including favorites like Lotus® 1,2,3 and dBase.®

And since Amiga is the last computer you'll want to buy, it was only fair to make it endlessly expandable and adaptable. You can plug in printers (almost any kind), joysticks, your video recorder, video camera, modems, musical keyboards, drawing pads, extra disk drives. You can even expand the memory to a whopping 8 megabytes.

Amiga will talk to you, read back what you write, answer your phone and compose music like a professional synthesizer. It can add new creativity to your life and bring new life to everything you create.

See an Authorized Amiga Dealer near you. Now that Amiga is here, the question isn't whether you can afford a computer, it is whether you can afford to wait.

Amiga by Commodore



AMIGA GIVES YOU A CREATIVE EDGE.

It's Absolutely Shocking!!!

After all these years, CMS Software Systems is still the only company providing professional quality accounting software for the complete line of Commodore business computers.

Whether you own an 8032, 8096, SuperPET, B-128, C-64, or the new C-128, we have a professionally written, fully integrated Accounting System designed especially for you.

Introduced in 1979, the **CMS Accounting System** was the first Accounting System available for Commodore computers. Not satisfied with just being first, we have continued to update, expand, and improve until today, the **CMS Accounting System** is widely recognized as one of the finest Accounting Systems available for any computer.

Now Available for the Commodore C-128

- General Ledger
- Accounts Receivable
- Billing
- Accounts Payable
- Job Costing
- Payroll

$179.95
Complete Price

For more information see your Commodore dealer or call Cathy York at 214/289-0677.

CMS Software Systems, Inc. • 2204 Camp David • Mesquite, TX 75149

Circle Reader Service No.7

VOLUME 7, NUMBER 1, ISSUE 39

CONTENTS

JANUARY/FEBRUARY 1986

commodore microcomputers

DEPARTMENTS

FEATURES

STAFF

PUBLISHER
ROBERT M. KENNEY
ASSISTANT TO THE PUBLISHER
MARY GRACE NICOLINE

EDITOR
DIANE LEBOLD
ASSISTANT EDITOR
CAROL MINTON
TECHNICAL EDITOR
JIM GRACELY
ASSISTANT TECHNICAL EDITOR
SUN WON KIM
CONTRIBUTING EDITOR
LOUIS F. SANDER

ART DIRECTOR
ROBERT C. ANDERSEN
ASSISTANT ART DIRECTOR
NANCY A. WALKER
COVER PHOTO
IMAGE BANK/PETE TURNER

ADVERTISING COORDINATOR
CHERYL GRIFFITH
CIRCULATION MANAGER
JOHN O'BRIEN
ADVERTISING REPRESENTATIVES

SOUTHEAST/WEST
WARREN LANGER, SPENCER O. SMITH
WARREN LANGER ASSOCIATES
9320 NW 2ND STREET
CORAL SPRINGS, FL 33065
ADVERTISING INQUIRIES ONLY
305/753-4124

MIDATLANTIC, MIDWEST AND NEW ENGLAND
PAMELA S. FEDOR, RANDY FEDOR
FEDOR & FEDOR
700 RIVER ROAD
FAIR HAVEN, NJ 07701
201/741-5784

Commodore Microcomputers, Volume 7, Number 1, Issue 39, January/February 1986. ISBN 0-88731-054-0.

Commodore Microcomputers (ISSN 0744-8724) is published bi-monthly by Contemporary Marketing, Inc., 1200 Wilson Drive, West Chester, PA 19380 U.S.A. U.S. subscriber rate is $15.00 per year. Canadian subscriber rate is $20.00 per year. Overseas subscriber rate is $25.00 per year. Questions concerning subscriptions should be directed to Contemporary Marketing Subscription Department, Box 651, Holmes, PA 19043, phone (800) 345-8112. In Pennsylvania (800) 662-2444.

Contemporary Marketing also publishes *Commodore Power/Play*.

Application to mail at Second Class postage rates is pending at West Chester, Pennsylvania 19380 and additional mailing offices. POSTMASTER send address changes to: Contemporary Marketing, Box 651, Holmes, PA 19043.

ABC Membership applied for.

LETTERS

Alpha-Sort

To the Editor:

How do I use the "Alpha-Sort" routine (listed in the May/June 1985 Letters) in my own programs?

Beverly Lorne
Poughkeepsie, New York

To use the "Alpha-Sort" routine, you must first initialize, define, and dimension the arrays you are going to use. The program is written to sort three arrays based on the information in the first array (arrays of title (A$), author (B$), and volume (C) are sorted by the title). By adding the following lines, the program will allow you to input the arrays directly, thereby making the routine an independent program (as long as you omit line 8070, also).

```
6900 INPUT "ENTER THE
   NUMBER OF ELEMENTS PER
   ARRAY";T
6910 DIM A$(T), B$(T), C(T)
6920 FOR X = 1 TO T:PRINT
   "A$(";X;")=";:INPUT A$(X)
6930 PRINT
   "B$(";X;")=";:INPUT B$(X)
6940 PRINT "C(";X;")=";:INPUT
   C(X)
6950 PRINT:NEXT X
```

If you want to use this routine to sort only one array rather than three, simply enter "0" for the elements of arrays B$ and C.

Programming Tip

To the Editor:

Teaching students to write structured BASIC programs would be much easier if the computer would allow multiple spaces for indentions after a line number, or line numbers followed by no text.

This *is* possible on the Commodore 64—the line number can be followed by [COMMODORE J]. The remainder of the line can be left blank or any number of spaces left before the first keyword for indentions. The graphic symbol which appears the first time this is done does not show up on LIST, which makes the program look quite nice. *But,* the extra spaces disappear if the line is edited after listing (unless COMMODORE J is inserted again).

Somehow, this simple procedure does not sink in on many students. However, an even simpler technique does: Following a line number with a colon (:) has exactly the same effect as [COMMODORE J]. The colon *does* show up when the program is listed, but editing the line *does not* eliminate the extra spaces.

Jack Ryan
El Dorado, Arkansas

Actually, the line may begin with just about any *shifted or Commodore-Key characters (including a shifted space) for the same results as a Commodore J.*

Amiga

To the Editor:

I am among a growing number of Amiga-philes who, out of common admiration for your revolutionary computer, have banded together to form local and regional users groups (Amare Amiga). We hope in this way to express our great interest in the Amiga as a tool for both creativity and productivity.

Members of our Amiga users groups (Amugs) are drawn from many different vocations (small businesses, health professionals and artists), but a majority are of students and professors from major state universities such as the University of Arizona, Arizona State, and University of Texas, El Paso. We applaud your daring innovation, and wish to become part of the bright Amiga future now.

As of September 3rd, local Amugs began bimonthly meetings at specified University locations. Regional meetings will be held every two months at the University of Arizona. Local and regional newsletters containing Amiga happenings, as well as hardware and software reviews, will be published monthly.

Ivan Lesnik, President
Amare Amiga
211 W. Roger #29
Tucson, AZ 85705
(602) 887-4117

Easy Script

To the Editor:

Your magazine has been of great interest and support to me ever since I bought my 64 over two years ago.

I read with interest the *User Hotline*

Continued on pg. 6

response to the question about special characters from *Easy Script* in the July/August 1985 issue. However, you failed to mention the special one-keystroke commands from the function mode (F1) that help make Commodore's product more powerful and versatile.

For instance, the escape character does not have to be included on the format line setting control parameters. It is simply accessed by F1 [up arrow]. This is covered in Section 8.2.11.3, pages 8–10 of the manual.

Likewise, most of the frequently used enhancements are accessed from the F1 mode through specified shifted or unshifted keystrokes. On my Epson RX-80, these include: [ ] = toggle expanded print on and off, ( ) = emphasized, &% = double strike, ' = one superscript character, , = one subscript character, ;: = underline, and 〈〉 = compressed. These symbols may be combined for multiple effects such as emphasized underlining. On my printer they do not need to be combined with the escape command. This is covered in Section 8.2.10, pages 8–9 of the manual.

Using these enhancements saves keystrokes and frees the formatted function lines for other controls such as italics, elite, and so on. They allow *Easy Script* to implement a truly impressive number of printer controls simultaneously.

Arnold J. Bradford
Falls Church, Virginia

Electronic Cottage Controversy

To the Editor:

After reading "The Electronic Cottage Controversy" (September/October 1985), I must confess with a sense of shame that I have, as a worker at a local factory and member of the AFL-CIO, been an inadvertent financial supporter of the anti-cottager effort. To compound this, I have found that, as a resident of the state of Illinois, I live in one of the AFL-CIO's target states.

I couldn't agree more with the author's viewpoint that the total ban on computer work at home proposed by the AFL-CIO is not acceptable. I would only add that it is no surprise to me that the union is not concerned with the individuals who would be adversely affected by the proposed ban. They are not being paid to be concerned about those people—yet.

I contacted the Association of Electronic Cottagers (AEC) at the phone number listed at the end of the article. The person with whom I spoke was extremely courteous, and expressed a desire for more people to get involved in the fight to keep anti-cottager legislation from passing. One way he suggested is through the AEC forum on CompuServe (GO HOME 146).

Since being laid off, I have been seeking work that will replace my factory job. I fully intend to become a gainfully self-employed cottager, and I consider it an insult to my intelligence when the AFL-CIO tries to tell me that I need to be protected from myself. Perhaps the group that the AFL-CIO needs to "protect" us from the most is themselves!

Continued on pg. 8

ONE GOOD THING LEADS *to* ANOTHER.

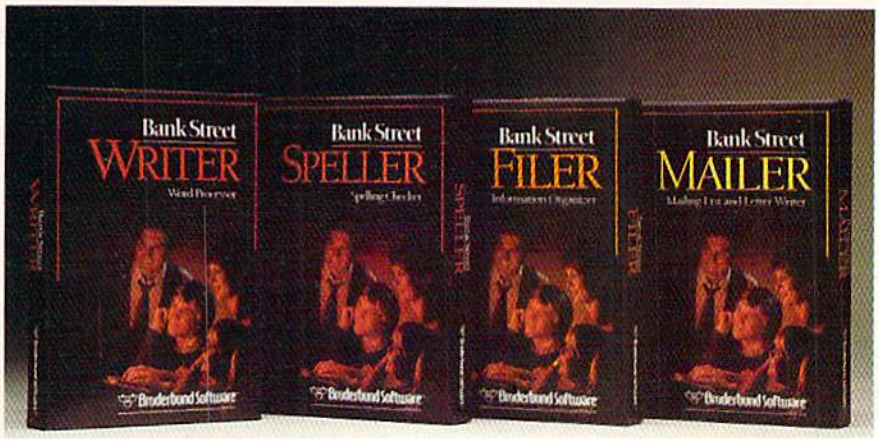
PRODUCTIVITY SOFTWARE ISN'T VERY PRODUCTIVE IF it's so complicated to learn, so complicated to use, that it never is used.

So when we sat down to design the Bank Street Writer,™ we kept one important objective in mind: to combine all the powerful features that people need in an affordably-priced word processor and make it so easy to use that just about anyone will be productive in moments.

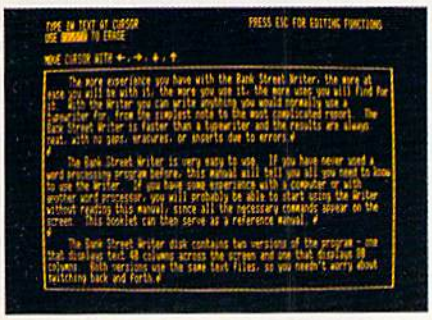
We must have succeeded. The Bank Street Writer is the number one word processing choice of over 300,000 users worldwide.

Now, that same philosophy can be found in a complete series of productivity software: the Bank Street Speller,™ Bank Street Filer™ and Bank Street Mailer.™ All the features you'll ever need. None of the complications you don't need. The perfect complements to the Bank Street Writer.

Bank Street products from Brøderbund give you more power for less money with less hassle. And when you think of it, that's the best kind of "productivity."

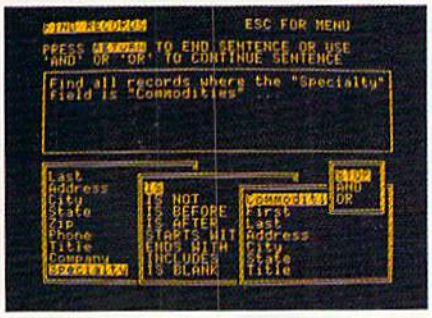


BANK STREET PRODUCTS FROM BRØDERBUND MAKE IT EASY TO MAKE YOUR COMPUTER WORK HARD.

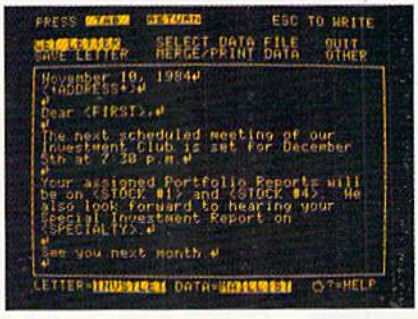


THE BANK STREET WRITER *lets you write letters, memos, articles or lengthy reports, better and faster. Continually enhanced and updated since its introduction three years ago, the Bank Street Writer is packed with features usually found only in far more expensive programs.*

THE BANK STREET SPELLER *finds and highlights spelling errors and suggests correct spellings; proofreads even the longest documents in seconds. The 31,000 word electronic dictionary can be amended with your own entries, including special terms, trademarks and proper names.*



THE BANK STREET FILER *helps you organize information and print out custom reports in moments. Collect, explore, organize and manipulate data in a variety of natural and flexible ways. For stamp collections or small business record keeping... for home financial and tax data... for bibliographies and reference files... for just about any kind of information you want to store and retrieve, the Bank Street Filer is as simple as using a file cabinet—only much faster, more convenient and more flexible.*



THE BANK STREET MAILER. *Whether you write occasional letters, produce a monthly newsletter or send out mailings to a long list of clients, the Bank Street Mailer does it quickly, easily and efficiently. You can insert names or addresses into a form letter, or send a personalized mailing to customers sorted by zip code, street address or any other aspect of your files. The Mailer can be used by itself (with its own built-in letter-writer) or with letters and lists from the Bank Street Writer and Filer.*

Brøderbund®

Versions of the BANK STREET WRITER are available for Apple, IBM-PC/PCjr, Commodore 64 and Atari personal computers. The BANK STREET SPELLER, BANK STREET FILER and BANK STREET MAILER are available for Apple and Commodore 64. For more information about Brøderbund products, please write us at 17 Paul Drive, San Rafael, California 94903-2101. Apple, IBM PCjr, Commodore and Atari are registered trademarks of Apple Computer, Inc., International Business Machines Corporation, Commodore Electronics, Ltd. and Atari Corporation respectively.

I strongly urge anyone who has any conviction regarding this issue to at least call or write to the AEC, and find out how they might be able to be a part of the stand against the proposed ban.

Name withheld on request

The AEC may be contacted at:
677 Canyon Crest Drive
Sierra Madre, CA 91204
818-355-0800
The AFL-CIO contact is:
Dennis Chamot, Associate Director
Department for Professional Employees
AFL-CIO
815 16th Street, NW
Washington, DC 20006
202-638-0320

Supermon

To the Editor:

Many an hour I have been spared from slaving over a hot computer by your Programmer's Tips articles. But some of these articles seem to make an easy task into an extremely difficult and confusing problem. One article in particular is "Getting Supermon to Print," featured in your September/October 1985 issue. I have found an infinitely simpler solution to the problem.

Line 130 of the BASIC part of SuperMon is as follows:

```
130 SYS(PEEK(43)+
  256*PEEK(44)+127)
```

To transfer output to the printer load SuperMon, but before you run it, type this (in direct mode):

```
OPEN4,4:CMD4:SYS(PEEK(43)+
  256*PEEK(44)+127)
```

When you press RETURN, SuperMon will run, and all output will be channelled to the printer.

B. J. Lindholm
Auburn, Alabama

Modem/300 File Translator

To the Editor:

Thanks for the article on using the Modem/300 file translator in the September/October 1985 issue. It supplied me with some information I have needed since I bought my Modem/300. However, I own a Plus/4 computer, in which the memory is arranged differently from the 64. So when I entered Mr. Nadler's updated file translator, all I could get when I tried to convert a SEQ file of a program was the first line of the program, followed by a "file not open error."

I soon figured out that one or more of the POKE statements in the program must be poking the value into the wrong part of Plus/4 memory. Comparing memory maps obtained from Jim Butterfield's book, *Machine Language for the Commodore 64 and Other Commodore Computers,* for the 64 and Plus/4 I found that all three POKEs needed correction. Lines 60600, and 61600 should be corrected to read:

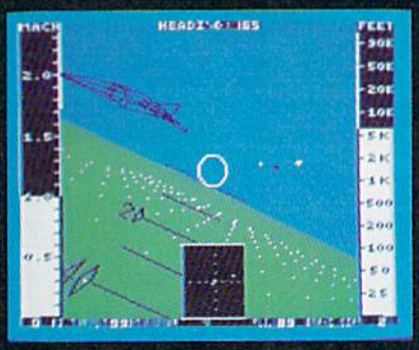
```
60600 POKE 151,1:PRINT
  "[CLEAR,DOWN2]";:
  S=6:GET#1,C$:
  C$=CHR$(ASC(C$+CHR$(0))
  AND 127)
61000 FOR I=1319 TO 1321:
  POKE I,13:NEXT: POKE
  239,3: PRINT "[HOME,DOWN4]
  GOTO60600[HOME]";:END
```

With these changes and a thorough reading of Mr. Nadler's article, any Plus/4 owner with a Modem/300 and Higgyterm software should be able to capture a program into the buffer. Save the buffer to disk as a SEQ file and finally use the file translator to convert the SEQ file to a PRG file.

Richard Rethorst
Kansas City, Kansas

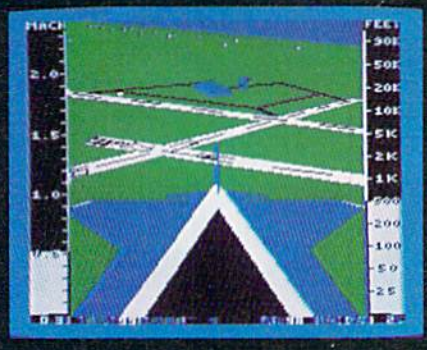
Commodore Microcomputers welcomes letters from readers. Do you have an unusual application for your Commodore computer? Do you wish to comment on an article? Would you like to make a suggestion on how we can better serve our readers? Please send them to:

Commodore Microcomputers
1200 Wilson Drive
West Chester, PA 19380
Attn: Letters



F-16 Dogfight with Enemy MiG-23 Fighters

F-18 on the Deck of a Nimitz-Class Aircraft Carrier (Control Tower View)



F-16 High-G Pullout over Detailed Wargame Scenery (Rear View)

Jet

From the author of Flight Simulator II comes a new dimension in realism. Jet simulates two fast and maneuverable supersonic jet fighters, a land-based F-16 or a carrier-based F-18. The simulator includes modern electronic flight instrumentation and the most advanced weaponry available. Jet's simulation sophistication, combined with excellent visual attitude references, makes it truly enjoyable to fly.

Easy aircraft control coupled with ballistic thrust gives you the kind of aerobatic maneuverability only a modern jet fighter can provide. Jet's attitude indicator is easy to read no matter what your orientation. A full-screen out-the-window view helps you get the most out of Jet's excellent flight controls. And that's a major consideration when flying at speeds in excess of 1300 MPH.

With Jet you can fly through either structured or non-structured environments. Engage in a deadly variety of combat missions. Explore the wargame territory, or relax by practicing precision aerobatic maneuvers. Load in scenery from optional United States Scenery Disks. You can even load in scenery off the Flight Simulator II disk.

New high-performance graphic drivers provide beautifully detailed scenery in either day or night-flight modes. You can look forward, left, right, rearward, or straight up out of the cockpit with a single keypress. The Jet simulator even includes a special view-magnification feature that lets you zoom-in to identify objects or details at a distance.

Jet will run on any Commodore 64 or Commodore 128 computer with one disk drive and either color or monochrome monitor.

See Your Dealer ...
or write or call for more information. For direct orders please enclose $39.95 plus $2.00 for shipping and specify UPS or first class mail delivery. Visa, MasterCard, American Express, and Diners Club cards accepted.

SCENERY DISKS FOR JET AND FLIGHT SIMULATOR II ARE NOW AVAILABLE.

subLOGIC Corporation
713 Edgebrook Drive
Champaign IL 61820
(217) 359-8482 Telex: 206995

Order Line: (800) 637-4983
(except in Illinois, Alaska, and Hawaii)

NEWS

Custom-Designed Space Shuttle

Design and launch a space shuttle in *Shuttle Designer*, a game for the Commodore 64 from Simpletec. Students do rocket theory and physics exercises to design the shuttle. If serious design flaws exist in the spacecraft, the program informs the student that NASA will not give them the contract. More subtle design errors may be caught by the "launch crew" during count-down. The problems can be rectified on the launch pad, or if undetected, may keep the shuttle from achieving orbit.

The design criteria of the shuttle adhere strictly to NASA parameters. The program has even passed a technical review by a panel of NASA engineers that included members of Werner von Braun's original rocket team from Peenemunde, Germany.

(Simpletec, 1852 Century Place, Suite 130, Atlanta, GA 30345)

Stereo Sound for the Amiga

Bose Corporation has introduced the RoomMate, a compact, lightweight speaker system for the Amiga with a built-in amplifier to provide high-fidelity sound with more signal and less noise.

There are two versions of the RoomMate. The original RoomMate retails for $229 and comes in charcoal gray. The Video RoomMate offers additional features over the basic unit, including volume control and shielded drivers to prevent television interference. Optional accessories for both systems include mounting arms, wall brackets, and a nylon travel bag.

(Bose Corporation, The Mountain, Framingham, MA 01701)

TV-Style Trivia

BrainBank, creator of *Murder by the Dozen* and its sequel *Felony!* (both published by CBS Software), has released *Millionware*, a TV-style quiz game for the Commodore 64 hosted by Bob BrainBank and his assistant, Donna Diskdrive. *Millionware* asks questions randomly selected from more than 40 categories (cartoons, science terms, Shakespeare, ethnic foods) and more than 1,200 possible questions.

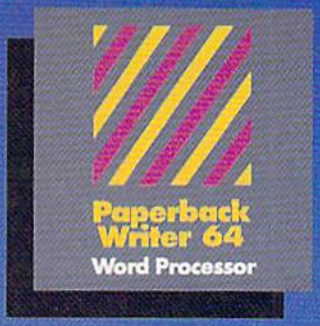
Players are given $10,000 each to start. Each wager is displayed along with the new balance for a right or wrong answer. Players increase their skill by competing against the clock and/or friends.

Millionware does not provide correct answers, so if a wrong answer is input, the player who loses the wager must look up the answer if he or she wants to know it for the next time.

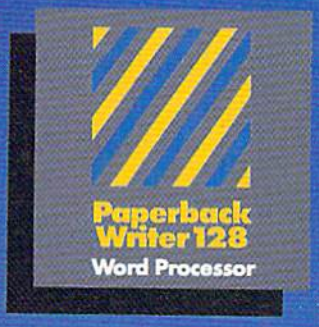
The program retails for $34.95.

(BrainBank, 220 Fifth Avenue, New York, NY 10001)

Solutions!



PW 128/64 Dictionary
also available at $14.95 (U.S.)



MAIL ORDERS:

CRYSTAL COMPUTER INC.
In Michigan 1-517-224-7667
Outside Michigan 1-800-245-7316
24 hours a day, 7 days a week.

U.S. DEALER ENQUIRIES:

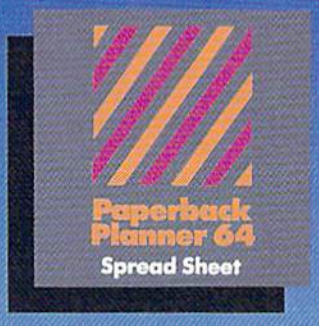
ALLEGRO TECH
1-800-544-1004
In Massachusetts
1-413-443-9443

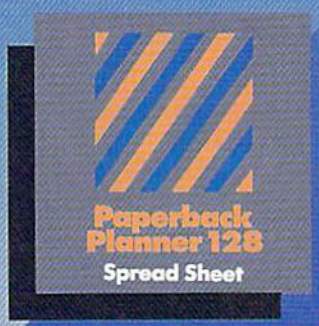
MICRO-PACE COMPUTERS INC.
1-800-362-9653
In Illinois
1-217-356-1884

MICRO-SYS DISTRIBUTORS
1-800-527-1738
In Texas
1-214-231-2645

CANADIAN DEALER ENQUIRIES:

FRANKLIN WINTER
Consulting & Sales
1-416-822-5974





Best-selling software for Your Commodore 128 or 64

You want the very best software you can find for your Commodore 128 or 64, right?

You want integrated software — word processing, database and spreadsheet applications — at a sensible price. But, you also want top-of-the-line features. Well, our Paperback 128/64 software goes one better.

With Paperback 128 or 64, you'll find all the features you can imagine . . . and then some. And Paperback 128/64 is so easy to use, you won't even need the reference guide. On-screen and in memory instructions will have you up and running in less than 30 minutes, even if you've never used a computer before.

The price? It's as low as you'd expect for a line of software called 'Paperback'. Suggested Retail Price for the 64 Software is $39.95 (U.S.) and $49.95 (U.S.) for the 128. Any of the 64 products may be upgraded to their 128 version for $15.00 (U.S.) + $3.00 shipping and handling. (Available to registered owners from Digital Solutions Inc. only.)

Paperback Writer 128 or 64, Paperback Planner 128 or 64 and Paperback Filer 128 or 64 . . . **Solutions** at sensible prices from Digital Solutions Inc.

International & Distributor enquiries to:

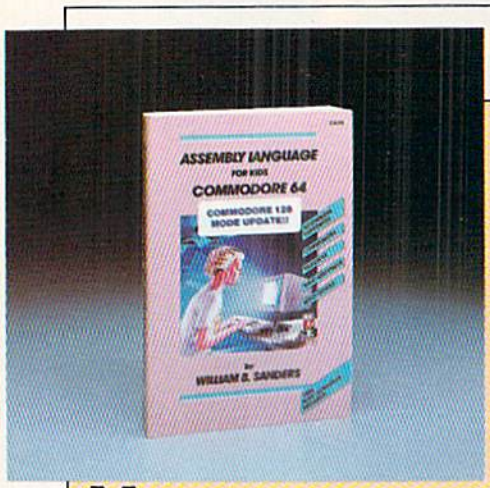
Serious software that's simple to use.

Digital Solutions Inc.

P.O. Box 345, Station A
Willowdale, Ontario
Canada M2N 5S9
1-416-221-3225

Paperback Writer 128 and 64 are now available in French.

NEWS



Assembly Language for Beginners

Microcomscribe has released an updated version of the book *Assembly Language for Kids: Commodore 64*, which now includes a *Commodore 128 Update*. The update contains reconfigured programs for programming in the 128 mode using the built-in mini-assembler. In addition, a 128 memory map, new BASIC 7.0 tokens, a 128 Sprite Assembler with the new addresses for sprite storage, information on switching memory banks, and instructions in using the monitor and mini-assembler have been included. The additional 29 pages, along with updated errata, have been added at no additional cost. The book retails for $14.95.

The book now includes instructions for using the major Commodore 64 assemblers and the 128 mini-assembler.

(Microcomscribe, 8982 Stimson Court, San Diego, CA 92129)

Monitor Arm

LinTek has introduced the Space Saver CRT Arm, a desk-mounted unit that holds the monitor off the desk. The Space Saver features steel construction and a heavy-duty mounting clamp, a 360-degree swivel base, and a CRT platform that swivels and tilts. The arm holds the monitor eight inches off the work surface. The Space Saver retails for $89.95.

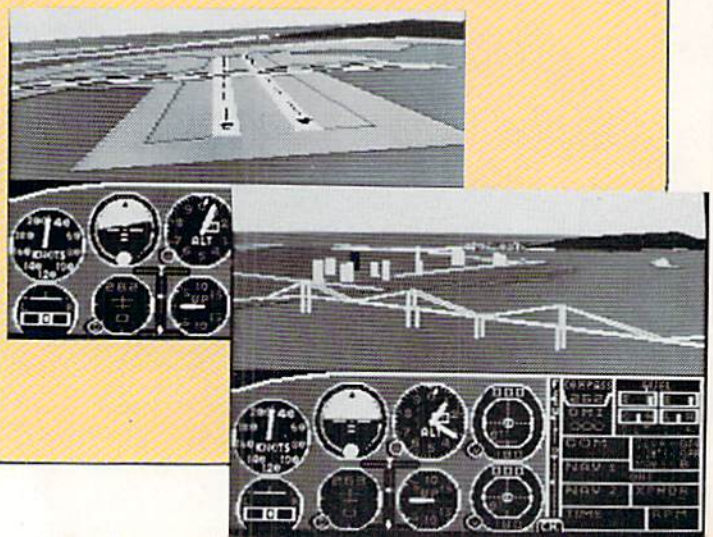
(LinTek, P.O. Box 8056, Grand Rapids, MI 49508)

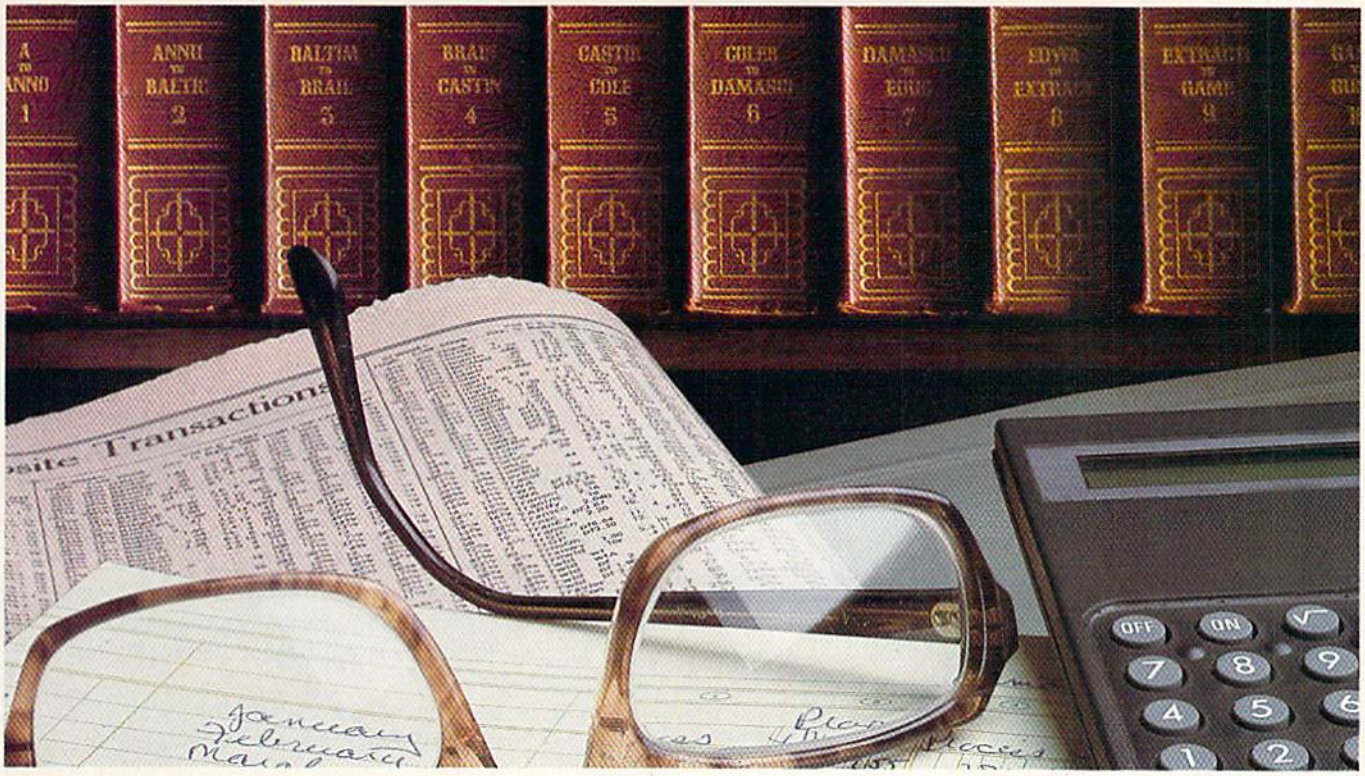
Broaden Your Horizons

SubLogic has released six scenery disks for the Commodore 64 that expand the flying environments of their *Flight Simulator II* and *Jet* flight simulators. The disks cover the entire western half of the United States, including the major airports, radio-navigation aids, cities, highways, rivers, and lakes located in each of six regions. Enough detail is included on each disk for either visual or instrument cross-country navigation.

Each scenery disk package comes with appropriate sectional charts plus full airport and navigational-aid directories. Individual scenery disk packages are available for $19.95 each. The Western U.S. six-disk set, packaged in a vinyl three-ring notebook with dividers, may be purchased for $99.95.

(SubLogic, 713 Edgebrook Drive, Champaign, IL 61820)





WE'VE TAKEN CARE OF BUSINESS SO YOU CAN TOO!

"If the 128 mode software appears on schedule, Commodore should have a winner on its hands."

—Personal Computing, July 1985

The challenge was issued when the Commodore 128 was conceived. Could an already impressive array of software for the 64 mode be complimented with solid 128 support? We always maintained it could. And now it's evident we've succeeded!

It's evident because a host of software, created expressly for the 128 mode, is now on dealer shelves. Software that incorporates the convenience of 80 column display. And uses every ounce of the 128K of available storage space to provide you with an invaluable combination for business and productivity needs.

The list is not unlimited—yet. But we have assembled an impressive lineup with still more on the way. Products from **Timeworks, Batteries Inc., Precision Software** and more. With such offerings from Commodore as Micro Illustrator 128, A Complete Course in BASIC Programming, and JANE.

We're pleased to announce we've taken care of business. The business of providing you with complete software support for your Commodore 128.

COMMODORE 128
A Higher Intelligence

NEWS

Sales Software

64 Sales Manager, a program for the Commodore 64 by Superior Micro Systems, maintains information about customers, prospective customers, scheduled appointments, and business expenses. It will generate reports for a customer master list, scheduled appointments, multiple business expense reports, mailing labels, and blank expense reports. The programs are menu-driven with on-line printable instructions. A disk utilities section is also included.

64 Sales Manager retails for $49.95 plus $3 for postage and handling.

(Superior Micro Systems, P.O. Box 713, Wheeling, IL 60090)

Real Estate Game

Sage Software has released *Shark*, a real estate trading game for the Commodore 64 which can be played by two to six players. In *Shark*, the object is to buy and develop properties, acquire monopolies, and charge opposing players rent in order to bankrupt them and force them out of the game. The game retails for $15 and is available from Sage Software, P.O. Box 2578-F, Freeport, TX 77541-2578.

West Coast Commodore Convention

The West Coast Commodore Association is holding its second annual convention February 8–9 at the Cathedral Hill Hotel in San Francisco. The show will run from 10 a.m. to 6 p.m. both days.

New hardware and software for Commodore computers will be featured, and Commodore's new Commodore 128 and Amiga computers will be shown. In addition, noted Commodore experts will speak on graphics, telecommunications, music, business applications and other subjects of interest to Commodore users.

For information, contact the West Coast Commodore Association, P.O. Box 210638, San Francisco, CA 94121, 415-982-1040.

Math Honors

Resource Software International has released the Honors Series—four programs to help students master advanced mathematics. The programs, *Honors Calculus*, *Honors Trigonometry*, *Honors Geometry*, and *Honors Algebra*, are in CP/M format to run on the Commodore 128 with a 1571 disk drive.

The Honors Series uses drill and practice to reinforce what students have learned in school. The programs are menu-driven and include a "help" feature. The program will compile performance reports during quizzes and drills.

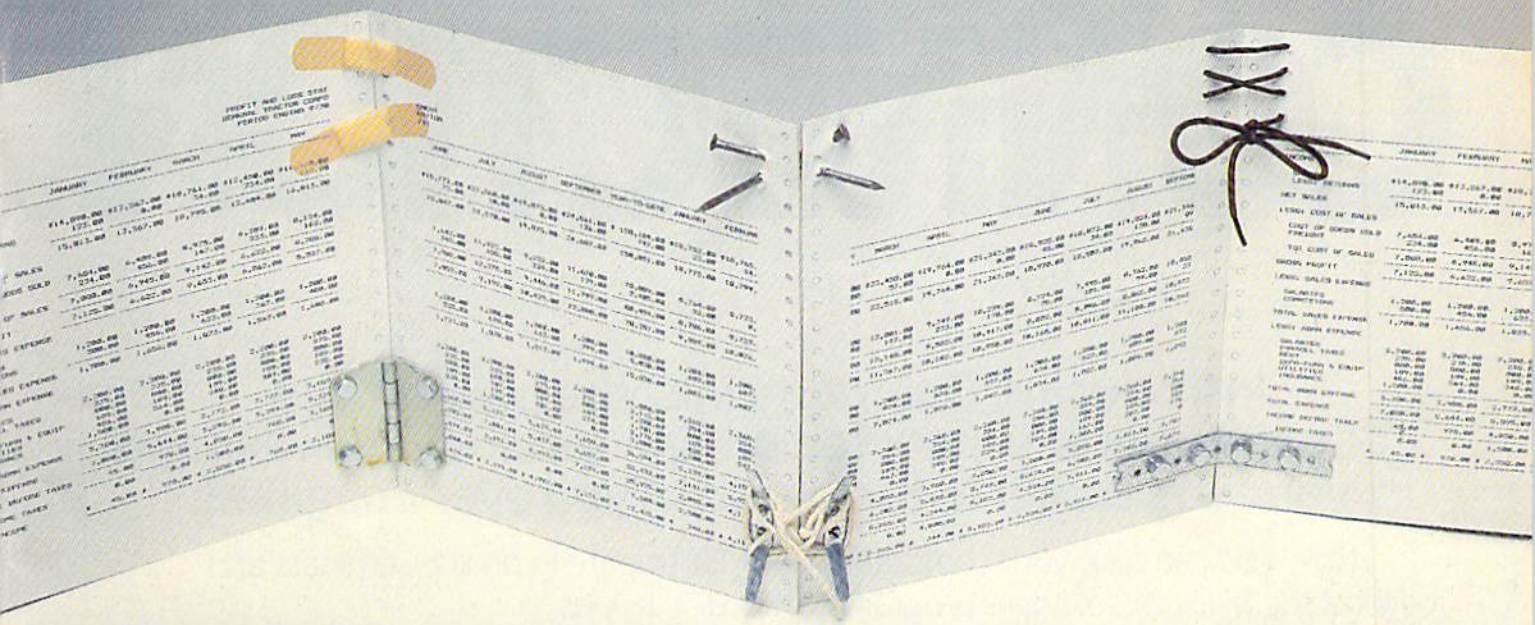
(Resource Software International, 330 New Brunswick Avenue, Fords, NJ 08863)

Talk It Over with the Computer

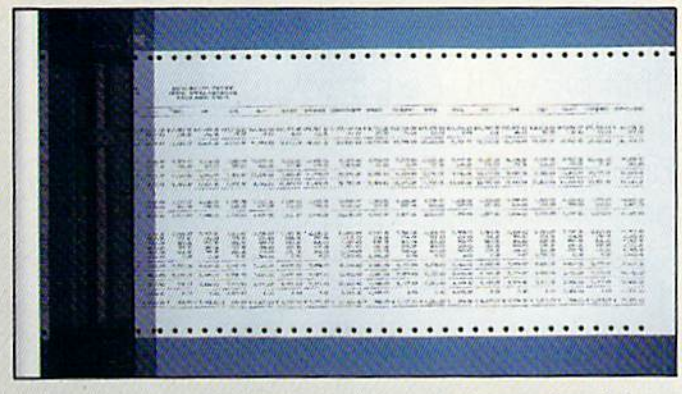
Personal Growth Technologies has released the *Software Listener*, a program that turns the Commodore 64 into a sounding board and sympathetic ear. The *Software Listener* requires no special training or computer expertise. It is designed so anyone who can use a keyboard can run the program and have conversations right away. The *Software Listener* is designed to pay attention, call the user by name, ask probing questions, offer helpful suggestions, and give encouragement. Integrated into this program is an Affirmations Write and Display feature designed to help develop a more positive self-image.

The *Software Listener* was written by a psychotherapist and educator. While it cannot replace a therapist, it does provide thought-provoking entertainment. It retails for $119.95. (Personal Growth Technologies, Box 1884, Boston, MA 02105)

SIDEWAYS... A NEW PROGRAM THAT SOLVES AN OLD PROBLEM.



Sideways. It prints your spreadsheet sideways.



The problem with spreadsheets is they get printed the wrong way. You still have lots of stapling and taping to look forward to before your printout is ready. Now, with SIDEWAYS, you can print a spreadsheet report that's wider than your printer paper – vertically, all at one time, on one continuous page.

SIDEWAYS rotates your spreadsheet 90 degrees as it prints out, causing your hard copy to print sideways. Nothing you create with today's most popular spreadsheet programs* is too wide for SIDEWAYS.

And, because you're no longer confined to the width of your printer paper, you have complete control over line spacing; left, top and bottom margins; character spacing; and you can choose from a wide variety of type sizes.

Now, get rid of that stapler and tape for good – go SIDEWAYS. Available now at your favorite dealer for only $29.95.**

***COMPATIBILITY:** Sideways works with any C-64 or C-128 spreadsheet program that can create text file information (ASCII) on a disk, or interfaces with a word processor. SIDEWAYS also works with these spreadsheet programs: Better Working Spreadsheet, Calc Now, Cal-Kit, Creative Calc, Multiplan, Practicalc, Syncalc, and Trio. Timeworks's SWIFTCALC already includes SIDEWAYS.

For Commodore 64 and 128 Computers.***

TIMEWORKS®

More power for your dollar.

**Manu. Sug. Retail Price. ***Reg. TM of Commodore Computer Systems

THE COMMODORE CONNECTION.

Introducing QuantumLink,™ the only official Commodore®-supported on-line service. It will expand your 64's universe further than you ever imagined.

Now you can take your computer beyond the limits of software programs. The new QuantumLink™ on-line service has been specially designed to turn your Commodore 64® or 128™ into a powerful "tele-computer." All you need is a disk drive and a modem to access hundreds of useful features.

Here are just a few of the services available.

EXCITING SOFTWARE

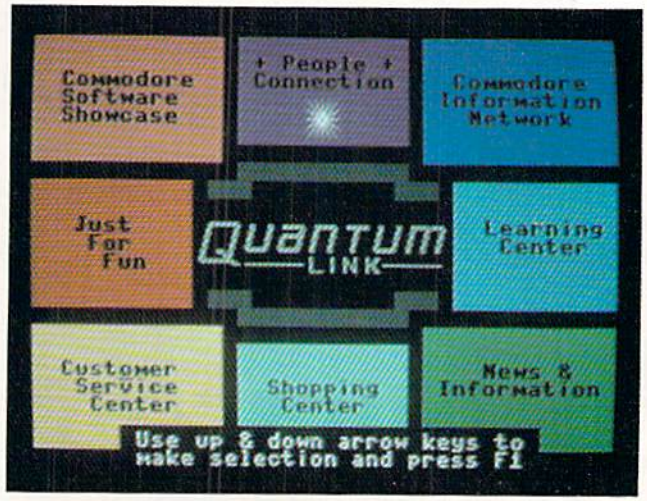
- Preview top new software before you buy it.
- Get hundreds of public domain programs.
- Read accurate, insightful software reviews.

HOT COMPUTER INFORMATION

- Exchange messages on bulletin boards.
- Get fast answers from Commodore computer experts.
- Read exclusive Commodore reports.
- "Ask Commodore" and get a quick reply.
- Participate in on-line seminars and discussions.
- Join Commodore's nationwide user group.

MAKE NEW FRIENDS

- Chat live with users across the country.
- Send and receive electronic mail.
- Find partners and play chess, hangman and other games *in full color.*



QuantumLink's full color, easy-to-use menu.

STAY INFORMED

- Access Grolier's Academic American Encyclopedia.™
- Read USA Today™ Evening Update.
- Get the scoop from RockNet™ news, Hollywood Hotline,™ Soap Opera Update.

FROM HERE TO INFINITY FOR ONLY $9.95 A MONTH.

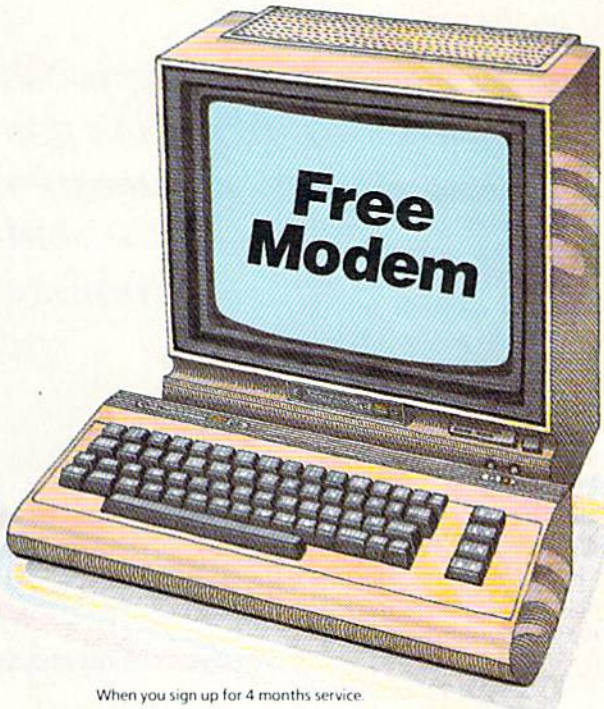
New QuantumLink is the lowest priced full service network. For only $9.95 a month you can expand your Commodore's universe with a whole range of QuantumLink services. And, you can access many features including the encyclopedia, software catalog, USA Today, entertainment news and trivia quizzes as often as you like, for no extra charge. Special "Plus" services cost only 6¢ per minute. *But, your first hour of "Plus" time every month is free.* (No surcharges for communications or 1200 baud access, either!)

If you have a modem:

When you sign up for an additional month.

Now, getting on-line with QuantumLink is easier than you ever imagined. If you have a modem, we'll send you the QuantumLink telecommunications software and when you sign up for a month, we will give you the second month of service *free.*

If you don't have a modem:



When you sign up for 4 months service.

If you don't have a modem, we'll send you a Commodore 300 baud modem and the QuantumLink software *free* when you sign up for four months.

QuantumLink™
The Commodore® Connection

Name_____

Address_____

City_____ State_____

Zip_____ Eve. Phone_____

☐ **I NEED A MODEM.** Please send my free Commodore® 300 baud modem, and the QuantumLink software. Charge my credit card $39.80, to cover the QuantumLink $9.95 monthly fee for the first four months.*

Credit Card: ☐ MasterCard ☐ Visa

Acct. No._____ Exp. Date_____

Signature_____

☐ **I HAVE A MODEM.** Please send me the free QuantumLink telecommunications software. I understand I will be charged the first month's fee of $9.95, and will get the second month free.*

2 WAYS TO ORDER:

BY PHONE:
If you have a modem, you can register on-line for fastest delivery! Hook up and call **1-800-833-9400!** If you need a modem, call **1-800-392-8200,** and ask for Terry.

BY MAIL:
Mail card or this coupon to Quantum Computer Services, Inc., 8620 Westwood Center Drive, Vienna, Virginia 22180.

*If you use more than the included 1 hour of Plus services each month, you will be billed for the extra time. Offer valid in Continental U.S. for new subscribers only. Expires 6/30/86.

REVIEWED BY CHRISTINE ADAMEC

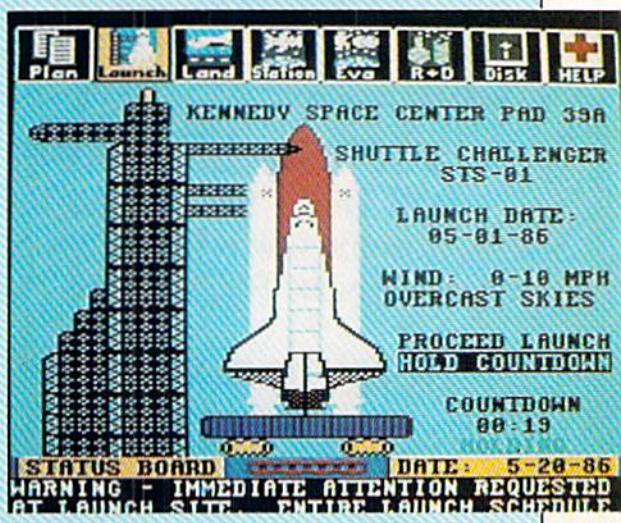
Project: Space Station

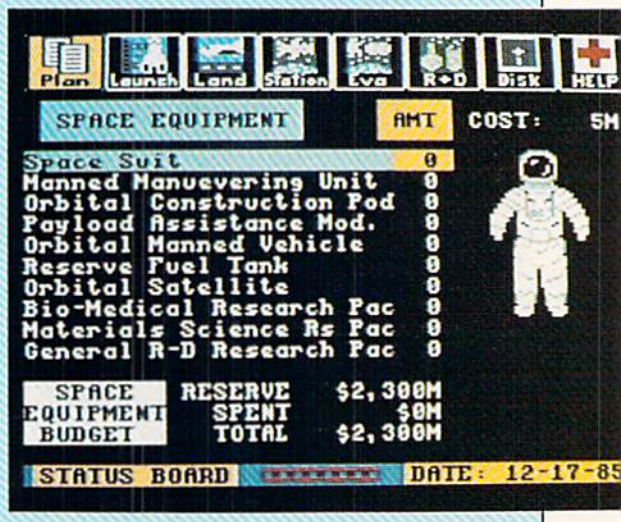
Computer: Commodore 64
Publisher: HesWare
390 Swift Ave. #14
San Francisco, CA 94080
Medium: Disk
Price: $24.95

You haven't been personally handpicked to launch the next Challenger or Atlantis at Kennedy Space Center? Not to worry, because now you get an insider's view without stepping out of your home. *Project: Space Station* is a multi-faceted menu-driven space simulation in which you are the mission coordinator.

As mission coordinator, you must make up your own budget and select the crew, equipment and modules. You will launch your shuttle and choose the type of research and development projects to pursue while in space. If successful, you'll accomplish one of NASA's greatest goals: to make the shuttle a commercial success.

If successful, you'll accomplish one of NASA's greatest goals: to make the shuttle a commercial success.





The program's main menu is comprised of eight icons, from planning, launching, landing and stationing the shuttle to extra-vehicular activities. Using your joystick, start with planning and move on to budgeting, crew selection, station design, and launch scheduling.

Budgeting is critical. You initially have ten billion dollars to play with, and you must determine how much to spend on planning, modules, and equipment. Next, move along to another important factor in the success or failure of your mission: selecting crew members.

You'll need at least six crew members, including a shuttle pilot. You must match their expertise and personalities to the type of mission you select, and choose people who'll work well together. The computer provides the instructions on-screen, and you can also refer to the excellent 125-page manual. Whether you begin by referring to your manual or relying on the computer, you'll glean information on each of the 32 potential crew members, including their technical background and personality.

For example, meteorologist Mac Stevens has a Ph.D. in atmospheric science and is a staff meteorologist at the National Weather Bureau. He is a "highly dependable individual..Not very remarkable. Extremely competent, but unimaginative . . . ." I didn't choose Stevens, but instead went with meteorologist Joe Church, who is "highly charismatic, garrulous . . . exceedingly ambitious . . . ." He has a few negative qualities, but sounds adventurous enough for my mission.

The program also provides you with the last book read by each astronaut, an evaluation by his or her associates, a favorite quote, and other information. For example, I loved the quote on flight engineer F. Frucci: "It's five o'clock. Time to party." Keep in mind that each crew member costs you money—a shuttle pilot, for instance, costs $85K, the highest wages.

Of course you'll also need equipment and modules for your mission, such as a docking module and a lab module. Keep your eyes on the money. Each piece of equipment or module costs you, and if you run out of bucks, it's all over.

The graphics of the equipment and modules are very realistic, and the graphics throughout the program are outstanding. At any time you can request a complete on-screen description of each module and its function, enabling you to decide whether it's really vital to your mission.

You then put your space station together, using the joystick to dock the modules. The manual provides easy-to-follow instructions.

Before launching, you need to make

Continued on pg. 126

Number one arcade hits for your computer.

KUNG-FU MASTER™

You are the KUNG FU MASTER. Battle the evil forces through the five dangerous floors in the wizard's castle to rescue the captive fair maiden.

KARATE CHAMP™

Put on your black belt and challenge your friends or the computer through nine picturesque settings in this leading Martial Arts Game to become the KARATE CHAMP.

Both available on diskette for the Commodore 64™ and the 48K Apple II® Series. Look for them at your local computer/software or game store today!

Data East, a leading manufacturer of video arcade games for over ten years including hits such as Burger Time*, and Commando*.

Commodore 64 and Apple are trademarks of Commodore Electronics, LTD and Apple Computer, Inc. respectively.

* These trademarks are owned by Data East USA, Inc.
** Elrem Corp. manufactured under license by Data East USA, Inc.

DATA EAST

470 Needles Drive
San Jose, CA 95112
(408) 286-7074

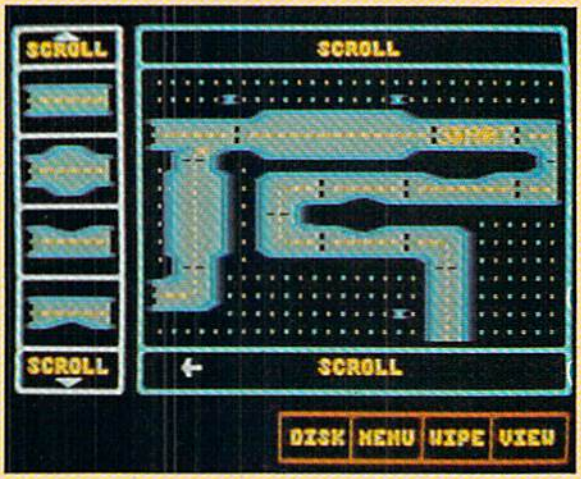
REVIEWED BY GARY FIELDS

Fast Tracks

The Computer Slot Car Construction Kit

Computer: Commodore 64
Publisher: Activision
P.O. Box 7287
Mountain View, CA 94039
Medium: Disk
Price: $34.95

Because there is no limit to the number of track pieces you can use, it is easy to design slot-car courses which would cost hundreds of dollars to construct using real plastic.



Fast Tracks is both a track construction kit and a race game that lets you design a race track and then compete on it. The program is so well designed that all the features can be accessed by simply using the on-screen prompts. Most options are joystick activated, though you do have to use the keyboard when creating a game for a friend.

For those who want to race first and lay tracks later, Activision has included five super-challenging tracks with the game disk. Try "924 Turbo" if you want a challenge, or, for a twist, try racing on the freeway of "Fun City." *Fast Tracks* does have its own version of Wrong Way Corrigan, so watch out. Sometimes when you bump one of the cars off the track, it reenters the race running the wrong way, so defensive driving is a must.

The tracks you design can be saved to play again and again. The number of separate courses you can construct is limited only by your imagination and supply of diskettes. A single diskette will hold up to 20 tracks.

You can also create entire personalized *Fast Tracks* game disks, which will feature the tracks you construct, and credit you, in the title screen, as one of the developers. Activision encourages you to give these creations as gifts. The created game is complete in itself and will work without the original *Fast Tracks* software. This means that the friend who receives it does not have to own a copy of *Fast Tracks* to play your game.

During construction, the left edge of the screen displays boxes that contain pieces of track. To use a piece of track, just point at it and press the fire button. Then move the pointer where you want the piece to go and press the fire button again. All the track pieces you would expect in a good slot car race kit are included: curves, overpasses, underpasses, intersections, switchers, merging sections, a start line, and both wide and narrow straightaways. Some sections contain oil slicks to add a little danger.

The construction board is eight times larger than the monitor screen, so to view the entire board, you point at the word "VIEW" and presto, you get an overhead view of the entire race track. Actual racing is done on a semi-three dimensional display, which includes not only the race track and four cars, but shrubs and trees. The best sound effects are the screeching tires when rounding a curve and the crash sounds when cars collide or spin off the track.

The game has three races: one lap, five laps and ten laps. The best times are recorded for each, and high scores for the five-lap wins are recorded permanently on disk.

Just as a real slot car racer requires little control to round the track, *Fast Tracks'* cars also require little control. The only time a car spins off the track is when it either goes into a curve too fast or collides with another car.

I liked *Fast Tracks'* ease of use. Designing a track is simple and fast. Even young children should have no trouble laying out race courses.

What parents will like most is its cost and neatness. When the child wants to add to his or her race course, there is no added expense. Because the *Fast Tracks'* parts warehouse can never be emptied, the household budget need never suffer from a trip to the toy store to buy more track pieces. And parents who have ever walked through a bedroom filled with a slot car kit will love the idea of neatly storing the entire kit on a thin diskette.

I found designing complex tracks the most enjoyable and challenging feature of *Fast Tracks.* Because there is no limit to the number of track pieces you can use, it is easy to design courses which would cost hundreds of dollars to construct using real plastic. The inclusion of three-way and four-way intersections, overpasses and merging track means your creations can be elaborate. The construction mode screen editor makes changing, moving and adding pieces as easy as twisting your wrist.

I also really liked the create-your-own-game option. *Fast Tracks* is the first program I've seen that creates a complete game which can then be given as a gift. I'll use this option to save money on a few birthday gifts.

Fast Tracks should appeal to the age group between seven and fourteen, and is a good activity program which should supply hours of both entertainment and creativity. If you are a slot car lover, try this one. C

Introducing the Most Powerful Business Software Ever!

FOR YOUR IBM • APPLE • MAC • TRS-80 • KAYPRO • COMMODORE • MSDOS OR CP/M COMPUTER*

50% OFF!

The VERSABUSINESS™ Series

Each **VERSABUSINESS** module can be purchased and used independently, or can be linked in any combination to form a complete, coordinated business system.

VERSARECEIVABLES™ $99.95

VERSARECEIVABLES™ is a complete menu-driven accounts receivable, invoicing, and monthly statement-generating system. It keeps track of all information related to who owes you or your company money, and can provide automatic billing for past due accounts. VERSARECEIVABLES™ prints all necessary statements, invoices, and summary reports and can be linked with VERSALEDGER II™ and VERSAINVENTORY™.

VERSAPAYABLES™ $99.95

VERSAPAYABLES™ is designed to keep track of current and aged payables, keeping you in touch with all information regarding how much money your company owes, and to whom. VERSAPAYABLES™ maintains a complete record on each vendor, prints checks, check registers, vouchers, transaction reports, aged payables reports, vendor reports, and more. With VERSAPAYABLES™, you can even let your computer automatically select which vouchers are to be paid.

VERSAPAYROLL™ $99.95

VERSAPAYROLL™ is a powerful and sophisticated, but easy to use payroll system that keeps track of all government-required payroll information. Complete employee records are maintained, and all necessary payroll calculations are performed automatically, with totals displayed on screen for operator approval. A payroll can be run totally, automatically, or the operator can intervene to prevent a check from being printed, or to alter information on it. If desired, totals may be posted to the VERSALEDGER II™ system.

VERSAINVENTORY™ $99.95

VERSAINVENTORY™ is a complete inventory control system that gives you instant access to data on any item. VERSAINVENTORY™ keeps track of all information related to what items are in stock, out of stock, on backorder, etc., stores sales and pricing data, alerts you when an item falls below a preset reorder point, and allows you to enter and print invoices directly or to link with the VERSARECEIVABLES™ system. VERSAINVENTORY™ prints all needed inventory listings, reports of items below reorder point, inventory value reports, period and year-to-date sales reports, price lists, inventory checklists, etc.

VERSALEDGER II™ $149.95

VERSALEDGER II™ is a complete accounting system that grows as your business grows. VERSALEDGER II™ can be used as a simple personal checkbook register, expanded to a small business bookkeeping system or developed into a large corporate general ledger system **without any additional software.**

- VERSALEDGER II™ gives you almost unlimited storage capacity (300 to 10,000 entries per month, depending on the system),
- stores all check and general ledger information forever,
- prints tractor-feed checks,
- handles multiple checkbooks and general ledgers,
- prints 17 customized accounting reports including check registers, balance sheets, income statements, transaction reports, account listings, etc.

VERSALEDGER II™ comes with a professionally-written 160 page manual designed for first-time users. The VERSALEDGER II™ manual will help you become quickly familiar with VERSALEDGER II™, using complete sample data files supplied on diskette and more than 50 pages of sample printouts.

FOR COMMODORE OWNERS ONLY!

Commodore owners *only* may now take 50% off our listed price of any module(s) from our VersaBusiness Series. All sales are final (our normal 30-day money back guarantee does not apply to sale items).

To Order:
Write or call Toll-free (800) 431-2818
(N.Y.S. residents call 914-425-1535)

- add $3 for shipping in UPS areas
- add $4 for C.O.D. or non-UPS areas
- add $5 to CANADA or MEXICO
- add proper postage elsewhere

DEALER INQUIRIES WELCOME

All prices and specifications subject to change / Delivery subject to availability.

COMPUTRONICS INC

50 N. PASCACK ROAD, SPRING VALLEY, N.Y. 10977

REVIEWED BY DON VANDEVENTER

Supershipper

Computer: Commodore 64
Publisher: Progressive Peripherals & Software
2186 South Holly Suite 2
Denver, CO 80222
Medium: Disk
Price:

Supershipper is one of the most interesting programs I have seen. It provides the Commodore 64 owner with a powerful invoicing, billing, and shipping system. And the program is so easy to use that the documentation requires only 28 pages.

Who Can Use It

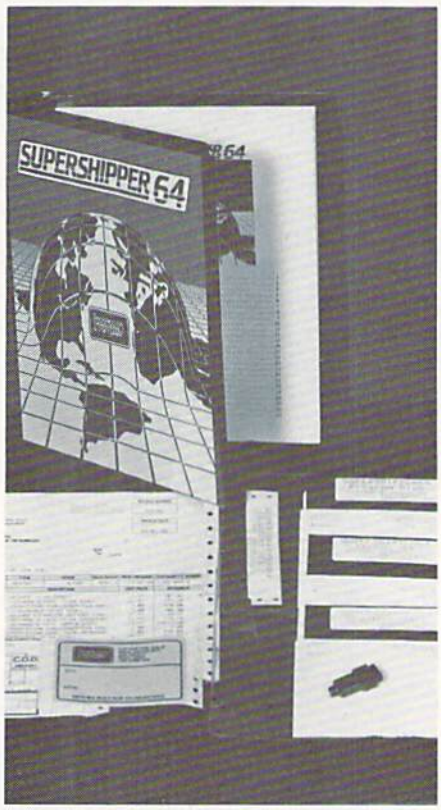
Businesses that involve mail-order or wholesale distribution will benefit the most from the program. However, almost any business that has a product or service to sell and which must keep track of regular customers and print invoices can use it.

One of the things that you must take into consideration with *Supershipper* is that the program requires two disk drives, or a dual drive such as the MSD or 4040. It also requires a printer capable of producing compressed print. (*NOTE*: Printers such as the 1525, 1526, 801, 802, 803 or 2023 will not print in the compressed mode.) The program may be used with up to four drives and/or four printers, making both disk and paper changes unnecessary.

Let's look at a wholesale book distributor as an example. Our sample firm provides books to a number of independent bookstores across the country. While they carry only 80 different book titles and deal with only 200 different stores, their invoicing and shipping takes up half the business day. As orders come in, the owner checks his inventory to be sure he can fill them. Next, he pulls the customer's address card and account history, types up an invoice, prints a mailing label, COD tag, and shipping statement. Then he updates his inventory, files the invoice, and enters the information into his client's account record. This requires a great deal of time.

Supershipper, however, can handle inventory, the customer's mailing information and credit history, invoice information, back-order product list, and shipping labels. It can even provide accounting information, which can be used with another of Progressive Peripherals' programs, the *Supershipper Accountant.*

With *Supershipper,* all the owner of our sample wholesale distributorship has to do is type in the customer's account number and order information. Then at the end of the day, he can print out all the invoices and shipping labels at one time. A job that required hours is now handled quickly on a Commodore 64.



Supershipper can handle inventory, the customer's mailing information and credit history, invoice information, back-order product list, and shipping labels.

The Program

Supershipper comes in a three-ring binder containing a program disk, a sample account disk, and a sample invoice disk. In addition, there are samples of the different reports, invoices, labels and shipping tags. And finally, there is a security key (dongle). Progressive Peripherals uses the dongle as a copy protection method, because only with the dongle plugged into joystick port two can you use the program. Since there is no protection on the disk itself, you can easily make backup copies of your program disk.

As I mentioned before, the program is designed to work with a minimum of two drives. However, this requires disk swaps, so for our sample business I used an MSD dual drive as device eight and a 1541 as device nine. Since the program requires numerous disk accesses to the program, my three-drive system worked much better than the system's minimum requirement of two drives.

In addition, I connected two printers: the first, a letter-quality Cardco printer (device four), and the second, a Star SG10 (device five) with an MW350 interface. This combination works very well, because you can change the printer device number with the interface dip switches. I used the letter-quality printer for the reports and invoices, and the dot matrix printer for printing product tags, shipping labels, and COD labels.

On the Account disk, *Supershipper* handles 800 accounts, each containing company, address, city, state, zip, phone, buyer, ship to address, UPS zone information, most recent invoice number, salesperson handling account, and two sort keys. In addition, it covers the account numbers and the default pricing and terms for each account.

The Invoice disk offers up to 500 invoices containing account number, purchase order number, date ordered, invoice data, shipping information, pricing, terms, credit information, and charge information. In addition, there is space for up to 200 products in inventory.

Another drive may be used for the Product disk. This disk handles up to

Continued on pg. 24

CAN WE TALK?

You Bet We Can!

Announcing The VOICE MESSENGER and EASY SPEECH For The Commodore 64 and 128.

YOU'RE NOT ALONE--ANYMORE! (Or how to program your best friend)
Ever ask yourself why you spend so much time alone. Or, if only someone would talk to me. Thanks to Welwyn/Currah, you won't have to think about those things again with the new, exciting VOICE MESSENGER™ speech synthesizer and accompanying EASY SPEECH™ text-to-speech system.

TALK TO ME
By simply plugging in the VOICE MESSENGER to your Commodore 64™ and 128™ home computer, it can really become your best friend. The VOICE MESSENGER and EASY SPEECH system will say anything you want, letter by letter, in words, sentences, or in conversational or story form.

SOUNDS GOOD
Unlike other types of synthesizers the VOICE MESSENGER and EASY SPEECH system has an allophone speech synthesizer that makes computer speech intelligible with an unlimited vocabulary that will speak to you clearly and understandably. Its text-to-speech converts your typed copy into conversational dialog at the touch of the keyboard.

NOT JUST ALL TALK
Welwyn/Currah along with Rist, Inc. are without a doubt the most innovative developers of computer speech technology. The VOICE MESSENGER and EASY SPEECH system was designed for all possible applications: educational, entertainment, business and home utility. And can immediately be put into use with most dedicated, menu-driven and pre-existing software including: Infocom's™ "Adventure Series", the CBS™ "Success with Math" educational series, and many, many more.

TALK IS CHEAP! (Inexpensive, anyway)
The VOICE MESSENGER and EASY SPEECH system is available at a suggested combination retail price of under $70.00, or may be purchased separately.

To find out how you can turn your Commodore 64 or 128 computer into your best friend, or to locate a store location nearest you call: (313) 547-8300; or write: Welwyn/Currah, 104 West Fourth Street, Suite 208-9, Royal Oak, Michigan 48067.

Welwyn CURRAH

an additional 2,000 products for a grand total of 2,200 products. Product information includes product name, number, code, retail price, wholesale price, distributor price, and the quantity in stock. This should be sufficient for even large distributors or mail-order firms.

One of the reasons *Supershipper* works so well is that it was originally designed by Progressive Peripherals to help keep track of their own products, customers and invoicing.

How It Is Used

After reading the manual (and I would recommend reading the entire 28 pages before you begin), you need to decide what disk drives and printers will be used, so you can configure your system. The system configuration is really very simple, requiring only about 15 minutes to complete.

Other information entered during the system configuration includes your company name and address, as well as information on how you normally ship, tax rates, and whether to charge extra on COD orders. Following this, you even have an option on password usage. Passwords are another example of how complete the program is. You can have system passwords for yourself and operator passwords for those who need only limited access.

After completing the system setup, you return to the main menu. Your next major task is to enter the product inventory. While the menus are all easy to use, they are one of my few complaints about the program. With every major menu change, there is another menu to be loaded, which can take as much as a minute or more. Using a dual drive with an IEEE interface helps considerably.

From the main menu, selection six will move you on to the Product/System Editor. Here you must make another menu selection to go to the product information area. The format is quite simple, yet it provides all the information needed, including retail price plus wholesale, distributor and other price information.

After entering the products your firm carries, the next step is to enter your customer accounts. This is done by returning to the main menu and going to the Account Editor. At the bottom of the screen are the key-one and key-two fields. Since the program will permit subsorts using either key field, these can be used to keep track of special information on your customers. For example, put accounts that have been slow on payment or paid with a bad check here. Then you can sort and print out a list of those customers, if needed.

When you have finished entering your major accounts, you will be able to see how *Supershipper* really shines at invoicing and labels. At the beginning of your business day, load the program and enter the Invoice Editor. As the day progresses and orders come in either by phone or mail, the information is entered directly into the system.

The first question the system asks is the account number. Enter a new account at this time or the number of an existing account. The system will then bring to the screen the default information that you entered into the customer's file. This information may be changed at this time for each invoice. After making changes, press the F1 key to take you to the second page of the invoice. Here you will type in the product number and quantity. The computer does the rest.

When you do your daily invoices, return to the main menu and then go on to the Invoice/Label Printer portion of the program. Again, since there is a menu change, this will take a couple of minutes. But for those printing ten or more invoices at a time, the wait is worthwhile. However, those of you who would like to use this as a "point of sale" program will find printing one invoice at a time extremely slow.

In printing your invoices, you have several choices as to the type of invoice forms to use. You can even create your own invoice on standard tractor-feed paper. *Supershipper* will also print your shipping labels and COD tags from this menu. Other features of *Supershipper* include printing back-order reports, inventory listings, accounts, and product labels. C

THE X-10 POWERHOUSE DOES EVERYTHING BUT PUT OUT THE CAT.

Model CP290

THE X-10 POWERHOUSE INTERFACES WITH YOUR COMMODORE TO CONTROL YOUR HOME...FOR SECURITY, COMFORT AND ENERGY SAVINGS.

This remarkable Interface lets you run your home through your Commodore 64 or 128 and a keyboard or joystick.

When you're away, it makes your home look and sound lived in. When you're home, it can turn off the TV at night and wake you up to stereo and fresh brewed coffee in the morning. It can even turn on your air conditioner and control your heating.

SPECIAL COLOR GRAPHICS MAKE PROGRAMMING A SNAP. You simply pick a room from the display screen. Use your keyboard or joystick to position graphics of lights or appliances. Then follow on-screen instructions to program any light or appliance to go on or off whenever you choose. You can even control thermostats, light intensity and more.

THE WAY IT WORKS. The X-10 Powerhouse Interface is cable-connected to the Commodore "User" port and plugged into a standard 110V outlet. After it is programmed, the Interface sends digitally encoded signals through your home wiring to special X-10 Modules. To control a lamp or appliance, you simply plug the electrical device into a Module

and then plug the Module into an outlet. The Interface can control up to 256 Modules throughout your home and won't interfere with normal use of lights and appliances.

There are plug-in Appliance Modules, Lamp Modules, Wall Switch Replacement Modules and Special 220V Modules for heavy duty appliances such as water heaters and room air conditioners. Plus Thermostat Controllers for central heating and air conditioning, Telephone Responders to control your home from any phone, and much more.

IT WON'T TIE UP YOUR COMPUTER. Use your computer only for programming. When you're finished, disconnect the Interface from the "User" or RS-232 port and keep it plugged into any convenient power outlet in your home. It will operate as a stand-alone controller with battery back-up and will run your home automatically.

SURPRISINGLY INEXPENSIVE. A Powerhouse System including the Interface, software and connecting cables costs less than $150. X-10 Modules are less than $20 each.

For the Dealer Nearest You Call: 1-800 526-0027
[in NJ: (201) 784-9700]
or, write to: X-10 (USA)
185A Legrand Avenue
Northvale, NJ 07647

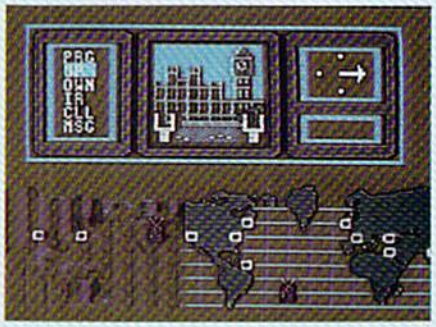
X-10® POWERHOUSE
NUMBER ONE IN HOME CONTROL

REVIEWED BY MARK COTONE

Hacker

Computer: Commodore 64/128
Publisher: Activision
2350 Bayshore Frontage Road
Mountain View, CA 94043
Medium: Disk
Price: $29.99

The mystery has a beginning and an ending, but how you get in between is up to you.



While toying with your computer, you accidentally stumble into a top-secret computer network. An operating screen greets you, bearing a single request: "Logon please." The cursor winks repeatedly, awaiting a reply.

At this point, you might refer to the manual. The first line of instructions requests you to "Insert Disk." The second line asks you to "Load." There is no third line.

What each player will quickly discover is that they are on their own: no clues, no rules, and no background information. A veiled adventure awaits without a hint of what, when, where or who is involved.

For players who enjoy the unknown, *Hacker* is a challenge with more surprises than Christmas morning. This mystery has a concrete beginning and a concrete ending, but how you get between these two points is entirely up to you. There are numerous paths, an array of dead ends, a host of characters, a trunkful of treasures, and at least a dozen ways to get caught. Just when you get comfortable with present operations, a sudden twist will send you scurrying.

But for the players who would rather be entertained, the game is designed so the play elements steer you in the right direction—even if approached incorrectly. There is immediate notification of judgment errors, and informative bulletins that help point the way.

A tough contest? Yes, but mainly due to its unconventional structure. There might be some initial confusion, but that will only breed creative deductions and gratifying results.

Now, I know what you're thinking. You see the review coming to a close, and you still don't know what this mission entails. And I'll keep it that way. The most unique aspect of this program is the thrill of discovery. Divulging any characters, theme or plot would ruin the fun.

If you think you might enjoy a mixture of text adventure, arcade action and world-wide mystery, *Hacker* is sure to satisfy. C

Hacker's Hindsight

Here are a few basic adventure guidelines that I think are worth repeating before you venture into *Hacker*.

- When asked for information, be specific. Capitalize and punctuate carefully. The computer system you have accessed has strict security codes.
- Write down any information you think you may need to recall at a later time—even if it's only given in bits and pieces. Names and dates may at first seem obscure, but may later be very important.
- If you can map an area, do so. The short time it takes to jot down directions can save retracing your steps, should a location be visited again. Much of this contest is based on the amount of time it takes a player to complete tasks, so knowing where you're going and how to get there can only help.
- If unsuccessful, remember the sequence of events that led to your downfall, and see if you can't discover a different angle. With no instructions or clues, there are times when you may not realize where you went wrong. C

Flight Simulator II Scenery Disks

The Challenge of Accomplished Flight

With a realism comparable to (and in some ways even surpassing) $100,000 aircraft flight simulators, Flight Simulator II includes full flight instrumentation and avionics, and provides a full-color out-the-window view. Instruments are arranged in the format standard to modern aircraft. All the radios needed for IFR flight are included. Front, rear, left, right, and diagonal views let you look in any direction. Program features are clearly documented in a 96-page Pilot's Operating Handbook.

For training in proper flight techniques, Flight Simulator II includes another 96-page instruction manual, compiled by two professional flight instructors with over 8,000 hours flight time and 12,000 hours of aviation teaching experience. You'll learn correct FAA-recommended flight procedures, from basic aircraft control through instrument approaches. To reward your accomplishments, the manual even includes a section on aerobatic maneuvers.

The Realism and Beauty of Flight

Go sight-seeing over detailed, realistic United States scenery. High-speed graphic drivers provide an animated out-the-window view in either day, dusk, or night flying modes.

Flight Simulator II features over 80 airports in four different scenery areas: New York, Chicago, Seattle, and Los Angeles. Six additional Scenery Disks covering the entire Western half of the United States are now available in IBM and C64/128 disk formats.

Apple and Atari versions will be released soon. Each disk covers a geographical region of the country in detail, and is very reasonably priced.

The Pure Fun of "World War I Ace"

When you think you're ready, you can test your flying skills with the "World War I Ace" aerial battle game. This game sends you on a bombing run over heavily-defended enemy territory. Six enemy fighters will attempt to engage you in combat as soon as war is declared. Your aircraft can carry five bombs, and your machine guns are loaded with 100 rounds of ammunition.

See Your Dealer. Flight Simulator II is available on disk for the Apple II, Atari XL/XE, and Commodore 64/128 computers for $49.95. Scenery Disks for the C64 and IBM PC (Jet or Microsoft Flight Simulator) are $19.95 each. A complete Western U.S. Scenery six-disk set is also available for $99.95. For additional product or ordering information, call (800) 637-4983.

subLOGIC Corporation
713 Edgebrook Drive
Champaign IL 61820
(217) 359-8482 Telex: 206995

Order Line: (800) 637-4983
(except in Illinois, Alaska, and Hawaii)

REVIEWED BY DAN RUSTIN

Paperback Writer

Computer: Commodore 64
Publisher: Digital Solutions
P.O. Box 345
Willowdale, Ontario
M2N 5S9 Canada
Medium: Disk
Price: $39.95

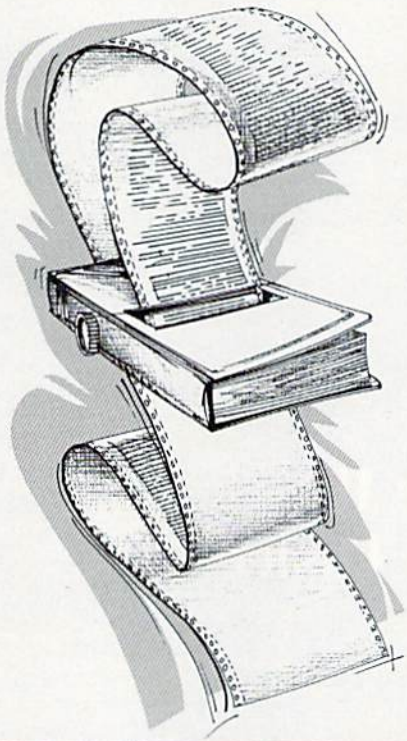
David Foster, the programming mind behind the extraordinary *Script 64* word processor (reviewed here April, 1985), has created another startling piece of writing software for the Commodore 64. His latest opus is *Paperback Writer,* and again it offers both 40- and 80-column options on the same disk without additional hardware.

But *Paperback Writer's* major contribution to the already cramped world of 64 word processing is that it puts on-screen exactly what you get on paper—italics, bold, underline, super- and subscripts—even page breaks. There is no "view to screen" option prior to print because you are always in this mode, even while you write or edit. You can, if you wish, create in 40-column mode, look over your work as it appears in 80 columns, and add hyphenation then and there, before printing. Very few programs offer this kind of flexibility.

Another major breakthrough of this software is that it affords remarkable legibility in 80-column mode, even—and especially—on a color monitor. In fact, the screen output in 80 columns is more readable on a Commodore 1702 color monitor than it is on a monochrome receiver. This is because the high-resolution on-screen characters are more delicate than the standard Commodore character set.

You cannot change screen colors with *Paperback Writer,* as you can with most other 64 word processors. The only time you will see another color is as a differentiation between some type fonts, such as when you enter a super- or subscript (yellow or green respectively), or prepare to move a block of italic text, which turns red for a few seconds. Otherwise, on-screen italics are displayed slanted, bold is bright white, and underlined text is just that.

Paperback Writer *could easily become the standard by which all word processors for the 64 are measured.*



Also, text format symbols do not appear on-screen—return arrows or any other. If you want to know where these flags are embedded in text, you use a two-key code to display them. In short, your working screen is an exact replica of your finished page.

Unlike *Script 64,* this program features complete wordwrap, and is not "screen based," which means it scrolls vertically. But, like Foster's previous program, his latest also contains an improvement over standard print-format command mode. Rather than obligate the user to initiate a series of codes for margins and spacing, *Paperback Writer* presents a pull-down menu of printing options. Roam through these options with the cursor keys. If you wish to read or change one, press ⟨RETURN⟩ and its default or current value appears in a window, where you can alter it.

You can't, however, make a permanent file of new default values, as you can with *Script 64.* This fine feature would have been easier to use if the values were posted on-screen, so you don't have to "capture" each one to read it.

When you load *Paperback Writer,* you're offered three choices: 40- or 80-column format, or spelling checker (user-created or 32,000 words available separately). After you've made your choice, the screen changes color a number of times. This may be a subtle suggestion, possibly, that you're not going to see much more color.

The title page appears next with printer options. You're offered a choice of the most common brands, and you can modify these to create a customized printer file. Some may require involved manipulation of the printer file to harness all their functions.

When the word processing program is lodged in memory, you see a blue "window shade" with standard command line, which informs you of cursor position and mode you're in (edit or other), plus a five-line help banner (which can be erased). If you need additional help, the program disk must be in the drive, always a nuisance when you're working with a single drive.

Help screens are called up by ⟨F7⟩, and are intended to be the main course of instruction with this program. As a result, the printed manual is succinct—but may be too barebones for some users. Nevertheless, the help screens should remedy this.

Most editing codes are accessed through ⟨CONTROL⟩ plus one other key, with the exception of the insert mode which is comfortably entered through ⟨F1⟩. Ranges are also easy to set with ⟨CONTROL⟩ "r," and blocks quickly moved, copied or deleted with other simple commands.

You can also alter the appearance of the cursor if you prefer flashing underline or solid block to the standard flashing brick. It moves by letter, word, screen or to the top or bottom

of the document. Eight tabs, included on screen, can also move the cursor quickly, or can be moved or removed.

Othe niceties include a numeric mode which aligns decimal points in a column, plus the ability to add or subtract columns of figures. Another of the many unusual functions of this program allows you to sort lists of names alphabetically, or rearrange a column of numbers automatically.

You can link files to print long documents, search and replace locally or globally (throughout a disk), alter print formats at any time, and merge with a mail list. Another unusual feature is the ability to alter the file type from text to sequential for storage of repetitive data such as mailing lists, or for telecommunications or transfer to other computers.

Print enhancements include headers and footers, justification, right alignment, centering, capitals, variable page length, and even the ability to include marginal notes in documents like contracts and scripts.

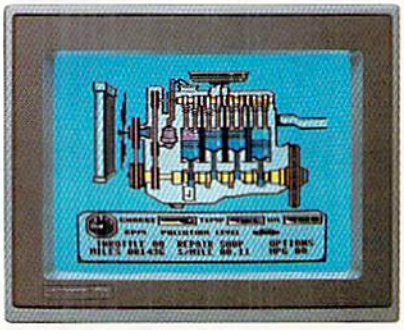
You can redefine keys as special or foreign characters (which appear as such on-screen). But you cannot easily create multi-use variables, such as designating a single key to represent a name throughout a long document (though you can, of course, use search-and-replace to do this). Nor is it possible to recover lost text in most instances.

When loaded, *Paperback Writer* offers 7,668 free bytes of memory in 80-column format (only about four double-spaced typewritten pages), or 16,116 bytes in 40-columns (eight pages). You can read available memory at any time. A global mode allows you to treat separately saved files as one for text uniformity and pagination. Saved files are recorded in the standard directory, and are loaded with the ⟨RETURN⟩ key.

There are a few small inconveniences. Text movement in 80-column mode is somewhat sluggish, and the constant reformatting of text with every line change may tend to restrain a burst of creative inspiration. All in all, however, *Paperback Writer* is a truly outstanding program. It could easily become the standard by which all word processors for the are measured. C

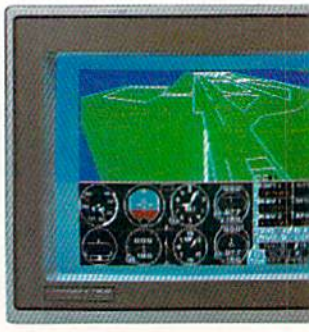
All you need to do this

graph a spreadsheet



fix an engine

write a novel



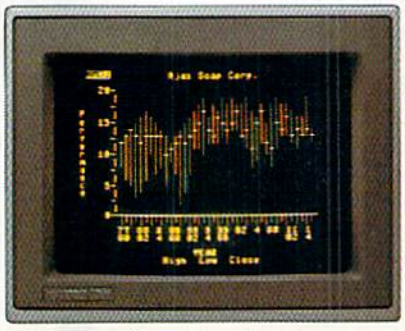
learn to fly

do your banking

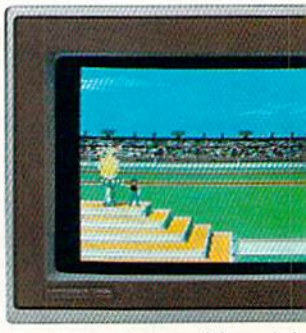
study astrology

organize a data base

tell a story



forecast sales



win a gold medal

a birthday card

compose a song

design a dream house

tra

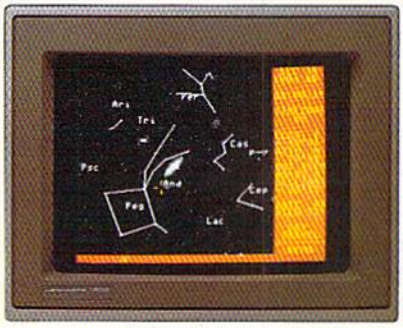
master logic

paint a picture

learn to type

dissect a frog

improve your chess



follow the stars

e

When it comes to personal computers, you want the smartest you can own. At a price that makes sense.

Feature for feature, the new Commodore 128™ system outsmarts its competition. It comes with a powerful 128K memory, expandable to 512K, more than enough to start with and stay with for years. An 80-column display lets you see more of your spreadsheets and word processing. And with its 64, 128 and CP/M® modes you have easy access to thousands of educational, business and home programs now available.

It also has a state-of-the-art keyboard with built-in numeric keypad to execute complex tasks with little effort.

The new Commodore 128. From the company that's sold more personal computers than IBM® or Apple.® Look for it at a store near you. And discover the personal computer that does more for you. At the price you've been waiting for.

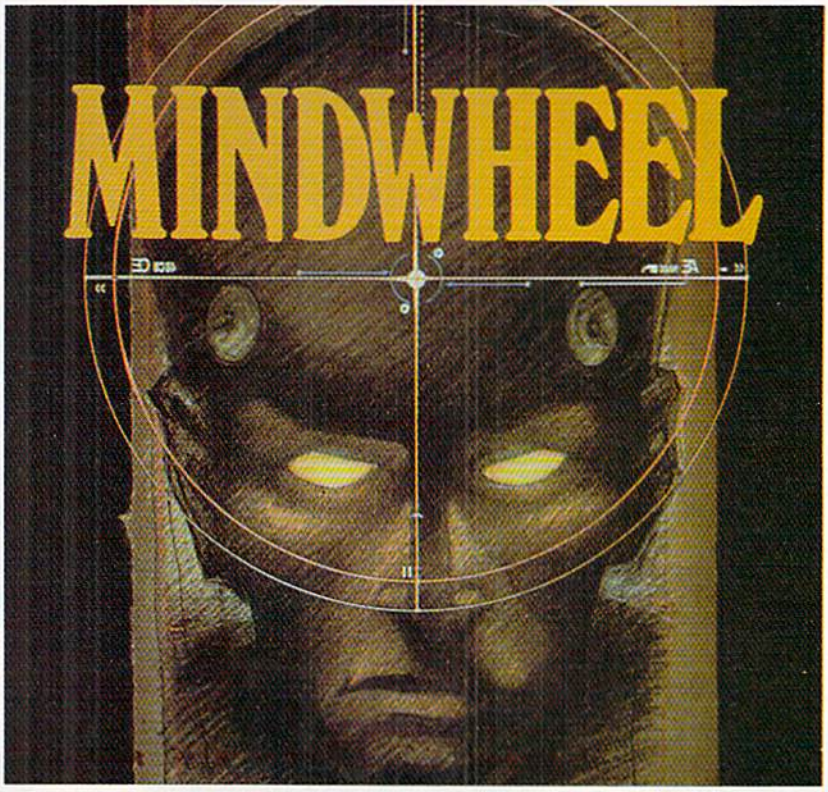
COMMODORE 128 PERSONAL COMPUTER

A Higher Intelligence

REVIEWED BY GARY FIELDS

Mindwheel

Computer: Commodore 64
Publisher: Synapse/Broderbund
17 Paul Drive
San Rafael, CA 94903
Medium: Disk
Price: $39.95



The world is about to self-destruct. Civilization is no longer civil. Mobs are taking control of the cities. Mankind has lost direction, and soon the human race will return to the mires from which it rose. *Unless* someone can telepathically journey back to the first moments of civilization and retrieve the Wheel Of Wisdom.

To perform this feat, you need the help of Doctor Virgil, who has experimented in "neuro-electronic matrix travel." This form of travel makes it possible for a living voyager to enter the minds of the dead. To succeed, the voyager must travel through the minds of a poet, dictator, rock star, and genius.

So prepare for an unusual adventure. *Mindwheel* spins its tale in real time. This means that just because you pause to think doesn't mean the program pauses. For instance, when you are voyaging through the mind of the rock star, you'll be accompanied by a bodyguard. As you sing on-stage, he will be fighting off thugs who are trying to get to you. This background action continues while you decide whether to sing another song or retreat off-stage. While a published book waits until the reader is ready to read more, *Mindwheel* keeps turning without you.

Talking to *Mindwheel* is easy because of Synapse's advanced parser. While some adventure games restrict the user to a few commands, *Mindwheel* allows the user much more freedom, which means you can use synonyms. In fact, *Mindwheel* will answer almost any command with a logical response. This greatly increases playability, since you can spend more time playing and less time checking the list of possible inputs.

I always test an adventure to see how it will respond to unusual input. So the first time I met the winged lady locked in a cage, I kissed her through the bars. I expected a response of "I DON'T KNOW THE WORD KISS." Instead, *Mindwheel* reported that the kiss had been warm and wet. It didn't free the lady, but it sure made me feel better.

Enter the minds of four extraordinary —but very dead— people, in a real-time adventure that reads like a good novel.

Mindwheel includes all the features you expect in a good adventure game. You can save eight different spots in the novel using the Bookmark command. The voyager can pick up and drop objects, examine his surroundings, check on his health and inventory his belongings. If you want a printed record of your progress, you can do it with the Printer On and Printer Off commands.

But *Mindwheel* is as much a novel as it is a game. It reads like a well written piece of literature. In fact, the two game disks are packed along with a short, hardbound book. Although reading the book is not essential, the information it contains will make a successful journey more probable.

The book also serves as a form of copy protection. To begin the game, you must type a word found on one of the pages of the book, and the word is rarely the same. The game disks are copyable. In fact, Synapse encourages users to make a backup copy of each disk, and includes a copy program just for that purpose.

If you've never read an electronic novel, this is a good one to start with. First time adventurerers should have no trouble conversing with *Mindwheel* because it will accept almost anything. If you're an experienced adventurer, you'll love *Mindwheel's* real-time action as well as the intricate storyline. Unlike other electronic novels, *Mindwheel* doesn't kill you every time you make a wrong move. I found this a happy switch.

This is a novel for the adult reader who enjoys solving riddles and digging through mysteries. If you enjoy strange and unexpected twists and turns, you'll love *Mindwheel*. C

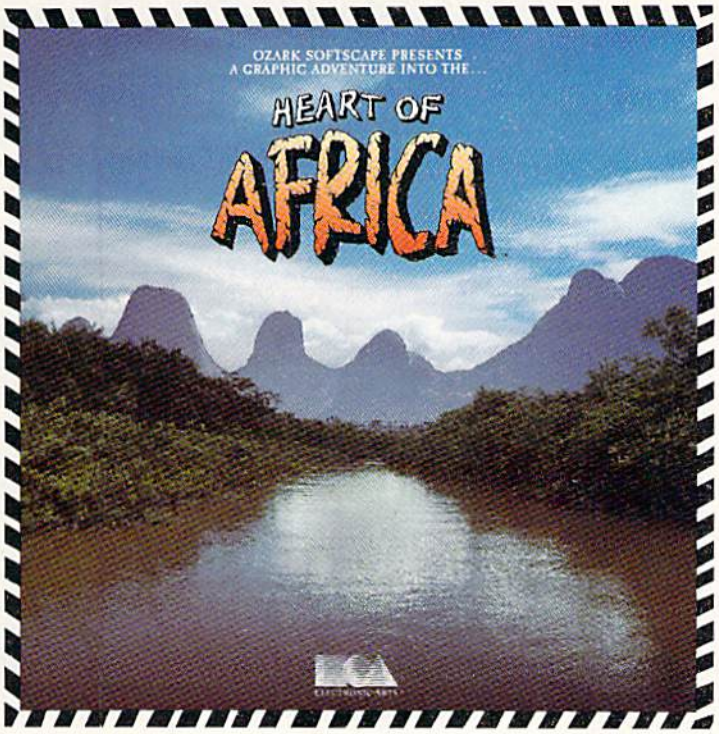
(Hi. I'm from Baltimore. Where's the pharaoh?)

A sequel to the Award-Winning Seven Cities of Gold.

Africa, 1890

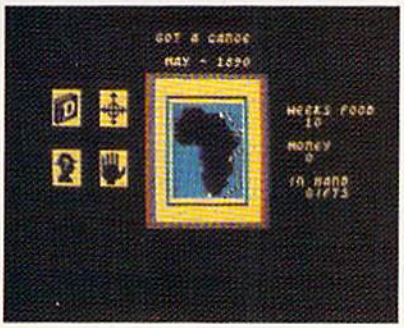
Up that river lies the African Adventure of your imagination.

The mysterious Dark Continent. Where the mighty Kilimanjaro towers over the Source of the Nile. Where the Congo snakes through steaming jungles and the scorching Sahara sun bleaches men's bones. Africa. Where legends of secret treasure drive men mad with desire. And where an adventurer can still find glory and gold...or a lonely death far from home.



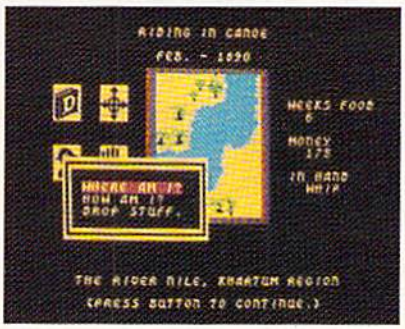
You have inherited a madman's diary ...and his dream to find the tomb of an ancient pharaoh.

Ancient songs still sung by tribal historians contain clues to the hidden tomb of a fabulously wealthy pharaoh. The madman's last feverish words drive you onward: "I now know it does exist. I am so close, but so near death. If you hear no more from me, then warn those that follow: this is not a journey for the weak of spirit or dull of mind. Come prepared."—H. Primm, May 21, 1889



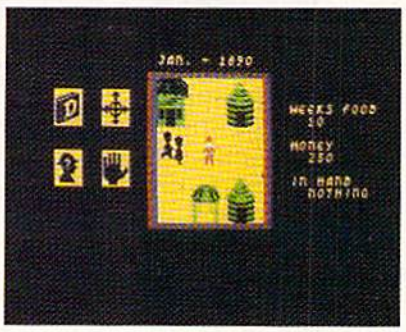
A Living Map of Africa

11 million square miles stuffed into your computer. Navigate 17 rivers. Hack through jungle and swamp. Sail to ten cities. Learn the ways of 22 tribes. Explore savannah, deserts and mountains. It's all geographically accurate.



Follow Your Dreams

Set your course and go for it. The computer automatically maps your progress and keeps your journal. If you get lost, it can tell you "The River Nile." What it won't say is "about to get sucked over the Kabelega Falls."



Decipher Native Clues

Local myths will help, if you can get the natives to talk. Enter their huts. Learn their customs. Pass out some bribes. The chief might tell you, "Look where the sun rises over the Childless Waters." Now try to figure it out.

ELECTRONIC ARTS™

Specifications: Available now for Commodore 64 & 128. Infinitely playable because the computer hides the tomb and treasure in new places each time you start. **How to order:** Visit your retailer or call 800-227-6703 for direct VISA or MasterCard orders (in CA call 800-632-7979). The direct price is $32.95. To buy by mail, send check or money order to Electronic Arts, P.O. Box 306, Half Moon Bay, CA 94019. Add $4 for shipping and handling. Allow 3-4 weeks for delivery. All E.A. products purchased direct have a 14-day "satisfaction or your money back" guarantee. For a complete catalog, send 50¢ and a self-addressed envelope to Electronic Arts, 2755 Campus Dr., San Mateo CA 94403.

REVIEWED BY WALT LOUNSBERY

Tale of Two C's

for the Commodore 64

The language C is nearly as motley as its origin. It was designed by Dennis Ritchie at Bell Laboratories in 1972. Based on the language BCPL by Martin Richards and B by Ken Thompson, C was intended for internal use at Bell. It is presently the language for the AT&T UNIX® operating system. Used in the job training programs of the massive Bell system, C became the language of choice for thousands of Bell programmers. Outside hobbyists soon caught on, and public domain versions proliferated.

Thus, C is now well documented and used on a wide variety of machines, from microcomputers to the Cray 2, the fastest computer in the world, while still retaining good portability and a hobbyist following. It is one of the few computer languages closely associated with a "style manual" (*The Elements of Programming Style* by Kernighan and Plauger).

Until AT&T recently began promoting the UNIX operating system, C received little fanfare. It is nevertheless one of the most widely used application languages next to assemblers. For example, *Multiplan* and *Lotus 1-2-3* are both coded in C, as are all of Digital Research's new products. The Commodore Amiga also uses this language in its operating system. For telecommunications, the language has the functionality of an assembler, even though it is technically a high-level language.

If you haven't been introduced to the language C, I recommend reading *The Master Handbook of High-Level Microcomputer Languages* by Charles F. Taylor. The authoritative reference is, of course, *The C Programming Language* by Brian W. Kernighan and Dennis M. Ritchie. There are many books out now that deal with the language, and a visit to your local bookstore should provide more than enough material.

Two very good C systems are now available for the Commodore 64, at reasonable prices. Let's take a look at them.

Super C's "loader" brings in a wedge and file copy utility, an editor, the compiler, the linker, or one of your C programs that is ready to run.

Super-C Compiler

The *Super-C Compiler* from Abacus is the better "starter" C package of the two, and the price is right. Abacus has done a good job with the manual, although other material may be required for someone without experience. *Super-C* comes on a single-sided disk, so it is easier to juggle disks or use two disk drives.

Super-C does not provide a true operating system for developing C programs. Rather, a simple "loader" is booted from the disk. The loader brings in a wedge and file copy utility, an editor, the compiler, the linker, or one of your C programs that is ready to run. This is menu-driven by the cursor, so you don't have to remember commands.

The wedge, called "C-Copy," allows file-by-file copying. The usual commands for reading the disk error channel and the disk directory are also provided. The wedge can handle disk drives with arbitrary unit numbers. This utility is mainly provided to copy certain files from the *Super-C* disk onto a work disk. Only three short files are needed on a work disk.

The "C-Editor" is a full-screen editor with unique color capabilities. Sections of your program can be highlighted with various colors. Line length is limited to 80 characters, however. Preset tabs are very useful for the indented coding style associated with C, and the tabs can be altered to suit your own style. The editor supports search-and-replace text functions, as well as an alternate text area. The text can be printed from the editor, which can adapt to a wide variety of printers. The only disadvantage is lack of support for disk unit numbers other than eight.

The C language makes heavy use of characters such as the bracket, tilde, vertical bar, and backslash, which are not on the Commodore 64 keyboard. Special keys and shift sequences provide these characters, which show up properly under the editor and when printed on certain printers. The editor can handle text files up to 41,000 bytes large.

The operation of the compiler (called "C-Compiler" of course) is very simple and straightforward. After the program is written under the editor and saved to the work disk, the compiler is loaded from the original *Super-C* program disk. You are then prompted for the name of your program text file (known as the "source" by programmers), and the name you want the intermediate compiled code file to have (this is known as the "object").

The compiler goes to work, and if any errors are present, it displays the errors and produces a file named "error-c," without creating an object file. The editor can then be loaded to correct the errors before trying again. The extra text area of the editor is very useful at this point, since the error messages can be put in the alternate area for reference while the pro-

A QUICK DATA™ DRIVE IS 20 TIMES FASTER THAN AUDIO CASSETTES, LOADS FASTER THAN A COMMODORE DISK DRIVE, AND COSTS $84.95.

Unbelievable? Believe it. The Quick Data Microwafer™ Drive is a **new drive technology** that loads a 24K program in 20 seconds vs. 8 minutes for a cassette drive or 1 minute for a Commodore Disk Drive.

Unbeatable price. One 170K drive: $84.95. Two drive 340K system: $154.95. Factory direct, with a full 30-day guarantee. If you're not delighted with your Quick Data Drives, return them within 30 days for a full refund. No questions asked.

Software included. All Quick Data Drives come with the QOS™ operating system, system utilities and easy-to-use instructions. Plus a blank Microwafer. You can be up and running in about five minutes.

Lots of software. Dozens of leading programs are available on Microwafers. Or, you can use Entrepo QOS to copy your existing Basic Programs to Microwafers from either cassette or diskette.

Mix-and-match. You can daisy-chain the Quick Data Drive with your existing cassette recorder by plugging it into the cassette port, or, use two QDD's linked together for dual drive capability. No problem.

Unconditional Guarantee: If you are not completely satisfied with your Quick Data Drive System, return it to Entrepo, Inc., within 30 days for a full refund. Your system is also covered by a 90 day warranty against failures or defects in manufacture.

Microwafer™ and Quick Data Drive™ are trademarks of Entrepo, Inc. Commodore® 64™ is a trademark of Commodore International, Inc.

MONEY-SAVING PACKAGES

THE MICROWAFER OFFICE: $139.95
Includes: 170K Quick Data Drive ($84.95); Word Manager word processor ($34.95); Plan Manager spread sheet ($34.95); the Speed File file manager ($69.95). A $189.85 value for $139.95.

THE MICROWAFER TEACHER: $129.95
Includes: 170K Quick Data Drive ($84.95); General Math; Spelling; plus Typing Tutor. A $159.80 value for $129.95.

THE MICROWAFER ENTERTAINER: $119.95
Includes: 170K Quick Data Drive ($84.95); plus, three of the most popular game programs for Commodore computers. A $174.80 value for $119.95.

THE PHANTOM QOS PACKAGE: $99.95
Includes: 170K Quick Data Drive ($84.95); Phantom QOS ROM Cartridge that quickly loads and locates the operating system in ROM outside the computer's main memory ($29.95). A $114.90 value for $99.95.

ORDERING DIRECT FROM ENTREPO.

Commodore® 64™ owners. Call us at the toll free number below. Give us your VISA or MasterCard Credit Card number, tell us which package you want. It will be on its way to you the following day. Simple as that. Allow $3.00 for shipping and postage. Really fast delivery? Add $10 for overnight package express.

TOLL FREE ORDERING: 1 (800) 231-0251
IN CALIFORNIA 1 (800) 648-4114

entrepo

gram text is in the primary area being corrected. The error messages, by the way, list the line number associated with each compiler error. This system helps speed development of a program in a manner friendly to a novice.

If a program is compiled successfully, its object code must be linked using the standard C function libraries in order to operate. This is done by the "C-Linker," which can handle up to seven files. Only two libraries are provided with *Super-C*, and whichever library is needed is totally linked with your object code. In other words, *Super-C* does not follow the usual course for C linkers in establishing a separate link file for extra functions. Rather, it has the equivalent of a run-time library that is attached to your program to provide any of the library functions.

This run-time library appears to add about 5,000 bytes to your program. This really shouldn't cause a problem under most circumstances, since about 53,000 bytes of memory are available for your linked program. The linker may also be directed to restrict memory use for storing machine language programs, or to produce a program that will run without the *Super-C* loader.

Variable types are always important, since they are dependent on hardware. The same computer can still have different definitions for the same variable type, though. In *Super-C*, short int and int are both two bytes, while long int is four bytes. Floating is four bytes (6 decimals), double is eight bytes (16 decimals).

Almost all the basic elements of C are available, except register storage and bit fields. The function libraries (run-time libraries) provided by *Super-C* are rather plain, providing no support for trigonometric functions, logs, and the like. This is not really unusual among microcomputer C compilers, though.

C Power Compiler

If there is one word that describes *C Power*, it is professional. I realize that this word is much maligned in advertisements, so I will explain. While computer professionals like as much capability as possible, often they must sacrifice to get it. In this case, a very fast C compiler with an extensive function library takes a lot of patience and skill to get it to work.

> C Power *achieves astounding speed, even faster than most compilers on the IBM PC.*

C Power, from Pro-Line, achieves astounding speed, even faster than most C compilers on the IBM PC. My only conclusion is that the compiler has an automatic sieve optimization mode built in. On the other hand, the system is provided on a "flippy" (a disk with files on both sides). The disk swapping on a single disk drive can get very hectic. And the documentation is almost as terse as the C language itself, but at this writing, the manual is being revised.

C Power operates as a "shell," which provides a simple environment for manipulating the disk drives and calling utilities and compilers. Separate work and system disk drives may be used and defined. Your own programs may be configured to run under the shell as utilities, so the shell can be extended as you please. The main C tools are two editors, the compiler, and the linker.

The editors are full-screen, with up to 240 characters per line. A text editor and a syntax-checking editor are provided. The usual search and move-text commands are provided, and special characters are supported. Unfortunately, the editors do not have tabs, so they require a bit more typing effort. Several buffers can be created under the editors. Unlike the *Super-C* editor, these editors do not overwrite files, so you must provide new names as you update and correct program source. A special command is provided under the syntax-checking editor to detect syntax errors.

Once your program is created, it is time to compile it. This compiler does not create error files, so you must read fast or use the syntax-checking editor heavily. My version of the manual neglected to mention that the compiler requires all included files on the work disk (for example, "stdio.h").

The linker is more of the same, requiring more disk swaps to get to the libraries. One of the nice features of the linker is the ability to produce an executable file (your final program) that will load and run from a given address. It is possible to direct the linker to create a program that runs under the shell. An arbitrary number of object files may be linked, but the error checking is poor.

C Power really comes into its own in program development. Besides the unique capability of assigning the address of executable programs, the interface of a C program to machine code is barely detailed in the manual. But Pro-Line does provide a good book on the C language, *C Primer Plus*, by Mitchell Waite, Stephen Prata, and Donald Martin, which explains this in greater detail. And the function library more than makes up for the brief manual. A few useful programs are included as examples, such as a sorter, a string finder, and a program to count word frequencies in text.

Perhaps of greatest interest to an applications programmer are the supported variables and program speed. *C Power* does not support a wide variety of variables. The integers short, int, and long are all two bytes, while both float and double are five bytes, with the same precision as the Commodore 64's native BASIC.

We are very fortunate that two fine C compiler systems are now available for the Commodore 64. The *Super-C Compiler* provides an ideal introduction to a very functional version of the C language, all at a reasonable cost. Those who want more power and don't mind the professional programmer's environment will consider *C Power* well worth the money.

Super-C Compiler
Abacus Software
P.O. Box 7211
Grand Rapids, MI
$79.95

C Power
Pro-Line Software
755 Queensway E Unit 8
Mississauga, Ontario
L4Y 4C5 Canada
$99.95

HOW TO GET OVER $2000 WORTH OF NEW CAPABILITIES FOR YOUR COMMODORE 64™

FOR $599*

The Spartan™ is the Apple™ II+ emulator for your Commodore 64™ that will open up a whole new world of hardware and software for you! Imagine adding these features to your Commodore 64™ for the Spartan™ price of $599: ☐ Apple™ II+ hardware and software capabilities ☐ 64K RAM expansion ☐ four software selectable Commodore 64™ cartridge slots ☐ non-dedicated 8-bit parallel port ☐ standard audio cassette deck capabilities for your C-64™. The suggested retail value of comparable products offering only these capabilities is over $2200.00* — but the Spartan™ gives you much, much more! By building on your investment in your Commodore 64™— an excellent introductory computer — you create a whole new system with both C-64™ and Apple™ II+ capabilities. There is a whole other world out there! The huge selection of Apple™ II+ hardware and software is now yours to explore! Call toll free for the Spartan™ dealer nearest you.

*All prices quoted are in U.S. funds, freight and taxes not included. Value of components equivalent to the Spartan™ system are quoted from Apple™ II+ CPU and Apple™ II+ single disk drive 1983 list prices, and from current suggested list prices and component specifications of other peripheral manufacturers. Commodore 64™ and Commodore logo are trademarks of Commodore Electronics Ltd. and/or Commodore Business Machines, Inc. Apple™ II+ is a trademark of Apple Computer Inc. Spartan™ is a trademark of Mimic Systems Inc. and has no association with Commodore Electronics or Apple Computer Inc. The Spartan™ is manufactured by Mimic Systems Inc. under license granted by ATG Electronics Inc. of Victoria, B.C. Canada.

MIMIC

FOR INFORMATION WRITE:
MIMIC SYSTEMS
1112 FORT ST., FL. 6F
VICTORIA, B.C.
CANADA V8V 4V2

TO ORDER CALL:
1-800-MODULAR
(663-8527)

BY ISAAC MALITZ, Ph.D

Impossible Tasks

One of the major concerns of computer science is to determine the overall capabilities and limitations of computers. An important question which arises in this context is: "Are there any tasks which computers are unable to perform?" If we could identify any such tasks, this might provide insight into the fundamental nature of computers. It also would be intrinsically interesting to see examples of tasks which are so difficult that they are beyond the capabilities of computers.

It turns out that, in fact, computers do suffer from certain fundamental limitations, which render them absolutely unable to perform certain kinds of tasks. In this article, we will look at several examples of tasks which are known to be impossible for computers. Following this, we will attempt to understand just what it is in the nature of computers which render them unable to perform these kinds of tasks.

The Endless-Printout Problem

The first impossible task is what we call the "Endless-Printout Problem." Here is an example of a BASIC program that generates an endless printout:

```
100 OPEN 1,4:CMD 1
200 PRINT "DUMB LOOP"
210 GOTO 200
```

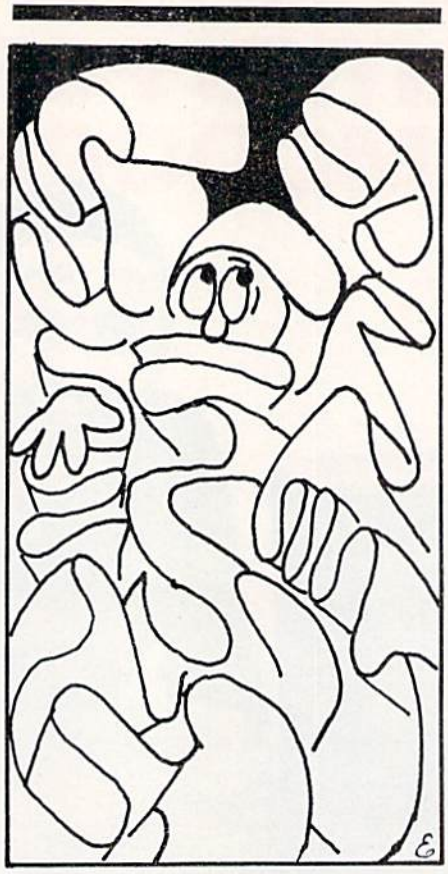
The first line in this program tells the computer to route all screen displays to the printer. The last two lines

```
200 PRINT "DUMB LOOP"
210 GOTO 200
```

are an "endless loop." They will cause the printer to print the words DUMB LOOP over and over again, forever.

It is easy to see why the above program would generate an endless printout. However, it is not always so easy to determine whether or not a given program would generate an endless printout. Consider the following program:

```
100 OPEN 1,4:CMD 1
200 X=X+7
210 PRINT X
220 IF X=22963 THEN END
300 GOTO 200
```



Is it possible to design an Ultimate Optimizing Program, *which would analyze any other program, and rewrite it in a fastest-possible version? The answer is no.*

This program consists of a loop which increases the variable X by 7, prints the value of X, and repeats this process over and over. There is one condition which can cause the loop to terminate, which is if X ever has the value 22963 (line 220).

Does this program generate an endless printout? It is clear that if X never has the value 22963, then the program generates an endless printout. But without analyzing the program carefully, it isn't clear whether X ever has this value. It turns out that X never has this value, so this program does, in fact, generate an endless printout. Nevertheless, it is not obvious from the program listing that this is so.

In very complex programs, it is increasingly difficult to determine whether they produce endless printouts. Here are two moderately complex examples:

```
100 OPEN 1,4:CMD 1
110 X=1
200 X=13*X + 12
210 X=X - INT(X/101)*101
220 PRINT X
230 IF X = 85 THEN END
300 GOTO 200
```

```
100 OPEN 1,4:CMD 1
110 X=1
200 X=13*X + 12
210 X=X - INT(X/101)*101
220 PRINT X
230 IF X = 15 THEN END
300 GOTO 200
```

These two programs are identical except for Line 230. One of them produces an endless printout, and the other does not. Can you determine which is which? It's the second one. The more complex the program, the harder we have to strain our intelligence to determine whether it produces an endless printout or not.

This leads us to an interesting possibility: Maybe we could program a computer to analyze given programs, and determine whether they generate endless printouts or not. It is clear that we could program a computer to identify some elementary kinds of endless-printout programs, such as our first example. And in fact, it's not too hard to develop a program which could deal with all of the examples presented so far. But *could we develop a program which could analyze any computer program—no matter how complex—and determine whether it would generate an endless printout or not?* That is the Endless-Printout Problem. The answer is no. It is impossible to program a computer to identify all possible endless-printout programs.

Why Computers Cannot Solve the Endless-Printout Problem

Why are computers unable to solve the Endless-Printout Problem? Basically, for these reasons:

THERE'S NO BEATING MASTERTRONIC

NEW

An all action large scale arcade adventure in which you'll need every ounce of skill and courage to destroy the forces of evil.

Traverse the Labyrinth, defy fireball breathing dragons and release the city from its curse by reuniting the 5 pieces of the Golden Talisman.

NEW

Caught on the surface of a nuclear devastated planet you have seconds to return underground before your radiation shield decays.

In any other car you would stand no chance – in the Last V-8, survival is possible . . . Maybe!

FEATURES

360° smooth scrolling, superb voice synthesis, 3 voice professionally composed theme music, realistic life simulation.

THE MASTERTRONIC PLEDGE

We promise to bring you two outstanding added dimension programs every month.

MASTERTRONIC HAVE DONE IT AGAIN!

Bringing you an added dimension for your Commodore 64/128 with these two outstanding programs on Disc. **Hero of the Golden Talisman** and **The Last V8** for the incredible price of only **$9.99** each.

Mastertronic International Inc. 7311B Grove Road, Frederick, Maryland 21701 Tel: (301) 695 8877

1. Endless-printout programs vary in complexity, from very simple to super-complex. There is essentially no upper limit to the potential complexity of these programs.

2. Any time you attempt to write a program which identifies all endless-printout programs, you will discover that your program "breaks down" when it attempts to analyze programs beyond a certain degree of complexity.

(Parts of the following discussion are moderately difficult technically. If you are not interested in technical details, skim this discussion and go on to the next section of the article.)

To begin, let's look at some terms and assumptions that will help you understand our explanation more easily.

1. We will imagine ourselves working with a "souped-up" version of Commodore BASIC which allows programs of unlimited length, and which allows numeric variables to hold numbers of unlimited size. (It is possible to implement these enhancements to Commodore BASIC, although an explanation of the method would be beyond the scope of this article.)

2. We will abbreviate "endless-printout program" as EPP.

3. We define an "EPP Identifier" ("EPPI") as a program which can analyze other programs, and determine in some cases whether the program in question is an EPP or not. An EPPI works in the following way: You "feed" the EPPI a program. The EPPI analyzes the program, and then eventually displays one of two messages on the screen:

Computers have certain fundamental limitations which render them absolutely unable to perform certain kinds of tasks.

THE PROGRAM IS AN EPP
or
THE PROGRAM IS NOT AN EPP

(We also allow the EPPI the possibility of sometimes giving no response at all—i.e., sometimes it may analyze a program and not reach a conclusion regarding whether it is an EPP or not.)

An EPPI never tells a lie. If an EPPI says that a certain program is an EPP, then it is in fact an EPP. If an EPPI says that a certain program is not an EPP, then in fact it is not an EPP.

It is easy to develop EPPI programs which correctly identify limited classes of EPPs. We are going to show however that *no* EPPI, no matter how sophisticated, can successfully identify *all* EPPs. Imagine, for instance, some EPPI program—let's call it FRED1. Let's see if there are some EPPs which are not successfully identified as such by FRED1.

Using FRED1, we can create another program, call it FRED2, which enumerates all of the EPPs identified by FRED1, and which numbers these EPPs in chronological order. In other words, FRED2 would generate a master listing of all programs which FRED1 says are EPPs. The programs in the master listing would be numbered 1, 2, 3, and so on. We will refer to the programs in this listing as EPP-1, EPP-2, EPP-3, etc.

Using FRED2, we could create a subroutine, call it SFRED3, which generates a "cross section" of all of the endless printouts produced by all the EPPs in the master listing. More precisely, SFRED3 would accept an integer variable N, and return a variable LN$.

The contents of LN$ are equal to whatever appears in the Nth line of the endless printout produced by program N in the master listing. For example, if N were 793, the subroutine would determine what is the 793rd program in the master listing produced by FRED2. The subroutine would then determine what would be the 793rd line in the endless printout produced by that program. The value of LN$ would be whatever appears in the 793rd line of that printout.

Using SFRED3, we can concoct the following program, which we'll call CROSS4:

```
1     REM CROSS4
100   OPEN 1,4:CMD1
200   N=N+1
210   GOSUB 1000
220   IF LN$ = "1" THEN PRINT
      "0"
230   IF LN$ = <> "1" THEN
      PRINT "1"
300   GOTO 200
1000  REM SUBROUTINE SFRED3
      BEGINS HERE
```

CROSS4 will generate an endless printout consisting of zeros and ones. Thanks to SFRED3 and lines 220-230, this endless printout is guaranteed to be at least slightly different from each of the endless printouts produced by the programs EPP-1, EPP-2, EPP-3 and so on. The first line of the endless printout produced by CROSS4 is different from the first line of the endless printout produced by EPP-1. The second line of the endless printout produced by CROSS4 is different from the second line of the endless printout produced by EPP-2, and so on.

But this means that CROSS4 is an EPP which was not identified as such by FRED1. This shows that FRED1 fails to identify some EPPs. There are some EPPs which FRED1 does not

Figure 1. Text Generation

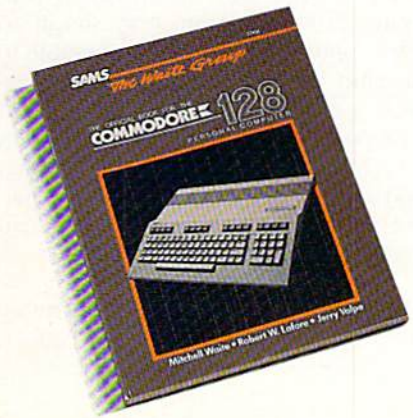
```
ABAZABAAABAZABAABAZABABBABAZABABABAZABAB
ABAZABAABAZABABAABAZABABABAZABAABAZABABA
BAZABABABAZABAABAZABABZZABAZABABABAZABAA
BAZABABZABAZABABABAZABAABAZABABZABAZABAA
BAZABABABAZABAABAZABABZAABAZABABABAZABAA
BAZABABZABAZABAABAZABABABAZABAABAZABABZA
BAZABABABAZABAABAZABABZABAZABAABAZABABAB
AZABAABAZABABZBBABAZABABABAZABAABAZABABZ
ABAZABAABAZABABABAZABAABAZABABZBABAZABAB
ABAZABAABAZABABZABAZABAABAZABABABAZABAAB
AZABABZBABAZABAABAZABABABAZABAABAZABABZA
BAZABAABAZABABABAZABAABAZABABZBAABAZABAB
ABAZABAABAZABABZABAZABAABAZABABABAZABAAB
AZABABZBABAZABAABAZABABABAZABAABAZABABZA
BAZABAABAZABABABAZABAABAZABABZB
```

New 128 news

If you've ordered the new Commodore 128™ or you're thinking about it, Sams has news for you! A new book: The first and only official guide to Commodore's exciting new business computer.

Now you can read all about the Commodore 128 and its three different operating modes—64,™ 128, and CP/M.® You'll learn how to use spreadsheets, and word processing, choose peripherals, create graphics and animation, run CP/M programs, and more.

Best of all, you'll learn how the 128 can work for you at the office or at home. So whether you've already ordered the 128 or are getting ready to, you'll find Sams Official Guide a wise investment.



Get the book that's making headlines. Get Sams Official Commodore 128 book. Visit your local Sams dealer. Or call our toll-free number and ask for Operator 205.

The Official Book for the Commodore 128 Personal Computer, No. 22456, $12.95.

800-428-SAMS
In Indiana, call 317-298-5566

SAMS

Howard W. Sams & Co., Inc.
4300 West 62nd Street
Indianapolis, IN 46268

"recognize" to be EPPs.

This concludes our argument that no computer program can (completely) solve the Endless Printout Problem. We have shown that even a program which can correctly identify some endless-printout programs, cannot identify them all—we can actually concoct an example of an endless-printout program which it cannot successfully identify. The fundamental obstacle to developing a program which can identify all endless-printout programs, seems to be this: Endless-printout programs vary in complexity from very simple to super-complex. There is no single EPPI (Endless-Printout Program Identifier) which can span the entire range of complexity.

Other Examples of Impossible Tasks

Computer scientists have discovered a wide variety of tasks which are impossible for computers. There is no easy way to take an overview of the range of impossible tasks—they fall into many categories, and some of the categories are highly technical in character. But to help you to get a better sense of what the range is like, here are a few examples of other impossible tasks for computers.

An optimizing program is defined as a program which analyzes other programs, and re-writes them so that they run faster. A very simple example of an optimizing program is one that removes all REM statements from another program. (Removing REM statements will speed up a program.) More advanced optimizing programs can locate inefficient logical structures in a program, and improve those structures. For instance, a good optimizing program might rewrite this program:

```
100 X = X+7
110 PRINT X
120 IF X < 98 THEN 300
130 IF X = 98 THEN 300
140 IF X > 98 THEN END
300 GOTO 100
```

to look like this:

```
100 X = X+7
110 PRINT X
120 IF X <= 98 THEN 100
```

The second version does the same job as the first version, but it runs faster.

Some scientists believe that human beings can behave in a "non-scripted" manner. If this is the case, then perhaps humans could perform some of the tasks that are impossible for computers.

Is it possible to design an *Ultimate Optimizing Program,* which would analyze any other program, and re-write it in a fastest-possible version? The answer is no.

A "text-generator" is defined as a program which generates a stream of text. For instance, the stream of text in Figure 1 will be generated by this text-generator program:

```
100 T$ = "ABAZABA"
200 FOR I = 1 TO 7
210 M$ = MID$(T$,I,1)
220 IF M$ = "A" THEN 400
300 E$ = E$ + M$:GOTO 500
400 E$ = E$ + T$ + E$
500 PRINT E$;M$;E$;
510 NEXT I
```

This example shows that a text generator can be much smaller than the stream of text which it generates. In such a case, we say that the text-generator program is a "compression" of the text which it generates.

It is possible to develop text-compression programs which analyze a body of text, and then concoct a text generator which will generate that body of text. Text compression programs can range from very simple to very sophisticated. but is it possible to develop an *Ultimate Text Compression* program? (An Ultimate Text Compression is defined as a program which can analyze any body of text, and then concoct the smallest possible text-generator program that can generate that body of text.)

The answer is no.

There are various games involving the manipulation of symbols, which computers are unable to fully analyze. For example, CHANGO is a game in which a player tries to change one word into another using these transformation rules:

1. If a word begins with a vowel, you may delete the vowel from the beginning, add it to the end, and add any one consonant after that. (For example, EGG may be changed to GGEZ.)
2. If a word begins with a consonant, you may delete the consonant from the beginning, add it to the end, and add any one vowel after that. (For example, DOG may be changed to OGDI.)
3. If two identical letters occur next to each other, they may both be deleted. (For example, DOLLAR may be changed to DOAR.)

The following example will show how these rules are applied to change the word RAG into REAL:

RAG	
AGRA	(RULE 2)
GRAAG	(RULE 1)
GRG	(RULE 3)
RGGA	(RULE 2)
RA	(RULE 3)
ARE	(RULE 2)
REAL	(RULE 1)

With the above three rules, it is fairly easy to determine whether or not it is possible to transform one word into another. However, with other sets of transformation rules, it is sometimes very difficult to determine whether it is possible to transform one word into another.

Is it possible to design *a computer program which can determine, for any set of transformation rules and any pair of words, whether it is possible to transform one word into the other?* The answer is no.

A Fundamental Limitation of Computers

You have now seen several examples of specific tasks which are impossible for computers. These examples show that inherent in the concept of the computer are certain fundamental limitations, which render computers incapable of performing certain kinds of tasks. In what follows, we will attempt to characterize, in part, the nature of these limitations.

The computer can be thought of as a multi-purpose machine, which can

"behave" in many different ways. Its "behavior" is determined by what program resides in it. The program which resides in a computer can be thought of as a *complete description* of how the computer will behave. If you know what program resides in a computer, you can predict exactly how the computer will behave. Sometimes the computer has been compared to an actor, and a program to a script. A computer behaves according to the script (program) which is given to it.

The ability of a computer to behave differently according to different scripts is, of course, one of its great strengths. However, it also turns out to be a limitation. A computer can behave only in ways which can be expressed as a program. Or, to put it another way: If there are forms of behavior which are too complex to be reduced to a script (program), a computer is not able to achieve that kind of behavior.

It turns out that there are tasks which are so complex that the behavior required to carry out these tasks cannot be expressed as a script. You have seen several examples of such tasks in this article.

It is conceivable that there are machines, organisms, or agents of some kind, which can behave in ways that cannot be reduced to a script. If there are such machines, organisms, or agents, it is possible that they could perform tasks which are impossible for computers. Some scientists believe that human beings (and possibly other kinds of organisms), can behave in a "non-scripted" manner. If this is the case, then it is possible that humans could perform some of the tasks listed in this article, which are impossible for computers. Can humans (or other organisms, agents, or machines) in fact behave in a "non-scripted" manner? This is a wide-open question, and it is certainly one of the deepest and most fascinating questions in all of computer science.

Isaac Malitz is a computer consultant and systems designer who specializes in accounting and database applications. He is co-author of *The Commodore 128 Mode: An Inside View*, published by Microcomscribe. C

Trusted Software and

Language Software

For Commodore Computers

ADA TRAINING COURSE

Teaches you the language of the future. Comprehensive subset of language. Includes: editor; syntax checker, compiler; assembler; disassembler, handbook. $39.95

ASSEMBLER/MONITOR

Macro assembler and extended monitor. Supports all standard functions plus floating point constants. Monitor supports bank switching, quick trace, single step, more. $39.95

BASIC COMPILER

Compiles complete BASIC language into either fast 6510 machine language and/or compact speedcode. Supports overlays and many extended BASIC commands. C-64 $39.95
C-128 $59.95

SUPER C COMPILER

Full compiler, Kernighan & Ritchie standard, but without bit fields. Includes powerful editor (41K source file); compiler, library (supports many functions) and linker. C-64 $79.95
C-128 $79.95

FORTH LANGUAGE

Based on Forth 79 (+ parts of '83). Supports hires graphics and sound synthesizer. Full screen editor, programming tools, assembler, samples, handbook. $39.95

MASTER

Professional development package for serious applications. Indexed file system, full screen & printer management, programmer's aid, multi-precision math, royalty-free runtime, more. $39.95

VIDEO BASIC

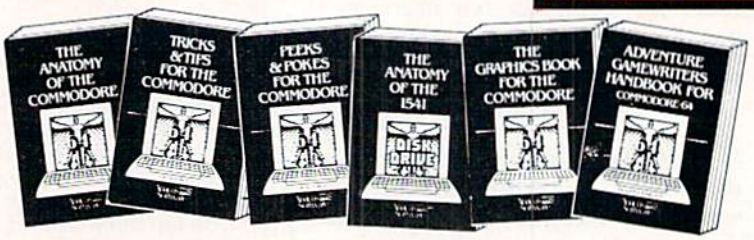
Add 50+ graphic, sound and utility commands to your programs with this super development package. Free distribution of RUNTIME version - no royalties! $39.95

XREF-128 & XREF-64

BASIC cross-reference

Indispensible tool for BASIC programmers. Finds all references to variables, constants & line numbers. Sorts in alphabetical order. C-64 $17.95
C-128 $17.95

Reference Books



ANATOMY OF C-64 Insider's guide to the '64 internals. Graphics, sound, I/O, kernal, memory maps, and much more. Complete commented ROM listings. 300pp $19.95

ANATOMY OF 1541 DRIVE Best handbook on this drive, explains all. Filled with many examples programs and utilities. Fully commented 1541 ROM listings. 500+pp $19.95

MACHINE LANGUAGE FOR C-64 Learn 6510 code & write fast programs. Many samples and listings for complete assembler, monitor and simulator. 200pp $14.95

GRAPHICS BOOK FOR C-64 Best reference, covers basic and advanced graphics. Sprites, Hires, Multicolor, 3D-graphics, IRQ, CAD, projections, curves, more. 350pp $19.95

TRICKS & TIPS FOR C-64 Collection of easy-to-use techniques: advanced graphics, improved data input, enhanced BASIC, CP/M, data handling and more. 275pp $19.95

1541 REPAIR & MAINTENANCE Handbook on the drive's hardware. Includes schematics & techniques to keep .1 running. Align drive w/ & w/o scope. Large handbook size. $19.95

ADVANCED MACHINE LANGUAGE Subjects not covered elsewhere: video controller, interrupts, timers, I/O, extensions to BASIC. Tips for the serious programmer. 210pp $14.95

PRINTER BOOK C-64/VIC-20 Understand Commodore, Epson compatible printers & 1520 plotter. Utilities, screen dump, 3D-plot, commented MPS-801 ROM listings. 330pp $19.95

SCIENCE/ENGINEERING ON C-64 In-depth introduction to computers in science. Some topics covered are chemistry, physics, astronomy, electronics & others. 350pp $19.95

CASSETTE BOOK C-64/VIC-20 Make your cassette run faster than a disk drive! Cassette data-base, disk to tape backup, tape to disk, FastTape operating system. 225pp $14.95

Productivity Tools

TECHNICAL ANALYSIS SYSTEM

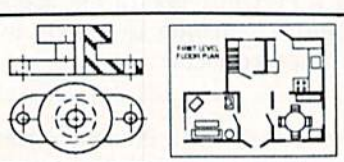
A sophisticated charting and technical analysis system for serious investors. By charting and analyzing the past history of a stock, TAS can help pinpoint trends & patterns and predict a stock's future. TAS lets you enter trading data from the keyboard or directly from online financial services. $59.95

PERSONAL PORTFOLIO MANAGER

Complete portfolio management system for the individual or professional investor. Allows investors to easily manage their portfolios, obtain up-to-the minute quotes & news, and perform selected analysis. $39.95

CADPAK

A deluxe graphics design and drawing package. Use with or without an optional lightpen to create highly-detailed designs. With dimensioning, scaling, text, rotation, object libraries, hardcopy and more. $39.95



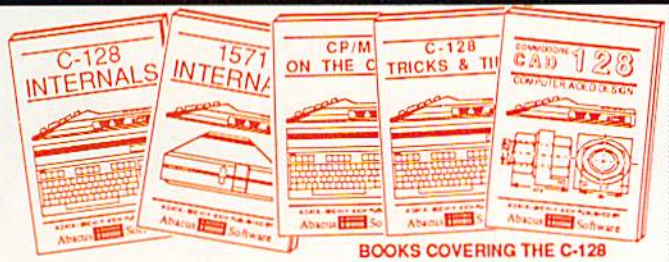
DATAMAT

Powerful, easy-to-use data management package using menu selections. Free-form design, 50 fields/record, 2000 records/disk. Sort on multiple fields in any combination. Complete selection and formatting for printing reports. $39.95

Authoritative Books

From Abacus Software

...a name you can count on



BOOKS COVERING THE C-128

IDEAS FOR USE ON C-64 Themes: auto expenses, calculator, recipe file, stock lists, diet planner, window advertising, others. Includes all program listings. 200pp $12.95

COMPILER BOOK C-64/C-128 All you need to know about compilers: how they work, creating your own and generating the final machine code. 300pp $19.95

Adventure Gamewriter's Handbook A step-by-step guide to designing and writing your own adventure games. Adventure game generator & four example games. 200pp $14.95

PEEKS & POKES FOR THE C-64 Includes in-depth explanations of PEEK, POKE, USR, and other BASIC commands. Learn the "inside" tricks about your '64. 200pp $14.95

OPTIONAL DISKETTES FOR BOOKS For your convenience, the programs contained in each of our books are available on diskette. All program thoroughly tested & error-free. Specify title of book when ordering. **$14.95 each**

C-128 INTERNALS Detailed guide presents the 128's operating system, explains the graphics chips, Memory Management Unit, and commented listing of Kernal. 500+pp $19.95

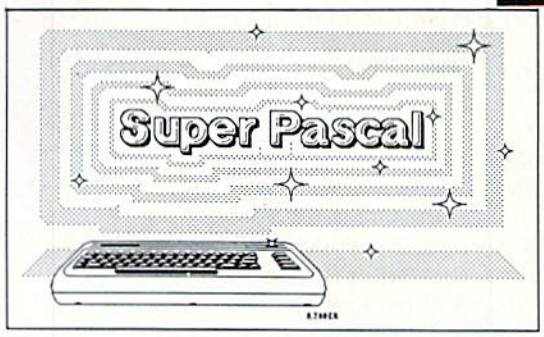
1571 INTERNALS Insiders' guide for novice and advanced users. Covers sequential & relative files, and direct access commands. Describes important DOS routines. Commented DOS listings. 500+pp $19.95

C-128 TRICKS & TIPS Chock full of info for everyone. Covers 80 column hi-res graphics, windowing, memory layout, Kernal routines, sprites and more. 300 pp $19.95

CP/M ON THE C-128 Essential guide to using CP/M on your 128. Simple explanations of the operating system, memory usage, CP/M utility programs, submit files and more. $19.95

COMPUTER AIDED DESIGN on your C-128 or 64. Create a CAD system using programs provided. Covers 3D objects & rotation, MACROS, hatching, zooming, mirroring, line widths, dashed lines, more.**300 pages** $19.95

Special Feature



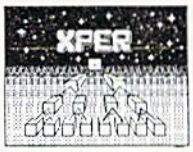
For school or for software development, **SUPER Pascal** rivals products that run on much more expensive computers.

In addition to its extensive compiler, **SUPER Pascal** includes an easy-to-use and very complete source file editor, a very capable assembler for optionally coding in machine language routines, and a comprehensive utility package. **SUPER Pascal** uses high-precision 11-digit arithmetic and includes a much faster DOS (3X with C-64/1541). With these features, **SUPER Pascal** transforms the Commodore into a highly capable Pascal machine.

If you've been wanting to learn Pascal or to develop software using the finest tools available, **SUPER Pascal** is your first choice. Complete with 200+ page handbook. C-64 $59.95
C-128 $59.95

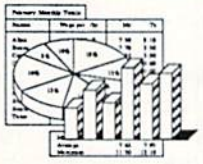
XPER

Capture your information on XPER's knowledge base and let this first *expert system* for Commodore computers help you make important decisions. Large capacity. Complete with editing & reporting. $59.95



POWERPLAN

One of the most powerful spreadsheets with integrated graphics for your Commodore computer. Includes menu or keyword selections, online help screens, field protection, windowing, trig functions and more. PowerGraph lets you create integrated graphs and charts from your spreadsheet data. $39.95

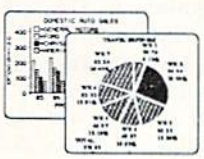


QUICKCOPY V2.0

Back up your valuable data with the fastest disk copier we've seen to date. Copies an entire disk in two and a half minutes on two drives or three and a half on one. $19.95

CHARTPAK

Make professional-quality pie, bar and line charts, and graphics from your data. Includes statistical functions. Accepts data from CalcResult and MultiPlan. C-128 has 3X the resolution of the C-64 version. Outputs to most printers. C-64 $39.95
C-128 $39.95



Ordering Information

Abacus Software

MasterCard VISA® AMERICAN EXPRESS Card

P.O Box 7211 Grand Rapids, Michigan 49510

For Postage and handling include $4.00 per order. Foreign orders include $10.00 per item. Money order and checks in U.S. Dollars only. MasterCard, VISA and American Express accected. Michigan residents please include 4% sales tax.

For *fast* service call (616) 241-5510 Telex 709-101

For **free** catalog, please return this coupon or a copy to:
Abacus Software, P.O. Box 7211, Grand Rapids, MI 49510

PHONE: (616) 241-5510

Name_____
Address_____
City_____
State_____ Zip_____

CM 2/86

BY JIM GRACELY

Tech Notes

Technical editor Jim Gracely keeps you abreast of the latest and greatest.

Toshiba has a tremendous new dot matrix, graphic printer called the P351 or 3-In-One. This is a very fast (288 cps in draft mode), full-featured printer. The "three" in the name refers to the three kinds of fonts available—built-in, downloadable and cartridge. The built-in fonts include a draft font at 288 cps and two letter-quality fonts (prestige elite and courier) at 100+ cps.

In addition, there are selectable pitches including condensed, proportional and elongated. Selections are made through DIP switches on the printer, and are software selectable. Both Centronics parallel and RS232 ports are provided on the printer (also DIP selectable), and the printer has a 4K buffer. Dot-addressable graphics are also supported with up to 180-by-360 dots per inch.

Silencing materials are built into the case and a fan is included. Whenever printing is to begin, the fan kicks into high and a few seconds after printing stops, the fan shifts to low gear.

The manual provides complete documentation on all of the printer functions, DIP switch settings, and escape codes. A technical manual is also available directly from Toshiba. A quick reference guide includes a very useful chart of comparable printers (in order of preference). So, if your word processor doesn't have a Toshiba P351 listed, you know what other selections may work (I tried selecting a C. Itoh F-10 and Diablo 630 in *WordPro 64* from Proline, and the P351 worked just fine.)

The price on this printer simply reflects the quality and range of features. That price? $1699 retail. Additional font cartridges are available for $49.50 each.

A new technology for data storage and retrieval has been developed by Entrepo, Inc. out of Sunnyvale, California. They have produced a wafer-tape drive for the VIC 20 (with 24K) and Commodore 64 called the Quick Data Drive. The drive plugs into the cassette port of the computer and requires no external power. The micro-wafers look like dictaphone tapes, and store from 25K (10 feet) to 170K (62 feet). The operating system for the drive (Quick Operating System or QOS) is loaded from a master micro-wafer (optionally available in a cartridge). Once installed, the computer uses datassette syntax to access the drive.

The transfer rate is never actually stated anywhere. However, the claims are that it is 15 times faster than a datassette and will load a 24K program in 20 seconds (approx. 1200-1500 bits/second). Keep in mind that this is in addition to the access time required to FIND the file on the wafer (up to 55 seconds for a 62-foot wafer).

File operations are handled in the same way as a datassette with a couple of nice improvements. First is the elimination of the fast-forward and rewind buttons—the Quick Data Drive takes care of finding the files for you. Secondly, there is a new reserved variable (ST) added to BASIC to provide the status of the last operation. Finally, a File Management Utility (FMU) is included with the drive, which contains routines for file manipulations. Included are routines for copying between wafers, disks and cassettes, looking at a directory of the wafer, deleting files, and formatting wafers. There's even a program for cleaning the head of the drive (used with an additional cleaning wafer).

In all, the Quick Wafer Drive seems to be a nicely designed, transportable, and inexpensive ($84.95) alternative to a full disk drive. Many nice features have been incorporated into both the overall design and the wafers themselves. The long access times can be kept to a minimum by using wafers that best conform to the files you are using. There's no need to have a 170K wafer for a couple of 5-10K public domain packages. Software support even seems to be coming, with both Epyx *(Impossible Mission)* and Cosmi *(Aztec Challenge)*, putting games in wafer format. C

ATTENTION PLUS/4 OWNERS!

My Plus/4 Software List.

Quantity	Product	Name	Price
____	TC301 (D&T)	Typing Professor	$12.00
____	TC600 (D)	The Hulk	$21.00
____	TC601 (D)	Cutthroats	$24.00
____	TC602 (D)	Suspect	$24.00
____	T112001 (C)	Script Plus	$45.00
____	T112003 (C)	Calc Plus	$45.00
____	T112004 (T)	Experiences In Software	$12.00
____	T112005 (T)	Money Decisions - Loans	$18.00
____	T112006 (T)	Money Decisions - Investments	$18.00
____	T112007 (T)	Productivity Starter Kit	$18.00
____	T113007 (T&D)	Easy Match/Easy Count	$15.00
____	T113008 (T&D)	What's Next - Letters/Numbers	$15.00
____	T113009 (T&D)	Letter Match - More Or Less	$15.00
____	T113010 (T&D)	Letter Sequences - Long/Short	$15.00
____	T113011 (T&D)	Shapes And Patterns/Group It	$15.00
____	T116001 (C)	Viduzzles	$18.00
____	T116002 (C)	Jack Attack	$18.00
____	T116004 (C)	Pirate Adventure	$18.00
____	T116005 (C)	Atomic Mission	$18.00
____	T116006 (C)	Strange Odyssey	$18.00
____	T117001 (T)	Math Games - Word Problems	$12.00
____	T117002 (T)	Super Spell	$12.00
____	T117003 (T)	Maze Mania	$12.00
____	T232001 (C)	Financial Advisor	$27.00
____	T232007 (D)	Loans And Investments	$30.00
____	T232008 (D)	Business Management & Statistics	$30.00
____	T262006 (D)	General Ledger	$45.00
____	T262007 (D)	Accounts Receivable	$45.00
____	T262008 (D)	Accounts Payable	$45.00
____	T262009 (D)	Inventory	$45.00
____	T262010 (D)	Payroll	$45.00
____	T262011 (D)	Construction Industry	$12.00
____	T262012 (D)	Personal Finance	$12.00
____	T262013 (D)	Retail Industry	$12.00
____	T262014 (D)	Wholesale Industry	$12.00
____	T262015 (D)	Manufacturing Industry	$12.00
____	T262016 (D)	Professional Services	$12.00
____	T262017 (D)	Maintenance & Services	$12.00
____	T263001 (C&D)	Logo	$45.00
____	T263010 (D)	Reading Professor	$36.00
____	T266001 (D)	Zork I	$24.00
____	T266002 (D)	Zork II	$24.00
____	T266003 (D)	Zork III	$24.00
____	T266004 (D)	Starcross	$27.00
____	T266006 (D)	Suspended	$27.00
____	T266022 (D)	Planetfall	$24.00
____	T266023 (D)	The Witness	$24.00
____	T266024 (D)	Hitchhikers Guide To The Galaxy	$24.00

(D) = Disk (C) = Cartridge (T) = Tape

Our New Year's Resolution Is To Save You Money!

It's that time of the year again when everyone is making resolutions—including Commodore. And it's a resolution that could save you a great deal of money on your favorite Plus/4 software.

Software that will educate you. Entertain you. Allow you to work more efficiently. If you own a Plus/4 or know someone who does, this could be the beginning of a very happy new year!

- -

Please Rush Me The Software I've Indicated On The Plus/4 Software List.

Name _____

Address _____

City _____ State _____ Zip _____

Signature _____

Enclosed is my check or money order for $_____.
Make check or money order payable to Commodore Direct Marketing, Inc. (PA residents add 6% sales tax)
Add $2.00 shipping and handling to all orders.

Send To:
Commodore Direct Marketing
1200 Wilson Drive
West Chester, PA 19380

Please make sure BOTH the list AND address coupon are enclosed.

COMMODORE

Commodore Business Machines, Inc.

BY MARK ZIMMERMANN

Random Thoughts

The Karmarkar Algorithm

If Karmarkar's algorithm pans out, we might see fast, efficient optimization become a way of life.

A few years from now, when computer people look back on the 1980's, they won't remember the brouhaha about the "fifth generation" threat, or the New York Times' scare headlines on Japanese supercomputers. The real news of the 1980's is the discovery of a new linear programming algorithm that just might overthrow the conventional wisdom of the last 40 years on optimization.

Programming in this context means not computer programming, but rather planning or arranging affairs to get the most desired outcome. Linear programming is an important special case where the constraints on your actions and the best final results are defined in terms of straight-line functions (no square roots, complicated curves, etc.).

An example may make this clearer. Suppose you want to maximize the amount of vitamin C you eat, and (for simplicity) you have to choose among oranges and lemons in the store. You're constrained by the total amount of weight you can carry home and by the amount of money you have to spend.

Figure 1 shows graphically the result of those constraints. The region outlined in the oranges-lemons plane is the zone of choices allowed by the weight and money constraints. (You can't buy a negative amount of fruit, so the horizontal and vertical axes are two more constraints.) An arrow points in the direction of increasing vitamin C. Clearly, the optimum solution is the circled point, where you buy a mix of fruits.

This is easy enough when there are only two fruits to choose between. Suppose there are dozens? Real-world optimization problems often involve thousands of choices, and thousands of constraints on what's allowed. In the multi-dimensional space of possibilities, these constraints carve out a polyhedral volume called a simplex. The classical method for attacking such a problem is the simplex algorithm developed by George Dantzig in the 1940's. This algorithm starts out at the origin and steps along the edges of the allowed region, from one corner to an adjacent one, moving to better choices until the optimum one is found. The best choice will always be a corner if the programming problem is linear.

In Figure 1, you can see that whichever way the simplex algorithm goes, it's bound to get the optimum solution in only two steps. In a many-thousand dimensional problem, the simplex approach has to take many thousands of steps, on the average. There are a few horrendously bad cases where the algorithm has to visit every corner on its way to the optimum, but they hardly ever come up in real life. Even though it's quite efficient, the simplex method often requires hours of time on large computers to solve big real-world problems.

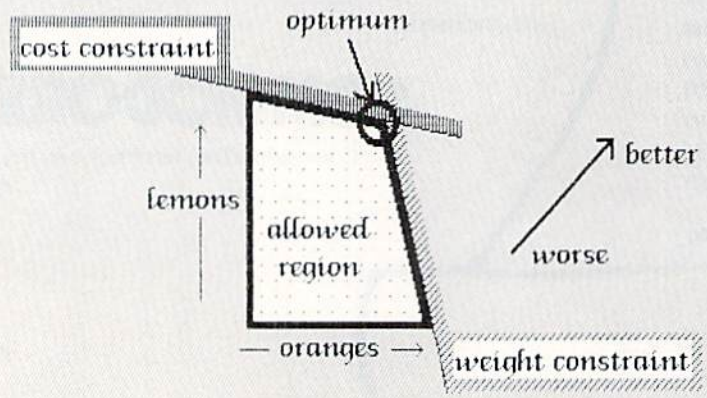


Figure 1. A Linear Programming Problem

A few years ago, a young Russian named L. G. Khachiyan put together some results from other mathematicians and discovered a new way to solve linear programming problems. His ellipsoid algorithm made a big splash, but unfortunately has turned out not to be of any practical importance. Figure 2 illustrates how it works.

The ellipsoid method starts out with a big ellipse (ellipsoidal solid, in higher dimensions) surrounding the entire space of allowable solutions. It then computes a new ellipse which includes the half of the original one the solution is in. At each further step, another ellipse is computed which includes the half of the previous one where the optimum resides. Eventually, the ellipses get small enough to trap the unique solution.

Khachiyan's ellipsoid algorithm is far better than the simplex method in the worst cases. Unfortunately, the best-case behavior of the ellipsoid method is about the same as its worst case! So, while it's of theoretical interest, it hasn't turned out to have much practical utility.

The new Karmarkar algorithm is different. It hasn't been officially published yet, but a description of it has circulated in preprint form. In preliminary tests, it runs as much as 50 times faster than the simplex method on realistic problems with thousands of constraints. What's more exciting is that the simplex approach has been highly optimized over four decades of intense effort, while Karmarkar's approach is still being developed. When fully understood, the new method should do much better.

Narendra Karmarkar is a mathematician at Bell Labs. He's small, dark, and speaks with a lilting Indian accent. When I saw him recently at a National Bureau of Standards seminar, his jeans and plaid shirt were a pleasant contrast with the coat-and-tie for-

MUST LIQUIDATE At Far Below Dealer Cost!
TOTAL Personal Computer System

Factory Reconditioned with Factory Warranty!

Carries easily as a suitcase! Plugs into 115V outlet

GREAT GIFT IDEA!

Sorry, we're not permitted to PRINT the famous brand-name. BUT, we CAN "tell all" if you call us TOLL FREE: 1-800-328-0609!

THE COMPUTER

Snap-on computer keyboard! 64K RAM, 20K ROM. Full-size typewriter keyboard. Upper and lower case letters, numerals, symbols, reverse characters. 2 cursor control keys, 4 function keys, programmable to 8. Music synthesizer with 3 independent voices, each with 9 octave range. Input/output ports accommodate . . . user, serial, ROM cartridge, joysticks, external monitor, phone modem.

Built-in disk drive! Intelligent high speed unit with 5¼" floppy disk recorder. 170K formatted data storage; 35 tracks. 16K ROM. Uses single sided, single density disk. Serial interface. Second serial port to chain second drive or printer.

Built-in color monitor! Displays 40 columns x 25 lines of text on 5" screen. High resolution. 320 x 200 pixels. 16 background, character colors.

Built-in ROM cartridge port! Insert ROM program cartridge. Multitude of subjects available in stores across the nation!

Original List Price $995.00

Liquidation Priced At Only $388

Item H-835-63631-00 Ship. handling: $20.00

THE PRINTER

Print method: Bi-directional impact dot matrix.
Character matrix: 6 x 7 dot matrix.
Characters: Upper and lower case letters, numerals and symbols. All PET graphic characters.
Graphics: 7 vertical dots — maximum 480 columns. Dot addressable.
Character codes: CBM ASCII code.
Print speed: 60 characters per second.
Maximum columns: 80 columns.
Character spacing: 10 characters per inch.
Line feed spacing: 6 lines per inch in character mode or 8 lines per inch selectable. 9 lines per inch in graphics mode.
Line feed speed: 5 lines per second in character mode. 7.5 lines per second in graphics mode.
Paper feed: Friction feed.
Paper width: 4.5" to 8.5" width.
Multiple copies: Original plus maximum of two copies.
Dimensions: 13"W x 8"D x 3¼"H. Wt.: 6½ lbs. **Power:** 120V AC, 60 Hz.

Original List Price: $200.00

Liquidation Priced At $118

Item H-835-63681-00 Ship. handling: $7.00

THE SOFTWARE

"Easy Script" One of the most powerful word processors at any price! Cut re-typing, create documents from standard paragraphs, do personalized letters, see and change a document before it is printed. Instruction manual has extensive training section that simplifies use . . . even for someone who has never used a computer or word processor before!

"The Manager" A sophisticated database manager for business or home use. Business uses: accounts payable/receivable, inventory, appointments, task manager. Home uses: mailing lists, home inventory, recipes, collection organizer, investment tracking, checkbook balancing. School uses: research article index, gradebook.

Original List Price: $73.98

Liquidation Price $24

Item H-835-64011-03 Ship. handling: $3.00

BUY INDIVIDUAL UNITS OR GET THIS ULTRA-FAMOUS SYSTEM AT ONE LOW PACKAGE PRICE!

TOTAL Personal Computer System available at FAR BELOW dealer cost!

Original List Price $1,293.00

TOTAL PACKAGE PRICE $488

Item H-835-64011-02 Ship. handling: $24.00

Compatible with above Computer System (Not included in package price.)

JOYSTICKS (Set of 2)
Mfr. List: $59.90 pr.
Liquidation Price . . . . . $18 pr.
Item H-835-63622-01 S/H: $6.00 pr.

64K MODEM (Factory Reconditioned w/Warranty)
Mfr. List: $124.95
Liquidation Price $19
Item H-835-63646-01 S/H: $4.00

VISA MasterCard

Credit card members can order by phone, 24 hours a day, 7 days a week.

Toll-Free: 1-800-328-0609

TeleCheck Your check is welcome! No delays when you pay by check!

C.O.M.B. CO.®
Authorized Liquidator
14605 28TH AVENUE NORTH
MINNEAPOLIS, MINNESOTA 55441-3397

SEND ME THE ITEMS I HAVE LISTED BELOW
Price subject to change after 60 days. Sales outside continental U.S. are subject to special conditions. Please call or write to inquire.

Item No.	#	Item	Price	S/H
		TOTAL		

C.O.M.B. CO.® Item H-835
14605 28th Ave. N./Minneapolis, MN 55441-3397

Send the items indicated at left. (Minnesota residents add 6% sales tax. Please allow 3-4 weeks delivery. Sorry, no C.O.D.)
☐ My check or money order is enclosed. (No delays in processing orders paid by check, thanks to TeleCheck.)
Charge: ☐ MasterCard® ☐ VISA
Acct. No. _____ Exp. _____
PLEASE PRINT CLEARLY
Name _____
Address _____
City _____
State _____ ZIP _____
Phone _____
Sign here _____

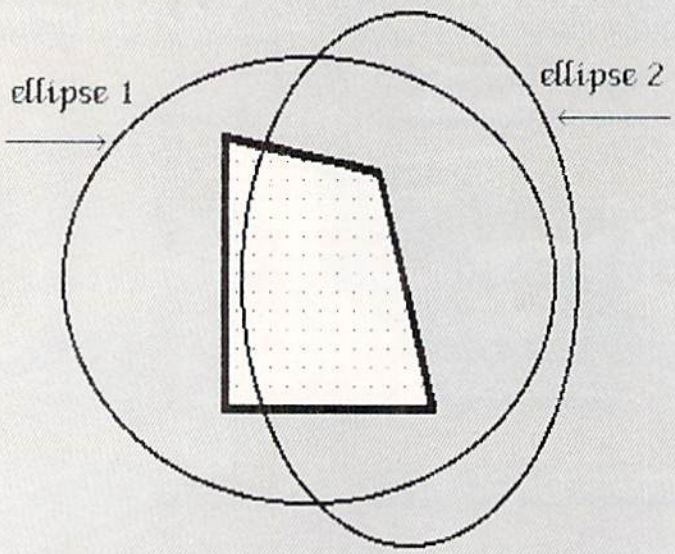


Figure 2. Khachiyan Algorithm

mality of the other speakers. Karmarkar's work is not yet complete; a mathematician friend estimates that it will be at least five or ten years before the proofs Karmarkar sketched out are made rigorous. But meanwhile, programs are already being written to test the ideas out.

Figure 3 shows the fundamental concept. Karmarkar's algorithm begins with a circle (or sphere, in higher-dimensional problems) inscribed inside the allowed solution space. It then moves to the point on that circle where the desired quantity is maximized. Another circle is then drawn around that point, still all inside the allowed region, and the process is repeated.

There are plenty of details left out of the above description, but the core concept is there. Karmarkar uses projective transformations (like the shadow cast by shining a light through a graph drawn on a transparency) in order to keep his circles circular and of maximum size as they approach the solution. He also remaps the original problem dynamically so that the optimum solution ends up at the origin of the coordinate system. And there are other tricks needed to make the method run efficiently on large problems in large computers.

So you see, when the Karmarkar algorithm takes a step, it's a big one, through the middle of the problem, not a mincing little move along the edge like the simplex method. Each of the first few steps of Karmarkar does as much work as thousands of simplex steps in a big problem. And as problems get bigger, the relative speed advantage of Karmarkar's algorithm over simplex grows.

What will it ultimately mean? That's hard to say. The famous line, "What good is a newborn baby?" comes to mind. The nearest analogy is the rediscovery of the fast fourier transform (FFT) in the 1960's. It led to startling advances in real-time signal processing that are only beginning to become apparent today, along with spin-offs to dozens of other areas.

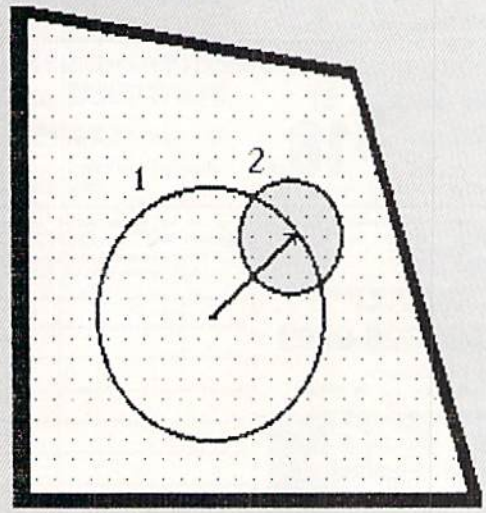
If Karmarkar's algorithm pans out, we might see fast, efficient optimization become a part of everyday life. A business could re-direct its activities to better respond to consumer demands; a transportation system could re-route buses, subway cars, and automobile traffic to minimize congestion and delays; a chemical factory could maximize output and minimize waste; an airplane could reconfigure itself in flight to go farther or faster on less fuel. None of these problems are computationally feasible now. To move around a room and respond fully to its environment, a state-of-the-art robot has to stop every few seconds to calculate for minutes on a big remote computer. What if all that computational work could be done instantly?

A few years ago, it was fun to try to find all the electric motors hidden in one's house. Clocks, washing machines, fans, shavers, old auto-tuning TV sets, refrigerators—the list went on and on. Cheap, small motors were ubiquitous, and we forgot about them.

Now, try finding all the computers in your house! Don't forget the telephone, the microwave oven, the electronic typewriter, the thermostat, the TV—if they're less than a few years old, odds are that they have at least one, and probably several, microprocessors inside. Cheap, small computers are becoming ubiquitous, and we forget about them.

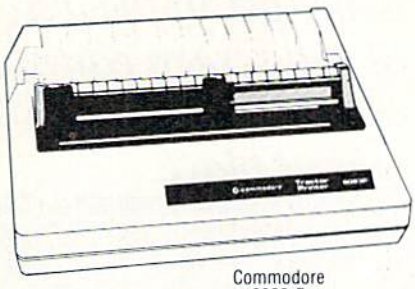
As the computers proliferate, algorithms to make them do more work in less time become the crucial limiting factor. Karmarkar's algorithm for optimization may be the big step forward for the 1980's in that department. C

Figure 3. Karmarkar Algorithm



The Commodore CBM System For Business Classes & School Offices.

- 80 Column x 25 Line screen . . . Typewriter style keyboard
- BIG Disk Capacity – either 1,000K-bytes or 2,000K-bytes
- Fast, big and reliable 150 cps, 132 column printer
- Proven dependability – over 100,000 in schoolroom use throughout the U.S.
- Lowest cost big drive system available anywhere
- Many, Many programs available including:
  - –Professional Wordprocessors
  - –Powerful spreadsheets
  - –Easy to use Databases
  - –Telecommunications software
  - –Comprehensive Accounting Packages
  - –School Management Software
  - –Administrative Software



Commodore 8023-P Printer

RECOMMENDED PRICES:

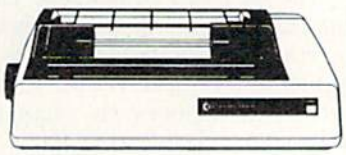
Computer: CBM8032	$449.00
Disk Drive: 8050 or 8250	$595.00
Printer: 8023P	$249.00
Cables	$68.00
TOTAL if bought separately	$1361.00

Commodore CBM-8032 Computer

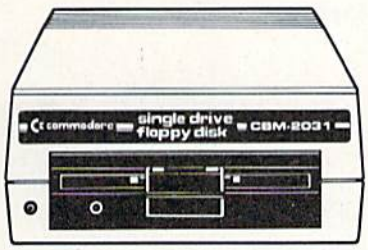
Commodore 8050 or 8250 Disk Drive

Special Commodore CBM System Price..... $1099

The Commodore PET Classroom Solution . . .



Commodore CBM Bi-Directional Printer Model 4023



Commodore CBM-2031 Single Disk Drive

ORDERS ONLY CALL
1-800-527-1738

INQUIRIES & INFORMATION PLEASE CALL
1-214-231-2645

Commodore PET Model 4032

PLUS 750 EDUCATIONAL PROGRAMS
VERY IMPORTANT! These programs will work with Commodore's C64 and the new C128, as well as with the PET and CBM.

If these programs were only worth $5.00 each, this software alone would be worth over $3750!

Package price for the Commodore PET System.....$749.00

- 40 Column x 25 Line screen . . . easily read by young students
- 170 Disk Drive
- Fast and reliable 100 cps, 80 column printer.
- Incredibly rugged . . . this system keeps on going, year after year
- Proven under heavy use in Entitlement Programs throughout the U.S.
- Low, low cost per system
- Over 750 Programs in this well-tested 4th Edition of the popular Public Domain series for K-12. Includes:
  - –Administration
  - –Business
  - –Computer Science
  - –English
  - –French
  - –Geography
  - –Health Education
  - –Language Arts
  - –Games
  - –Logic & Problem Solving
  - –Math
  - –Music
  - –Physical Education
  - –Science
  - –Technology
  - –Utilities

MasterCard, VISA, Money Order, Bank Check
COD (add 5.00)
Add 3% For Credit Cards
All Products In Stock Shipped Within 24 Hours

commodore
Micro-Sys
DISTRIBUTORS

F.O.B. Dallas, Texas
All Products Shipped With Manufacturers 90 Day Warranty **PRICES ARE SUBJECT TO CHANGE WITHOUT NOTICE.**

. 641 Presidential Drive • Richardson, Texas 75081 • 9:30 a.m.-6:30 p.m. (Mon.-Fri.) • 10:30 a.m.-2:30 p.m. (Sat.)

BY BENNETT COOKSON, JR.

Control-Q

for the Commodore 64

Commodore computers have one of the best screen editors I have ever seen. It is a great help to be able to move the cursor anywhere on the screen and change a program line. But sometimes while I'm typing in programs (especially ones with many strings in quotes), my cursor controls produce unwanted reversed characters. Then, I have to hit RETURN to get out of quote mode, go back up to the line and fix it. Does this sound familiar? If it does, then this is the program for you.

Control-Q gives you complete control of quote mode. As soon as you run the BASIC loader, the machine language routine will be in place and running.

Now when you type quote marks or insert spaces, the cursor controls will work just fine. Quote mode will never get in the way of full-screen editing again, because you have control over it. Typing quotation marks will no longer put your 64 into quote mode. Only pressing the CTRL-Q keys will cause this to happen.

Quote mode will never get in the way of full-screen editing again when Control-Q is in place.

You now have complete control over quote mode. Press CTRL-Q to toggle it on and off. Pressing RETURN will also turn off quote mode.

Lock?

The BASIC loader asks if you want to lock the program so RUN/STOP-RESTORE won't remove it. When the program is locked, the warm-start BASIC vector is changed so that every time BASIC makes a warm start (every time you see the READY message), the vectors at 655-656 will be changed to point to the Control-Q routine. That way Control-Q will always be there when you need it.

How It Works

Control-Q is interrupt-driven. That is, every time the computer reads the keyboard (60 times a second), Control-Q is executed.

The ML routine will POKE 212,0 and POKE 216,0 to clear quote mode and insert mode. Also, the routine checks to see if CTRL-Q is being pressed. If so, it then toggles quote mode, turning it off or on.

The 64 bytes of machine language are placed in a little used area of RAM starting at 679 ($02A7). The BASIC loader will relocate the ML to run anywhere in memory by changing the start address (SA) in line 200.

The last 19 bytes of the program are a routine that changes the vector at 655-656 ($028F-0290) to point to the Control-Q routine. C

Before typing this program, read "How to Enter Programs" and "How to Use the Magazine Entry Program."

Control-Q

```
100 PRINT CHR$(147):PRINT
    TAB(17)"CTRL-Q"'EJNB
110 PRINT TAB(11)"QUOTE MODE
    CONTROL"'CDJC
120 PRINT TAB(9)"BY BENNETT COOKSON
    JR."'CCLE
130 PRINT:PRINT"[SPACE2]
    CTRL-Q TO TURN QUOTE MODE ON AND
    OFF"'CBTJ
200 SA=679'BFKW
300 DEF FN H(D)=INT(D/256)'FKSC
305 DEF FN L(L)=L-FN H(L)*256'GMJI
500 I=SA'BDRA
510 READ A:IF A=256 THEN 700'EJPE
520 POKE I,A:I=I+1:GOTO 510'ELEF
530 DATA 165,212,240,4,169,0,208,7'BAHH
540 DATA 133,212,133,216,141,172,2,
    165'BERJ
550 DATA 145,201,187,208,21,166,203,
    228'BFDK
560 DATA 197,240,12,173,172,2,73,
    255'BCOK
570 DATA 141,172,2,133,212,134,197,
    76'BDJL
580 DATA 66,235,76,72,235,120,169,
    167'BDDM
590 DATA 141,143,2,169,2,141,144,
    2'BAXN
600 DATA 88,96,32,212,2,76,131,164,
    256'BEOG
700 POKE SA+47,FN L(SA)'DKPF
710 POKE SA+52,FN H(SA)'DKHG
750 PRINT:PRINT"LOCK? (STOP/RESTORE
    WON'T DISTROY IT)"'CBCR
755 PRINT" Y OR N"'BASN
760 GET K$:IF K$<>"Y" AND K$<>"N"
    THEN 760'IKXP
770 IF K$="N"THEN 850'DFML
800 POKE 770,FN L(SA+58):REM LOCK'EQNI
810 POKE 771,FN H(SA+58)'DLYH
850 SYS(SA+45)'CGYJ
860 PRINT:PRINT"SYS";SA+45;
    "TO RESTART"'DHXP
890 END'BACL
900 POKE 770,131:REM REMOVE LOCK'CSGI
910 POKE 771,164'BHCF
```

END

Sensational Prices!
. . . On Our Most Popular Items!

from microfal® . . .

THE 79¢ DISKETTE

Are you paying too much for diskettes? Try our first quality, prime, 5¼" diskettes (no rejects, no seconds) at these fantastic sale prices and save, save, SAVE! Disks are packaged in boxes of 50; each box contains 5 shrink-wrapped 10-packs that include diskettes in sleeves, labels, and write-protect tabs.

Each diskette is certified to be 100% error free and comes with a lifetime warranty (if you have a problem, we'll replace the diskette). All diskettes include hub reinforcement rings and write-protect notch.

All diskettes are double density and work in either single or double density drives.

SS, DD Diskettes, Box of 50
32391 **$39.50**–79¢ ea.!

DS, DD Diskettes, Box of 50
32403 **$49.50**–99¢ ea.!

Ready. . . Set. . . PRINT!
PRINTER PLUS GRAPHICS INTERFACE
ONLY $189!!

We've combined a famous name printer with an outstanding graphics interface to offer a complete printer package for your C-64 at an unbeatable value! Similar systems can cost $300 and up. . . but due to our high volume you pay ONLY $189!!

You get a standard printing speed of 100 CPS along with Near Letter Quality *and* compressed print modes available at the flip of a switch. Printing is bi-directional and logic seeking. Both friction and tractor feed are standard—handles original plus two copies.

Other features include graphics printing (compatible with almost all popular graphics software programs), programmable line spacing, and more!

Dimensions: 15.5" W × 11" D × 4" H". Weight: 10.8 lbs.

Here's another plus—the parallel, Centronics compatible printer can be used with *any* parallel output computer. . . lets the printer do double duty, or more!

90 days parts and labor warranty with a lifetime warranty on the printhead.

35044 Printer Plus Graphics Interface **$189.00**

Supply is limited — call now!

POWER and PROTECTION FOR YOUR C-64!

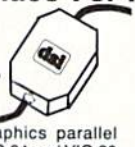
POW'R PAK 64

Pow'r Pak is a replacement power supply (1.5 amp) for the Commodore 64. . . but that's not all! Pow'r Pak also supplies two additional surge protected outlets (120V) for monitor, disk drive, or other peripherals. On/off switch. Fuse protection. Sturdy all-metal casing is ventilated for heat dissipation. Full 1 year warranty.

34910 **$49.95**

The Right Interface For All Your Printing Needs!!

PPI with graphics



This high-performance graphics parallel printer interface from DSI for C-64 and VIC-20 emulates a Commodore printer. Comes complete with all necessary cables for quick installation and includes easy-to-follow user's manual.

33565 **$44.95**

GWIZ Interface

Built-in buffer provides super high-speed printing for parallel printers. Dumps high-resolution screens up to 18 times faster. Prints all Commodore characters. All cables and connectors included. From Cardco. Lifetime warranty.

34484 **$CALL**

CARTRIDGE EXPANDER

Plus $22.95 FAST LOAD OFFER!

Stop wearing out your computer and cartridges by endless swapping! The Navarone cartridge expander features 3 cartridge slots. Select any cartridge, or the "off" position if no cartridge is to be used. Reset the computer independently of the power switch. Cartridge slots are vertical for easy access—no blind fumbling behind the computer.

33227 Cartridge Expander **$22.95**

The cartridge expander is a great companion for the Epyx Fast Load cartridge—you can keep it permanently installed plus have two slots free for other cartridges!

34216 Fast Load Cartridge (Sug. Retail $34.95) **$29.95**

Fast Load ONLY $22.95 with purchase of cartridge expander!

DUST COVER and "EVERYTHING BOOK" SPECIAL OFFER

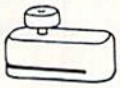
Get to know us by ordering this great dust cover for your VIC-20 or Commodore 64 and our catalog, "The Everything Book for the Commodore C-64 and VIC-20 Home Computers," for $2.95 (no extra shipping and handling charges). Cover is antistatic, translucent 8 gauge vinyl sewn to our exacting standards with reinforced seams. Discover the savings and easy shipping available from TENEX Computer Express!

$2.95

31627 Dust Cover and Catalog (J1D)

DISK DOUBLER

Double your storage by using both sides of your disks!



The single-sided disks you use on your Commodore actually have magnetic media on both sides. With our Disk Doubler you can punch a new write protect notch on the opposite side of the disk and use the "hidden" second side! This handy device will pay for itself the first few times you use it.

Sug. retail $14.95

37374 **ONLY $9.95**

From Your Friends At

TENEX Computer Express

We gladly accept mail orders!

P.O. Box 6578
South Bend, IN 46660

Questions? Call 219/259-7051

Ad J1D

SHIPPING CHARGES

ORDER AMOUNT	CHARGE
less than $20.00	$3.75
$20.00–$39.99	4.75
$40.00–$74.99	5.75
$75.00–$149.99	6.75
$150.00–$299.99	7.75
$300 & up	8.75

MasterCard VISA

NO EXTRA FEE FOR CHARGES

WE VERIFY CHARGE CARD ADDRESSES.

ORDER TOLL FREE

1-800-348-2778

BY BRUCE JAEGER

Scrambler

A Machine-Language Data Scrambler for the Commodore 64

The two basic ways of enciphering a message or data are substitution and transposition.

While most of us don't do a great deal of top-secret Defense Department work on our Commodore 64's, occasionally there is a need or desire to make our data files unreadable. Teachers may wish to have quiz answers on the same disk as the quiz, yet indecipherable to the bright students who know how to sneak into data files. Or you may wish to leave a message on a computerized bulletin board that only your friends "in the know" can read.

There are two basic ways of enciphering a message or other data: *substitution* or *transposition*. A *substitution* cipher is the familiar schoolboy code, in which different letters are made to stand for each other. (A very simple cipher, called the "Caesar Cipher," uses a number to determine the cipher letters. If our "key number" is 3, then all A's become D's, all B's become E's, etc. The word "COMPUTER" would be enciphered "FRPSXWHU".) There are many, many other substitution cipher methods.

A *transposition* cipher hides the message by scrambling the letters in a predetermined sequence (so the recipient can unscramble it). If our secret message is "THE BOSS INSPECTS ON TUESDAY," we might transpose the text by the following method:

```
T H E B O
S S I N S
P E C T S
O N T U E
S D A Y E
```

(That extra "E" on the end is to make the square come out even.) This is called a simple columnar transposition, and by reading down the columns, our enciphered text would read: TSPOS HSEND EICTA BNTUY OSSEE.

Most simple substitution and transposition schemes like the above are easily broken by experienced cryptologists, especially with the aid of computers. But we can make things pretty tough for the average person!

I chose to use the transposition method for "Scrambler." The program is in machine language for speed and extra security. Enciphering a long disk file could take forever in BASIC, and Scrambler transposes and retransposes the text several times, depending upon the user's keyword!

Scrambler transposes text a character string at a time, and can handle strings of any length up to the maximum 255 characters allowed by the computer. And Scrambler doesn't care what characters are in the string; they can be alphanumeric, color codes, cursor commands, etc. The keyword used to encode the message can also be of any length (except an empty or "null" string) and can even consist of non-printing characters.

To use Scrambler, first enter and save the BASIC loader below. When run, the program pokes the machine language program into the 64's memory starting at $C000 (49152 decimal). The memory area from $C200 to $C2FF (49664 to 49919) is used as a workspace, so don't store any other routines there.

Now load and run the short demo program. Note how a null string ("") is added to ke$ after the input statement. This is necessary only in text entered directly from input statements, and forces the 64's BASIC interpreter to relocate the string ke$ where the Scrambler machine code can look at it.

Scrambler transposes each message several times, according to the length of the keyword. Let's use the keyword "dog" as an example. Scrambler transposes the message four times, once by using the ASCII value of each character of the keyword as a key, plus a final time with the sum of the characters' values as a key. On decoding a message, it does everything in reverse.

Using Scrambler in Your Own Programs

Scrambler is called using the format:

SYS (49152),X,KEY$,TEXT$

Make X equal to zero to encode a message, and equal to one to decode.

KEY$ contains the keyword. As stated above, it must contain at least one character (it cannot be empty). You don't have to call it KEY$, as long it's in the second position.

TEXT$ contains the message to be scrambled. When Scrambler is finished, TEXT$ contains the scrambled text.

All of the above strings can also be string arrays. The following will also work: SYS (49152),X,KEY$(X), TEXT$(X). Again, the strings may have any name, as long as they are in the correct position. C

Recommended (and very interesting) reading:

Cryptanalysis for Microcomputers
Caxton C. Foster
Hayden Book Company, Inc.
5174-3

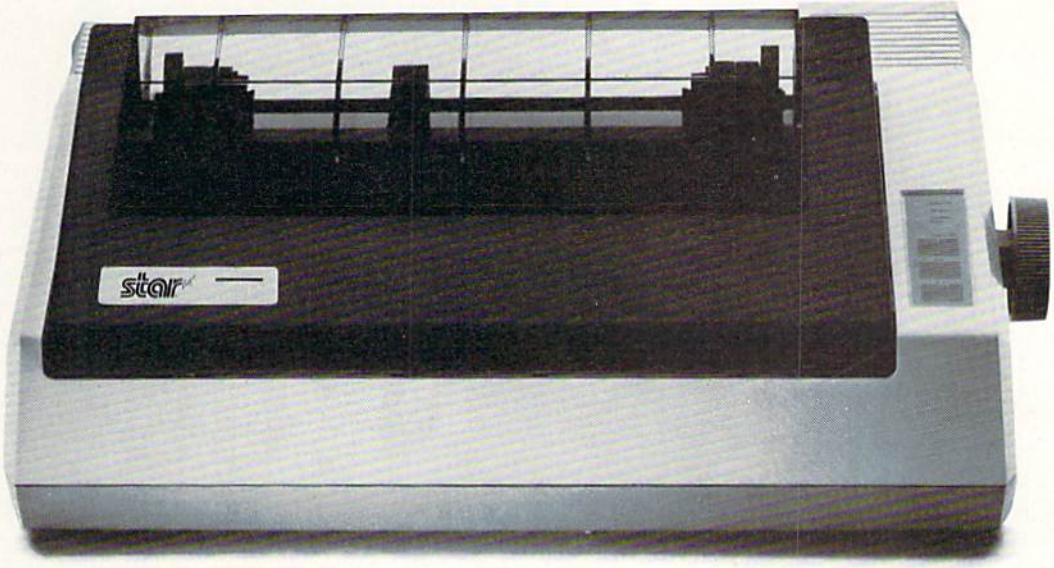
Before typing this program, read "How to Enter Programs" and "How to Use the Magazine Entry Program."

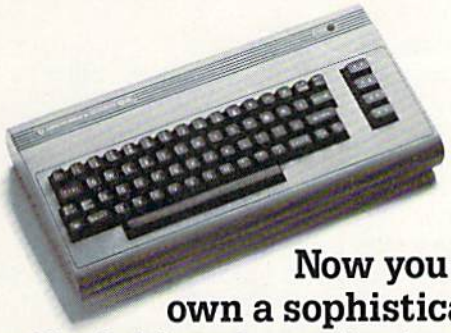
Scrambler: BASIC Loader

```
2000 FOR J=49152 TO 49420:READ A
     :POKE J,A:NEXT'GSKB
2010 DATA 32,1,193,32,234,192,165,156'BCMA
2020 DATA 133,155,165,251,133,253,165,
     252'BGFC
2030 DATA 133,254,32,234,192,165,158,
     240'BFHD
2040 DATA 3,76,210,192,76,187,192,
     169'BCGD
2050 DATA 0,162,0,157,0,194,232,
     208'BAAE
2060 DATA 250,96,32,31,192,160,0,
     177'BBKF
2070 DATA 251,201,32,208,2,169,160,
     153'BDBG
2080 DATA 0,194,200,196,156,208,240,
     96'BDQH
```

Continued on pg. 56

A BRIGHT NEW STAR FOR A COMMODORE TO PRINT BY





Now you can own a sophisticated, affordable printer that's made just for your C-64.™

Here's a dual-mode Star printer that delivers 120 cps draft and 30 cps *near-letter-quality.* With outstanding graphics ability. Interface cable. And friction and tractor feed. All standard.

And it's the best-selling printer for the best-selling home computer!

Star's made-for-Commodore® printer needs no add-ons. Just plug it into your computer and go to work. It's made at a price that can't be beat.

You've got the computer. Now get everything out of it. Get the Star that's labelled "C." There's no better mate made for a Commodore.

star® MICRONICS

Commodore and C-64 are trademarks of Commodore Business Machines, Inc. and/or Commodore Electronics Ltd.

```
2090 DATA 160,0,185,0,194,145,251,
     200'BCXI
2100 DATA 196,156,208,246,96,164,155,
     169'BFJB
2110 DATA 0,133,176,24,177,253,101,
     176'BDJB
2120 DATA 133,176,136,208,247,9,1,
     133'BCOC
2130 DATA 176,96,132,177,32,42,192,
     166'BDCD
2140 DATA 176,160,0,189,0,194,208,
     8'BASE
2150 DATA 24,138,101,176,170,76,107,
     192'BEIG
2160 DATA 145,251,169,0,157,0,194,
     200'BCGG
2170 DATA 196,156,208,231,164,177,96,
     132'BFWI
2180 DATA 177,32,31,192,160,0,169,
     1'BAHI
2190 DATA 153,0,194,200,196,156,208,
     246'BEIK
2200 DATA 166,176,160,0,189,0,194,
     201'BCMB
2210 DATA 1,240,8,24,138,101,176,
     170'BBCC
2220 DATA 76,156,192,177,251,157,0,
     194'BDDD
2230 DATA 200,196,156,208,231,32,64,
     192'BEGF
2240 DATA 164,177,96,160,0,177,253,
     9'BBCF
2250 DATA 1,133,176,32,98,192,200,
     196'BCRG
2260 DATA 155,208,242,32,77,192,32,
     98'BCAH
2270 DATA 192,96,32,77,192,32,135,
     192'BCCI
2280 DATA 164,155,136,177,253,9,1,
     133'BCRJ
2290 DATA 176,32,135,192,136,192,255,
     208'BFPL
2300 DATA 242,96,32,253,174,32,158,
     173'BDVC
2310 DATA 160,0,177,71,133,156,200,
     177'BDHD
2320 DATA 71,133,251,200,177,71,133,
     252'BEXF
2330 DATA 96,32,115,0,32,158,173,
     32'BAJF
2340 DATA 247,183,132,158,96'BSPF (END)
```

Scrambler Demo

```
1010 REM SCRAMBLER / UNSCRAMBLER'BVBA
1020 :'ABHU
1021 REM  CALL SCRAMBLER/UNSCRAMBLER'
     BADD
1022 REM  ROUTINE WITH:'BMRB
1023 REM  SYS 49152,0,K$,
     T$ (ENCODE)'BYTE
1024 REM  SYS 49152,1,K$,
     T$ (DECODE)'BYKF
1025 REM K$=KEYWORD  T$=TEXT STRING'BXXH
1026 :'ABHB
1030 POKE 53272,23'BIRX
1040 DIM TEXT$(20),KEY$(10)'BSPB
1050 FOR X=1 TO 10:READ KEY$(X)
     :NEXT'FNSD
1060 FOR X=1 TO 20:READ TEXT$(X)
     :NEXT'FOFF
1070 PRINT"[CLEAR,RVS,WHITE,SHFT S,
     SHFT C,SHFT R,SHFT A,SHFT M,
     SHFT B,SHFT L,SHFT E,SHFT R]
     "'BADI
1080 FOR X=1 TO 20:PRINT"[WHITE]
     "TEXT$(X):NEXT:GOSUB 1410'GTTI
1090 K=INT(RND(1)*10)+1
     :KEY$=KEY$(K)'GVBK
1100 PRINT"[HOME,GREEN,RVS] [SHFT K,
     SHFT E,SHFT Y,SHFT W,SHFT O,
     SHFT R,SHFT D]: [RVOFF]"KEY$'BEMC
1110 FOR X=1 TO 20'DESW
1120 SYS 49152,0,KEY$,TEXT$(X)'BVRB
1130 PRINT"[YELLOW]"TEXT$(X):NEXT
     :GOSUB 1410'DOVC
1140 PRINT"[HOME,YELLOW,RVS,SHFT U,
     SHFT N,SHFT S,SHFT C,SHFT R,
     SHFT A,SHFT M,SHFT B,SHFT L,
     SHFT E,SHFT D,RVOFF,SPACE10]
     "'BAJJ
1150 FOR X=1 TO 20'DESB
1160 SYS 49152,1,KEY$,TEXT$(X)'BVSF
1170 PRINT"[CYAN]"TEXT$(X):NEXT
     :GOSUB 1410:GOTO 1070'ETFI
1180 REM KEYS'BECD
1190 DATA DOG,CAT,ELEPHANT,COMPUTER,
     DISK DRIVE'BKXN
1195 DATA APRIL,AUTOMOBILE,TV,BOOK,
     KEY'BDOQ
1200 REM MESSAGES'BIFW
1210 DATA "[SHFT T]HIS IS A TEST.
     [SHFT T]HIS IS ONLY A TEST."'BAFG
1220 DATA "[SHFT F]OUR SCORE AND
     SEVEN YEARS AGO..."'BAAG
1230 DATA "[SHFT I]F [SHFT I] OWNED
     [SHFT H]ELL AND [SHFT T]EXAS,
     [SHFT I]'D RENT OUT"'BAFL
1240 DATA "[SHFT T]EXAS AND LIVE IN
     [SHFT H]ELL. [[SHFT P].[SHFT H]
     . [SHFT S]HERIDAN]"'BAEM
1250 DATA "ABCDEFGHIJKLMNOPQRSTUVWXYZ
     0123456789"'BAVK
1260 DATA "[SHFT C]ONDUCTOR,
      WHEN YOU RECEIVE A FARE,"'BAPL
1270 DATA "[SHFT P]UNCH IN THE
     PRESENCE OF THE PASSENJARE"'BATN
1280 DATA "[SHFT A] BLUE TRIP SLIP
     FOR AN EIGHT-CENT FARE"'BAMO
1290 DATA "[SHFT A] BUFF TRIP SLIP
     FOR A SIX-CENT FARE,"'BACO
1300 DATA "[SHFT A] PINK TRIP SLIP
     FOR A FIVE-CENT FARE,"'BAWG
1310 DATA "[SHFT P]UNCH IN THE
     PRESENCE OF THE PASSENJARE"'BATI
1320 DATA "[SHFT P]UNCH, BROTHER,
      PUNCH WITH CARE."'BAQH
```

```
1330 DATA "[WHITE]WHITE[YELLOW]YELLOW
     [GREEN]GREEN[CYAN]CYAN[L. RED]
     PINK"'BAPI
1340 DATA "[SHFT N]OTHING TICKLES
     THAT DOESN'T PINCH."'BAPK
1350 DATA "[SPACE19][[SHFT M]ONTAIGNE]
     "'BAYH
1360 DATA "[SHFT A]LWAYS DO RIGHT.
     [SPACE2,SHFT T]HAT WILL
     GRATIFY"'BANM
1370 DATA "SOME PEOPLE,
      AND ASTONISH THE REST."'BANM
1380 DATA "[SPACE21][[SHFT M]ARK
     [SHFT T]WAIN]"'BAEL
1390 DATA "[SHFT N]O PIETY CAN DELAY
     THE WRINKLES [[SHFT H]ORACE]
     "'BAKR
1400 DATA "### [SHFT E]ND OF [SHFT T]
     RANSMISSION ###"'BACF
1410 PRINT"[HOME,DOWN24,RIGHT4,SHFT P]
     RESS [RVS,SHFT S,SHFT P,SHFT A,
     SHFT C,SHFT E,RVOFF] [RVS,SHFT B,
     SHFT A,SHFT R,RVOFF] TO CONTINUE
     [HOME]"'BACO
1420 POKE 198,0:WAIT 198,1:GET A$
     :IF A$<>" "THEN 1420'HVMH
1430 RETURN'BAQA                    END
```

Scrambler/Unscrambler

```
100 REM DEMO PROGRAM FOR
    SCRAMBLER'BXCC
110 REM (ASSUMES MACHINE LANGUAGE'BXRD
120 REM  ROUTINE ALREADY LOADED)'BVLD
121 :'ABHX
130 PRINT CHR$(147):POKE 53272,23'DOCC
140 POKE 53280,0 :POKE 53281,0'CPLD
150 PRINT CHR$(5)'CDEB
155 TE$=""'BDKG
160 PRINT "[SHFT P]HRASE TO [SHFT S]
    CRAMBLE:"'BAKH
170 INPUT TE$'BDPD
175 IF TE$=""THEN END'EDHK
180 PRINT "[SHFT K]EYWORD:"'BACG
190 INPUT KE$'BDGF
195 KE$=KE$+"" : REM NOTE THIS
    OPERATION'DYVR
200 PRINT CHR$(18)"[SHFT C,SHFT L,
    SHFT E,SHFT A,SHFT R,SHFT T,
    SHFT E,SHFT X,SHFT T]:"'CEMF
210 PRINT TEXT$ : PRINT'CGMY
220 PRINT CHR$(158)'CFDA
230 PRINT CHR$(18)"[SHFT S,SHFT C,
    SHFT R,SHFT A,SHFT M,SHFT B,
    SHFT L,SHFT E,SHFT D]:"'CEGI
250 SYS 49152,0,KEY$,TEXT$'BSEF
260 PRINT TEXT$ : PRINT'CGME
270 PRINT CHR$(159)'CFEF
280 PRINT CHR$(18)"[SHFT D,SHFT E,
    SHFT C,SHFT O,SHFT D,SHFT E,
    SHFT D]:[SPACE2]"'CEEM
300 SYS 49152,1,KEY$,TEXT$'BSFB
310 PRINT TEXT$ : PRINT'CGMA
320 GOTO 150'BDFA                   END
```

BY IAN ADAM

Control Your Data

for the Commodore 64 and VIC 20

Power to the people!! Seize control of your data statements!! The data you type into your program is rightfully yours, to read as you wish!

Here are some simple programming tips that will give you the capability to select and read data statements under complete control. Data statements provide an extremely valuable method for entering information into a computer program. For example, they can be used to supply the results of repeated scientific experiments in a program that will analyze and display the resulting findings. Other very typical applications are information for graphics (e.g., sprites), machine language routines, or numbers to be interpreted as notes in a melody. Either numeric or string information can be included in a data statement, but it is important that the program be expecting the right kind.

When it first encounters a read statement, your BASIC program searches through the program lines until it finds a data statement, then takes the first available item. The second READ gets the second item, and so on. Successive read statements will access each item in turn, until all of the data has been read. After that, any further attempt to read will result in an out-of-data error, or a type-mismatch error. Your only choice then is the restore statement, which causes the program to read data beginning at the start of the listing again.

In a simple program, this procedure is great. Each of the items is read in turn, and dealt with as you specify. Very simple! As your programs become more complex, however, the limitations of this procedure will quickly become apparent. You have little control over which data you read.

A particular limitation occurs when you want to access one group of data items repeatedly. Suppose you want to play a short tune each time a child answers a question correctly in an educational program. First, you have to restore to the start of the program, then count how many data items there are before the ones you want. Each of these must be read into a dummy variable and thrown away. Then, and only then, are you ready to read your desired data.

In many instances, it would be very useful, for example, to be able to select one line in the program, and have subsequent data read beginning at that point. One of the following tips will show you how to do that, thus greatly increasing your programming flexibility. We will also look at some simple methods for avoiding errors.

The extent to which you need to take advantage of these suggestions will depend upon the nature and complexity of your program. For example, if your program uses only numeric data, you are unlikely to encounter a type-mismatch error. Some of the other problems could still crop up, however.

I would suggest, then, that you scan through them all, and decide which ones would be of the greatest benefit to you. I would also be interested in hearing about any other snappy ways you have found for avoiding the limitations of the system.

Out of Data Error

This error notice will appear when the program runs out of data items, but you are still instructing it to read. In a perfect world, of course, this would never happen, because you would always have exactly the right number of read statements to match your data. In the real world we inhabit, it is all too common!

Here is a very simple tip to avoid the error message: Add a line at the end of your program, with a line number large enough to ensure that it will always be last:

```
60000 DATA 9999
```

You can insert as many data lines as you want anywhere else in the program. When you want to read them, just check to be sure you haven't reached the end yet. For example:

```
100 READ A : IF A = 9999 THEN
    200
110 POKE 54273, A : FOR I = 1
    TO 250 : NEXT : GOTO 100
200 RESTORE
```

The beauty of this method is that you can add or subtract as many data items as you wish, without having to change anything else in your program. The program knows automatically that when it reaches 9999 there is no more data to be read. As a result, it doesn't try to read non-existent information, and you don't get an error.

Integrity of Data

Sometimes you have to ensure that every data item is absolutely correct. This occurs, for example, with machine language programs that are to be poked into memory. The slightest mistake could cause the program to fail, possibly locking up the computer so that you have to turn it off and on again, thus losing your program.

If you want to avoid this, simply use a checksum to verify the integrity of the data. For example:

```
100 FOR J = 49152 TO 49202
110 READ A : CH = CH + A
120 POKE J, A : NEXT
130 IF CH <> 5130 THEN STOP
```

In this example, CH is the checksum. It keeps a running total of all data items as they are read. When all have been read, the checksum variable must equal the previously calculated total. If they are not equal, then there must be an error in the data, so the program stops before any damage can be done.

Type Mismatch Error

This error can easily occur if you are mixing numeric and string data in the same program. This is common in adventure programs, where names of objects and their numeric point values

may be combined in the same line with other parameters. You can read numeric data into a string variable without difficulty. However, if you attempt to put a string value into a numeric variable, you will be chided with the error message.

The solution to this problem lies in first reading all data as strings, then converting back to a numeric value where appropriate. If a non-numeric variable is found when a number was expected, it will simply return a value of zero, which is better than an error message.

Try this method:

```
100 READ A$ : IF A$ = "9999"
    THEN 200
110 PV(J) = VAL(A$)
120 READ TR$(J) : J = J + 1 :
    GOTO 100
200 . . .
1000 DATA 20, SILVER SWORD
1010 DATA 50, GOLD CHARM
1020 DATA (etc.) . . .
60000 DATA 9999
```

This method will avoid giving the error message, although it won't correct any fundamental flaws in a program. What we have done is read in all items as string variables, to avoid errors. When we are expecting a number for point value, we convert it to variable PV using the VAL function. When we expect a string for the treasure variable TR$, we just read it directly. Notice that we also included the trick we discussed earlier to indicate the end of data. This way you can add more treasure items later, as long as you remember to dimension the array variables properly.

Restore to Specific Location

As we mentioned earlier, in a child's educational program, you sometimes want to instruct BASIC to begin reading data at a specific point—for example, where your "reward" melody is stored. What's really needed is a RESTORE TO 5000 instruction, or something similar. This would allow you to specify the line number to begin reading.

In the absence of that particular BASIC enhancement, here's something that's almost as good. BASIC maintains a pointer to the current data item; all you have to do is reset that pointer to a more appropriate location. But what to set it up to? Fortunately, BASIC also maintains another pointer to where it's currently working (sort of like moving your finger over the text as you read). This statement will reset the data pointer to wherever the statement is located:

```
A = PEEK(61) + 256 * PEEK(62):
POKE 66, A/256:POKE65,
A—256 * PEEK(66)
```

Locations 61 and 62 are the pointer to the current BASIC instruction, while 65 and 66 are the pointer for read. What this statement says to BASIC, in effect, is, "I want you to start reading *right here.*"

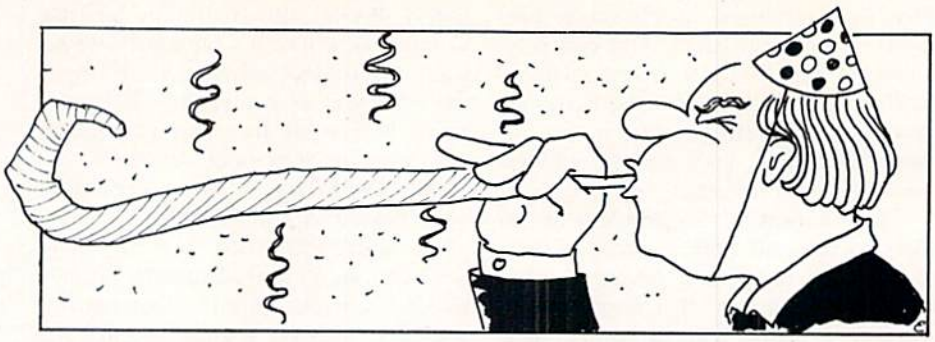
Now suppose you have several different tunes you want to play—one for a correct answer, one for an incorrect one, another for the ending, and so on. All you have to do is put the data items for each tune in a subroutine by themselves. The first line of the subroutine must reset the pointer, and the last will indicate end of data. Here's the whole thing pulled together:

```
100 ON B GOSUB 5000, 6000,
    7000, . . .
110 READ A : IF A = 9999
    THEN 200
120 POKE 54273, A : GOTO 110
200 . . .
5000 A = PEEK(61) + 256
     * PEEK(62) : POKE 66, A/
     256
5010 POKE 65, A—256
     * PEEK(66) : RETURN
     :REM TUNE FOR RIGHT
5020 DATA . . .
5980 DATA 9999
5990 :
6000 A = PEEK(61) + 256
     * PEEK(62) : POKE 66, A/
     256
6010 POKE 65, A—256
     * PEEK(66) : RETURN
     : REM TUNE FOR WRONG
6020 DATA . . .
6980 DATA 9999
6990 :
7000 . . .
```

The variable B is set to one for the right answer, two for the wrong answer, and so forth. Each subroutine is self-contained. It resets the data pointer, contains all the data for one tune, and even indicates automatically when the tune is finished. You could have as many subroutines—as many different tunes for different purposes—as you wish. C

BY BRUCE JAEGER

A Computer User's New Year's Resolutions



I will look under my chair wheels for diskettes before rolling away from my desk!

I will not turn up my stereo really loud, and then accidentally stick in a computer cassette.

I will remember to save that long program I just typed in, *before* I try to run it.

I won't put my ashtray on top of the disk drive vents.

I won't keep my tape head demagnetizer in the same *room* as my diskettes.

I will remember that using all of my strength on the joystick doesn't make the game play any better.

I promise I will *never* say, "But it will increase my productivity!"

Before turning up my monitor's brightness control, I'll see how thick the layer of dust is on the screen.

I will read the instructions before I try to run a new program.

I *will* make backups of my data disks.

I will always turn my computer off before sticking in a new cartridge.

Before complaining about a program at a user's group meeting, I'll make sure that its author is out of earshot.

Before cussing out the computer or the programmer when an input doesn't work, I'll look and see if my SHIFT/LOCK key is locked down.

I will somehow try to keep the ratio of sleeves to diskettes somewhat equal.

I will not use words like "hex," "dump," or "registers" in ordinary conversation—unless we're talking about witches, sanitary landfills or hot-air heating systems.

I promise I won't set a can of pop on top of the disk drive.

For that matter, this year I'll try not to spill anything on the disks either.

I will remember how dumb it is to eat fried chicken while working at the keyboard.

I will check my paper supply *before* I begin that really long printout.

I will untangle my cables.

I will turn down the volume when playing "World War III/Alien Armageddon Shoot-Out" at 3:00 a.m.

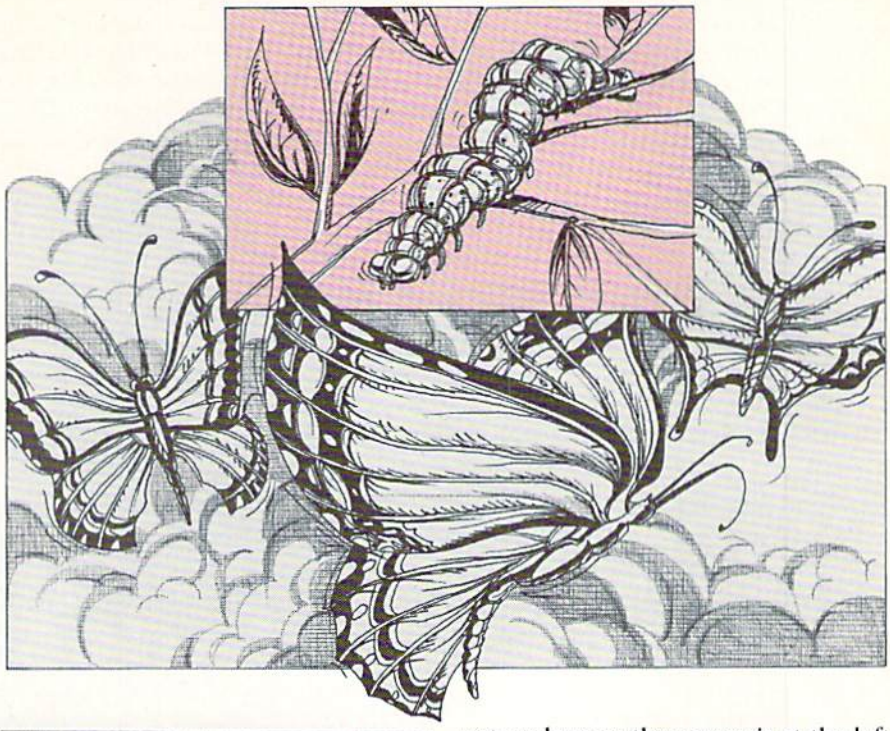
Before I call a new BBS number, I'll call "voice" first to make sure it isn't really somebody's poor Aunt Harriet.

I will keep the dog out of the computer room. C

BY LOUIS F. SANDER

Datastater

A Utility for the Commodore 128, Plus/4 and Commodore 16



Convert machine language, sprites, or other sections of memory to BASIC data statements—with ease.

BASIC programmers often add machine-language (ML) routines to their programs to take advantage of ML's speed or other attributes. The most common way of adding the ML is to convert it to data statements, which are then read and poked by a simple loop. This method has much to recommend it, but making up the data statements is a tedious exercise in hex-to-decimal conversion, and in typing in series of numbers.

Datastater is a full-featured program for converting machine language, sprites, or other sections of memory, to BASIC data statements. It comes in two versions: one for the Commodore 128, and another for the Plus/4 and Commodore 16. The program takes advantage of several features of those machines: the built-in machine-language monitor, the hex-to-decimal converter function, and the escape sequences used in screen editing. The data statements produced are in hex, which makes for easier typing and better understanding.

Running Datastater is simple, and the program even includes brief instructions. But before using it the first time, you should read the rest of this article. You should also be familiar with using the MONITOR command to call up your machine-language monitor, and with using that monitor's "M" command to dump memory to the screen.

Right now, select and enter the Datastater version appropriate for your machine. As you type it, be particularly careful with quotation marks and semicolons, since they are critical to the program's operation, and since it's easy to misplace them. When you've finished typing, save Datastater and get ready to give it a test.

Your first step is to decide what area in memory you want to convert to data statements. Get into the monitor by typing MONITOR ⟨RETURN⟩, then use the "M" command to display the desired memory range. Since Datastater manipulates the screen display, it will convert a maximum of 16 monitor lines at a time, which is equivalent to 128 bytes of ML ($80 bytes in hex). If you need to convert more, just break the project into manageable parts.

When the desired memory range is displayed on the screen, use the monitor's "X" command to exit back to BASIC. The memory dump display, of course, will remain on the screen. Using your cursor-down key, scroll that display upward until its first line is on the top line of the screen.

Then move your cursor until it's on the line just below the last monitor line you want to convert to data, in the leftmost screen position. It doesn't matter if that line is blank or not, as long as the cursor is at the left margin.

To initiate a conversion, just type the word RUN, plus a colon, and then press RETURN. (The colon is there in case the cursor is not on a blank line). By the way, when you run Datastater, it checks to see that a monitor dump is actually on the top screen line; if it isn't, you're shown a series of instructions.

In a normal run, you'll be prompted for a First Line#, and you should respond with the number you want to assign to your first data line. If you respond by pressing RETURN without entering a number, Datastater will choose a line number equal to the decimal value of the ML's starting address, since that's the most logical number to use.

After choosing a starting line number, you'll be prompted for a line number interval. If you enter nothing and hit RETURN, the interval defaults to eight, which retains the correspondence between the line number and the address of the first data item in the line.

As soon as you choose an interval, you'll see the monitor lines turn into BASIC data lines. After a short wait, some other BASIC lines will appear on the screen.

The first one is a loop that computes the checksum of the bytes you

have converted. (A checksum is simply the numerical total of all the bytes.) If your data statements are ever typed from a listing, the typist will appreciate having this line, since it can be used to eliminate errors.

The next line is a FOR-NEXT loop that will poke your ML into memory. Datastater determines the start and end of the loop, which saves effort on your part, and reduces the chance of error. Notice that while your ML data is in hex, the loop is in decimal. Smart people, these computers.

And finally, there are two REM lines showing the number of data items, the checksum, and the memory range of the ML, in decimal as well as hex. This information can be useful to you later, so Datastater gives it to you now, in a form that is easily preserved.

After all the lines have been printed, Datastater terminates with the cursor flashing in the "home" position. Your BASIC lines have been printed on the screen, but have not yet been added to the program in memory. You do that yourself, by pressing the RETURN key over each program line. If you don't want the extra lines, use your cursor keys to skip over them. When you're finished with the RETURN key, your data lines, plus the checksum, poker and REM lines, are in memory along with Datastater. You can use the DELETE command to get rid of Datastater itself, which will leave you with the lines from Datastater's output. You can save them to disk or printer, where they can readily be added to the balance of your BASIC program.

That's all there is to Datastater, a very handy program to keep around the house. I've used similar utilities since the days of the Commodore PET, and I've found that I need them infrequently, but when I do, they're worth their weight in gold. C

Before typing these programs, read "How to Enter Programs."

C128 Datastater

```
130 IF PEEK(1024)<>62 THEN 330
140 NL=PEEK(235)-1:IF NL>16 THEN
    PRINT CHR$(18);"TOO MANY LINES!!"
    :END
150 E$=CHR$(27):CL$=E$+"@":BL$=E$+"J"
    :ER$=E$+"Q"
160 A$=","+CHR$(29)+CHR$(29)
    :FOR J=1 TO 7:CO$=CO$+A$:NEXT
    :CU$=CHR$(145)
170 DA$="[SPACE6]"+E$+"ADATA"+E$+CHR$
    (67)
180 FOR J=1 TO 4:K=PEEK(1025+J)
    :K=K-64*(K<7):BA$=BA$+CHR$(K):NEXT
190 HI$=CHR$(PEEK(1025))
    :HI=65536*VAL(HI$):BA=HI+DEC(BA$)
    :BA$=HI$+BA$
200 EA=BA+8*NL-1:EA$=HI$+HEX$(EA-HI)
    :PRINT CU$;CL$
210 FL=BA:PRINT CU$;"FIRST LINE#";
    :INPUT FL
220 IN=8:INPUT"[SPACE2]INCREMENT";IN
    :PRINT CHR$(18);"WORKING";
    CHR$(19);
230 FOR J=1 TO NL:LN=FL+IN*(J-1)
    :PRINT TAB(30);ER$;BL$;DA$;BL$;LN;
    TAB(13);CO$:NEXT
240 FOR J=1 TO NL:FOR K=0 TO 7:BY$=""
    :FOR L=0 TO 1:M=PEEK(1035+40*(J-1)
    +3*K+L)
250 M=M-64*(M<7):BY$=BY$+CHR$(M):NEXT
    :CK=CK+DEC(BY$):NEXT:NEXT
260 PRINT CL$;LN+IN;"FORJ=1TO";8*NL;"
    :READK$:L=DEC(K$):CS=CS+L:NEXT
    :RESTORE:IFCS<>";CK;"THENSTOP"
270 PRINT LN+2*IN;"FORJ=";BA;"TO";
    BA+8*NL-1;":READK$:POKEJ,DEC(K$)
    :NEXT"
280 PRINT LN+3*IN"REM"NL*8"BYTES.
    CHECKSUM ="CK
290 PRINT LN+4*IN"REM $";BA$;"-$";
    EA$" /";BA;"-";EA
300 POKE 842,19:POKE 208,1:END
310 :
320 REM INSTRUCTIONS
330 CD$=CHR$(17):PRINT CHR$(147);CD$;
    "THIS CONVERTS C-128 MONITOR
    MEMORY"
340 PRINT"DUMPS TO BASIC DATA LINES.
    TO USE IT:"
350 PRINT CD$;"1. USE THE MONITOR TO
    DUMP THE AREA OF"
360 PRINT"[SPACE3]INTEREST ONTO THE
    SCREEN, THEN EXIT."
370 PRINT CD$;"2. CURSOR DOWN UNTIL
    THE TOP LINE OF"
380 PRINT"[SPACE3]THE MEMORY DUMP IS
    ON THE TOP LINE"
390 PRINT"[SPACE3]OF THE SCREEN."
400 PRINT CD$;"3. PUT THE CURSOR JUST
    BELOW THE LAST"
410 PRINT"[SPACE3]DUMP LINE YOU WANT
    TO CONVERT."
420 PRINT CD$;"4. TYPE 'RUN
    :' AND PRESS RETURN."
430 PRINT CD$;"5. RESPOND TO THE
    PROMPTS. DEFAULTS ARE"
440 PRINT"[SPACE3]BEGINNING ADDRESS,
     INCREMENT OF 8."
450 PRINT CD$;"6. ADD THE NEW DATA
    LINES BY"
460 PRINT"[SPACE3]SUCCESSIVELY
    PRESSING RETURN."
```

END

Plus/4, C16 Datastater

```
530 IF PEEK(3072)<>62 THEN 740
540 NL=PEEK(205)-1:IF NL>16 THEN
    PRINT CHR$(18);"TOO MANY LINES!!"
    :END
550 E$=CHR$(27):BL$=E$+"J":ER$=E$+"Q"
560 A$=","+CHR$(29)+CHR$(29)
    :FOR J=1 TO 7:CO$=CO$+A$:NEXT
    :CU$=CHR$(145)
570 DA$="[SPACE6]"+E$+"ADATA
    "+E$+CHR$(67)
580 FOR J=1 TO 4:K=PEEK(3072+J)
    :K=K-64*(K<7):BA$=BA$+CHR$(K):NEXT
```

```
590 BA=DEC(BA$)
600 EA=BA+8*NL-1:EA$=HEX$(EA)
    :PRINT CU$;ER$
610 FL=BA:PRINT CU$;"FIRST LINE#";
    :INPUT FL
620 IN=8:PRINT ER$;"[SPACE2]
    INCREMENT";:INPUT IN
    :PRINT ER$;"WORKING...";CHR$(19);
630 FOR J=1 TO NL:LN=FL+IN*(J-1)
    :PRINT TAB(30);ER$;BL$;DA$;BL$;LN;
    TAB(13);CO$:NEXT
640 FOR J=1 TO NL:FOR K=0 TO 7:BY$=""
    :FOR L=0 TO 1:M=PEEK(3083+40*(J-1)
    +3*K+L)
650 M=M-64*(M<7):BY$=BY$+CHR$(M):NEXT
    :CK=CK+DEC(BY$):NEXT:NEXT
660 PRINT ER$;LN+IN;"FORJ=1TO";8*NL;"
    :READK$:L=DEC(K$):CS=CS+L:NEXT:";
670 PRINT ER$;"RESTORE:IFCS<>";CK;
    "THENSTOP"
680 PRINT ER$;LN+2*IN;"FORJ=";BA;"TO";
    BA+8*NL-1;":READK$:POKEJ,DEC(K$)
    :NEXT"
690 PRINT ER$;LN+3*IN"REM"NL*8"BYTES.
    CHECKSUM ="CK
700 PRINT ER$;LN+4*IN"REM $";BA$;"-$";
    EA$" /";BA;"-";EA
710 POKE 1319,19:POKE 239,1:END
720 :
730 REM INSTRUCTIONS
740 CD$=CHR$(17):PRINT CHR$(147);CD$;
    "THIS CONVERTS BASIC 3.5 MONITOR
    MEMORY"
750 PRINT"DUMPS TO BASIC DATA LINES.
    TO USE IT:"
760 PRINT CD$;"1. USE THE MONITOR TO
    DUMP THE AREA OF"
770 PRINT"[SPACE3]INTEREST ONTO THE
    SCREEN, THEN EXIT."
780 PRINT CD$;"2. CURSOR DOWN UNTIL
    THE TOP LINE OF"
790 PRINT"[SPACE3]THE MEMORY DUMP IS
    ON THE TOP LINE"
800 PRINT"[SPACE3]OF THE SCREEN."
810 PRINT CD$;"3. PUT THE CURSOR JUST
    BELOW THE LAST"
820 PRINT"[SPACE3]DUMP LINE YOU WANT
    TO CONVERT."
830 PRINT CD$;"4. TYPE 'RUN
    :' AND PRESS RETURN."
840 PRINT CD$;"5. RESPOND TO THE
    PROMPTS. DEFAULTS ARE"
850 PRINT"[SPACE3]BEGINNING ADDRESS,
     INCREMENT OF 8."
860 PRINT CD$;"6. ADD THE NEW DATA
    LINES BY"
870 PRINT"[SPACE3]SUCCESSIVELY
    PRESSING RETURN."
```

END

NEW LANGUAGE BREAKS OLD RULES. GIVES PROGRAMMERS POWER, SPEED AND SIMPLICITY.

Try this remarkable language, PROMAL™, for 30 Days AT NO RISK and...

We think you'll be thrilled with this breakthrough system when you discover its power, ease of use, and dazzling performance on your IBM PC, Apple IIe/IIc, or Commodore 64. **But we don't expect you to accept our claims for PROMAL without proof,** so we invite you to explore the power of PROMAL on your own during our 30-day trial period.

Broken Rules

Now that PROMAL 2.0 has broken the rules, a structured language doesn't have to be slow, unwieldy and difficult to use. PROMAL is fast, elegant, and simple.

What Is PROMAL?

PROMAL stands for **PRO**grammer's **M**icro **A**pplication **L**anguage. But PROMAL is more than a high-level language, it's a total structured programming development system with a fast, one-pass compiler, a versatile full-screen editor, plus an integrated machine-language subroutine library. And for APPLE and Commodore systems it includes a DOS-like system "Executive."

Better By Design

PROMAL was designed from "scratch" for **optimum performance** and ease of use on microcomputers. It has a simplified syntax with no awkward terminators like ";" or "}" and **indentation is part of the syntax,** so structuring your code is natural and easy. Just compare PROMAL with BASIC in this example:

Equivalent Program Segments

```
--------PROMAL--------
REPEAT
   PROMPT_AT 5,24, "Add/Chg/Quit?"
   IF Reply = 'A'
      ADD Item
      New_Items = New_Items + 1
   ELSE IF Reply = 'C'
      CHANGE Item
UNTIL Reply = 'Q'
```

```
--------BASIC--------
11910 REM
11920 CL = 5:LN = 24:PR$ = "Add/Chg/Quit?"
11925 GOSUB 9490:REM GET REPLY
11930 IF RP$<>"A" THEN 11950
11940 I9 = IT:GOSUB 10100:REM ADD
11945 NI = NI + 1:GOTO 11920
11950 IF RP$<>"C" THEN 11970
11960 I9 = IT:GOSUB 6050:REM CHG
11970 IF RP$<>"Q" THEN 11920
```

PROMAL is readable and understandable. You see the logic from the structure. And PROMAL lets you call procedures by name – so no more GOSUBs. But there's more.

Slick Editor

Editing your source is a snap with the specially-designed and integrated **full-screen Editor** – it not only helps you structure your program, it **even finds compilation errors** – automatically.

Quick Compiler

The compiler is a **lightning-fast,** one-pass, recursive descent design. On the IBM PC it crunches source to object at **2000 lines per minute,** and it's equally impressive on the Apple and C64. And your PROMAL source code is **portable from machine to machine.** That means your source can be used on all PROMAL target machines.

Run-Time Speed Demon

PROMAL blows away Apple II and C64 languages from BASIC and PASCAL to FORTH. (Send $3 for a copy of our full benchmark report.) It's **2000% faster** than BASIC. And on a normal IBM PC, the native 8088 code from **PROMAL beat Turbo Pascal 3.0** by 10% on the standard sieve benchmark!

DOS For Those Without

If you don't have a real "DOS," then PROMAL gives you a **true operating system** environment with the built-in operating system Executive. (See box.)

Outside Opinion

Naturally we're enthusiastic about PROMAL, **but here's what other programmers are saying:**

"Excellent... an ideal development system.... Well done indeed!"
M. T. V.
Naperville, Ill.

"I am... so amazed by PROMAL... I cannot believe the high degree of excellence of this entire package."
C. P., Ph.D.
Ridgeway, New York

"I don't know that I've ever seen a [system] as thoughtfully designed and as skillfully executed as PROMAL. Its logic and ease of programming are truly remarkable. Its speed of execution is phenomenal... congratulations."
E. C. R.
Alexandria, VA

Safety In Numbers

SMA, Inc. has been satisfying customers (over 100,000) since 1982 with innovative microcomputer products. Now you can join our thousands of satisfied PROMAL users, by trying it today.

Try It For 30 Days On Us

Send us some bucks and we'll send you PROMAL on trial for 30 days. **If for any reason whatsoever you are not satisfied, just send it back** for a quick refund of your purchase price. No questions asked. No risk.

How To Order.

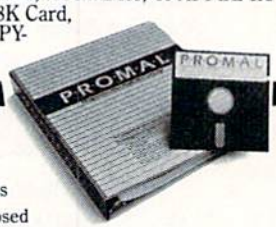
Call TOLL-FREE to order with your credit card or use the handy order form below to send in your check or money-order for your 30-day trial. Don't wait, you deserve the power of PROMAL today!

1-800-762-7874

In NC: 919-878-3600

Systems Management Associates, Inc.
3325 Executive Drive, Dept. C-1
Raleigh, North Carolina 27609

PROMAL runs on IBM PC/PCjr with 192K, Commodore 64/128, APPLE IIc, or APPLE IIe with 80 Col. 128K Card, and is NOT COPY-PROTECTED.



PROMAL 2.0 FEATURES

COMPILED LANGUAGE

- Structured indentation syntax
- No line numbers or terminators
- Long variable names (31 characters)
- Global, Local, & Argument variables
- Byte, Word, Integer & Real data types
- Decimal or Hex number types
- Functions & Procedures with passed arguments
- Predefined DATA of any type
- Multi-Dimensional Arrays (any type)
- Strings & pointers
- Control Statements: IF, IF-ELSE, WHILE, FOR, CHOOSE, REPEAT-UNTIL, BREAK, NEXT, INCLUDE, ESCAPE, REFUGE
- Bit-operators, shifts, type casts
- Variables at any memory location
- Simple Machine Language interface
- Recursion supported
- Program chaining and overlays (IMPORT/EXPORT)
- Separate compilation of modules
- Load and run relocatable M/L programs
- Compile errors trapped for Editor

EXECUTIVE (APPLE II & C64 Only)

- Command driven, with line editing
- Multiple user programs in memory at once
- Function key definitions
- Progam abort and pause
- Prior command recall
- I/O Re-direction & batch jobs
- "DOS"-like commands: COPY, RENAME, DELETE, display FILES, TYPE, HELP, etc.
- Memory MAP, SET, and display commands

EDITOR

- Full-screen, cursor driven
- Function key controlled
- Line insert, delete, search
- String search and replace
- Block copy, move, delete & file read/write operations
- Auto indent, undent support

LIBRARY

- 50 Resident Machine-language commands
- Call by name with arguments
- String handling (9 routines)
- Re-directable I/O (STDIN & STDOUT)
- Formatted numeric output
- Decimal & Hexadecimal I/O
- Block fill/move/read/write
- Cursor control & line editing
- Data type conversion
- Random number function
- Real function support (in PROMAL): ABS, ATAN, COS, EXP, LOG, LOG10, POWER, SIN, SQRT, TAN
- Modem device support & much more

Order Form for PROMAL 30-Day Trial!

My system is (check one)
☐ IBM PC/100% compatibles ☐ APPLE IIc/IIe
☐ COMMODORE 64/128

Please RUSH me:
☐ PROMAL Developer's System – Compiler, Editor, Library, Demo disk, 280-page manual, (Plus Executive for Apple and C-64) and stand-alone program generation (no royalties).
$99.95 + 5.00 Shipping & Handling
☐ End-User System for Apple IIc/IIe and Commodore 64/128 – all features of Developer's Version except stand-alone program generation (Executive needed for program execution)
$49.95 + 5.00 Shipping & Handling
☐ Graphics ToolBox (Apple/C64 only) – 20 routines for hi-res graphics: windows, clipping, text-on-graphics using scaled, rotated, user-defined fonts.
$29.95 + 2.50 Shipping & Handling

Please charge my
☐ Visa
☐ MasterCard
☐ American Express
☐ My check is enclosed

Card Number _____
Signature _____ Exp. Date _____
Name _____
Address _____
City _____
State _____ Zip _____

NC residents add 4½% sales tax.
Foreign orders add $20.00 additional shipping.

PROMAL is a trademark of Systems Management Associates, Inc. Turbo Pascal is a trademark of Borland International, Inc.

is

THE MISSING LINK!

From magazine...to diskette...to disk drive

Without LOADSTAR you are probably not getting the full benefit from the magazine that you are reading at this moment. Tell the truth now.... Have you typed in all the program listings? Of course not. Your time is worth much more than that. On the other hand, it is a shame to miss out on all the rich treasures that are available to you. The solution is LOADSTAR....

LOADSTAR is published monthly on a double-sided diskette. It contains the programs from Commodore's own magazines with lots more.... There are bonus programs, tutorials, articles, subscriber feedback, reviews, and music.

FOR VISA® MasterCard **CALL TOLL-FREE 1 (800) 831-2694 8-5 CENTRAL TIME**

In Louisiana call (318) 868-7247

If Loadstar fails to meet your expectations in any way, you can return the disk for a full refund.

"If you can afford only one disk service, make it Loadstar. If you can afford more than one, make sure Loadstar is one of them. You'll soon wonder why you bothered with the others!"

Stephen S. Leven, Sept./Oct. Commodore Microcomputers, page 32

to order by mail send to:

LOADSTAR
P.O. Box 30007
Shreveport, LA
71130-0007

- ☐ $9.95 for the current LOADSTAR disk companion to Power Play
- ☐ $9.95 for the current LOADSTAR disk companion to Commodore Microcomputers

OR

- ☐ $39.95 for the next 6 issues of LOADSTAR.
- ☐ $69.95 for the next 12 issues of LOADSTAR.

Name _____

Address _____

City/State _____ Zip _____

Visa/MC# _____ exp.dt _____

These rates also apply to Canada and Mexico — Write for overseas rates

Control Your World with Your Commodore Computer

by Jerry Houston

"Real computing" describes a computer's ability to communicate with the outside world. A computer might read a temperature sensor, for example, and decide to turn on the air conditioning. In another application, it might read a variety of meteorological sensors, and print a report to be used for a weather broadcast.

A new type of computer peripheral makes it possible for ordinary people—without formal training in digital interfacing—to take advantage of this "real-world" power of computers. These peripherals are known as Analog/Digital Data Acquisition and Control Systems, which shortens conveniently to "ADC".

Many people have turned to the Commodore 64 for "real-computing" power, because the 64 is very capable, offers reliable computing power at a modest cost, can support RS-232 serial communications, and is easy to maintain. The 64 also contains an easily accessed system clock, important for many projects that require the computer to handle timing.

Although the user port of the Commodore 64 doesn't support *true* RS-232 communication with 12V signals, it does provide TTL (transistor-transistor-logic) communications using all the parameters that are standard to RS-232. Some manufacturers of ADC's have taken this into consideration, and have provided the option for their equipment to work directly with the TTL lines available at the 64's user port. Other RS-232 equipment can also be used if the computer is equipped with an RS-232 adapter.

Analog/Digital Differences

Much information from the outside world is analog in nature. That is to say, it is expressed as a value on a continuous scale of some kind. Temperatures, pressures, distances, speeds, and directions are good examples. Analog information can be expressed easily as a voltage by any of a number of different sensors. A voltage, of course, is another analog value, and one that has significance to electronic devices.

Most computers in use today, including all the Commodore models, are digital computers. They can deal with digital values (which are expressed as bits that are turned either on or off), but not directly with analog values. In other words, they can monitor the presence or absence of a voltage, but not measure that voltage. A conversion must be made that will express analog values as digital information in order for a computer to be able to make any sense of it.

Applications

I have nothing against traditional uses for computers, such as word processing or database management—in fact, I work with both nearly every day. I just feel a certain excitement when a computer does something "real." I've spent the last month of weekends developing an environmental monitoring system for use on semi-submersible oil-drilling rigs, and it's a thrill to watch the system responding to temperature sensors, a wind-direction sensor, an

electronic barometer, and a wave-rider buoy that measures the height of waves. The system computes all sorts of statistics, writes data files to a diskette, prints permanent reports, and warns the weather observer of any conditions that appear to threaten the safety of the rig or its crew. This is all accomplished with relatively inexpensive equipment, and runs on a computer no more sophisticated than a Commodore 64.

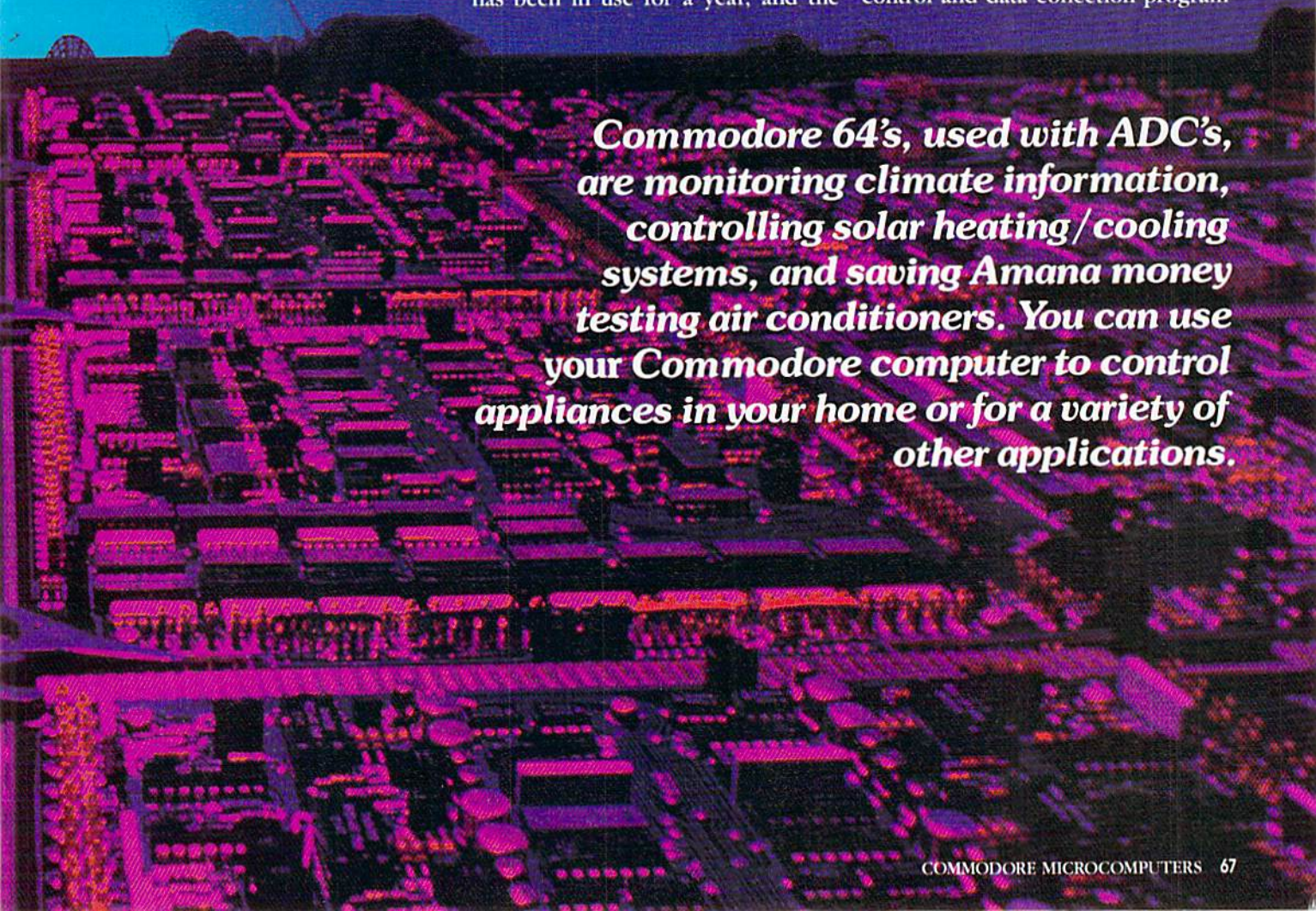
Here are some applications that are actually in use with Commodore computers.

Michael C. Head, an architect in Ojai, California, has designed a state-of-the-art solar home that includes solar heating and air-conditioning of the house, and solar heating of a pool and hot-tub. The primary requirements for operating the system are the accurate measurement of temperatures at 16 separate locations, and control of a number of fans and vents. A large plenum chamber under the house stores solar heat until needed.

An arboretum in the Bloedel Reservation on Bainbridge Island, Washington, monitors climate information. This is a home of exotic plant species from all over the world, and accurate weather data is essential in determining whether a particular species might survive there. The system they use includes a Commodore 64, modified somewhat to run on a car-type battery. The battery is kept charged normally by AC power, and is able to operate the system for up to two days following a complete power failure. Measurements are taken from two sensors each for rainfall, solar radiation, air temperatures, dewpoint, and soil temperatures, along with wind speed and direction. Once-a-minute readings are averaged each hour, and at midnight the computer writes the 24 hourly readings for each parameter to a disk file for later processing. Reports are produced for hourly, daily, and monthly averages of all readings, as well as accumulated totals. Says Richard Hinshaw, meteorological technician and designer of the project, "The system has been in use for a year, and the only failure of the Commodore was attributed to a nearby lightning strike."

When Michigan winters are really cold, one resident supplements his groundwater heat pump with heat from a 200-gallon wood-fired boiler, with everything supervised by a VIC 20. When the cold is extreme, the boiler is fired up and its hot water is passed through a heat exchanger that is common to the heat pump. Using an ADC, the VIC provides all the monitoring and controlling power needed for automatic operation.

A Brigham Young University professor uses his CBM-2001 (PET) computer to automate his catalytic-converter research. When he enters his lab in the morning, the results of a night's experiments are waiting for him. The ADC varies the flow rates of four gases into the test chamber according to programmed instructions, and injects a reaction sample into his gas chromatograph every ten minutes. The chromatograph output is digitized by the ADC twice per second, and analyzed by the PET to produce a printout of composition. The control and data collection program



Commodore 64's, used with ADC's, are monitoring climate information, controlling solar heating/cooling systems, and saving Amana money testing air conditioners. You can use your Commodore computer to control appliances in your home or for a variety of other applications.

is written in BASIC.

Surrounding Atlanta, Georgia, is an agricultural area that is characterized by extreme variations in local weather. The sun can be shining brightly in Atlanta, while a torrential downpour wipes out crops a few miles away. As a result, station WSB-TV in Atlanta, uses the Commodore 64 as an integral part of their weather reporting system. Thirteen ADC's in the surrounding areas are connected to weather sensors, and communicate with the 64 at the station's headquarters via modem. Every 15, 30, or 60 minutes, depending on the operator's instructions, the 64 calls each of the weather stations and downloads the current meterological data. When weather broadcasts are made, information is available that is complete, accurate, and up-to-the-minute for each suburb. Says Herbert Gilbert, Chief Engineer, "The reason the system's on the air is because there are a lot of little communities near here without access to their own weather information. This new system makes it possible for us to provide that coverage."

Finally, Amana, manufacturer of appliances that include refrigeration and air-conditioning equipment, is now using several Commodore 64's equipped with ADC's to optimize testing of their room air conditioners. Previously, a test booth was maintained at 90 degrees F, and special equipment was used to monitor the energy consumption of each air-conditioner as it lowered the temperature of the test booth by a certain amount, typically over a 20-minute period. Now the equipment is tested on the factory floor, the tests require only two minutes and test data is stored to a 1541 disk file for processing by another Commodore 64 in the company's DP department. By testing the equipment in one-tenth the time that was required before, energy savings have been significant.

Although most of these applications sound professional, keep in mind that the developers of the projects are in most cases *not computer professionals!* They're ordinary folks, with skills in their own specialties, and enough knowledge of the Commodore 64 to program in BASIC. Actually, most ADC's respond to single-byte commands sent to the RS-232 port, and could care less what language sent the byte there. Where high-speed operation is not a consideration, BASIC is a fine language. For those projects where processing speed IS critical, the program may be compiled or written in machine language.

In addition to the applications we just discussed, many people are interested in computer automation of their homes, and ADC's provide a

Table 1. ADC's for Commodore Computers

Model	Manufacturer	Features	Cost
ADC-1 Data Acquisition & Control System	Remote Measurement Systems, Inc. 2633 Eastlake Ave E Ste 206 Seattle, WA 98102 (206) DATA-255	16 analog inputs of 12-bit resolution plus sign, 4 digital inputs. 6 controlled outputs, transmitter for BSR R/C modules. Interfaces with Commodore user ports, no RS-232 adapter required. D/A sampling rate up to 100/sec. CMOS for low power use—powered by user port.	$449 standard model. Many options available.
BUSSter D16R	Connecticut Microcomputer Inc. 150 Pocono Rd Brookfield, CT 06804 (203) 354-9395	16 analog inputs of 8-bit resolution, 100-byte buffer for storing readings. D/A sampling rate 100/sec. RS-232 or IEEE-488 available.	$495 for model described. Devices also available for digital input/output, and analog outputs, $495 to $695.
MTM1	Software Science PO Box 44232 Cincinnati, OH 45244 (513) 561-2060	8 analog inputs of 8-bit resolution, 12 digital outputs. RS-232. D/A sampling speed dependent on baud rate, up to 480/sec.	$249 circuit board, $89 power supply, $129 enclosure.
ONE/05	Tarus Computer Products, Inc. 1755 Woodward Dr Ottawa, ONT K2C OP9 (613) 226-5361	16 analog inputs of 12-bit resolution, 16 digital I/O channels. Has built-in amplifier for direct reading of thermocouples. Z-80A microprocessor controlled. RS-232.	$2595 (US) for standard version. $2975 with two analog outputs and 4 counter/accumulators.
PL-1000 Measurement & Control System	Elexor Associates PO Box 246 Morris Plains, NJ 07950 (201) 299-1615	16 analog inputs of 12-bit resolution plus sign, 16 digital outputs. CMOS for low power use. Stand-alone operation possible with built-in BASIC interpreter and 8K RAM. Unit can accommodate up to 2 I/O boards (optional) for additional channels. RS-232.	$899 basic system. Expansion boards $329–$499. Many system options.
Q-3024 Remote Data Collection System	Quasitronics 211 Vandale Drive Houston, PA 15342 (800) 245-4192	2 single-ended analog inputs, 4 digital outputs. A/D resolution 1 in 5000 or 1 in 20000 using BCD digits, sample rate is 7.5/sec. RS-232.	$495
WB-31 "White Box" Interface	Omega Engineering, Inc. One Omega Drive Stamford, CT 06907 (203) 324-FLOW	2 analog inputs of 12-bit resolution, 4 digital outputs. A/D sample rate 7.5/sec. RS-232.	$395. Other models to $2000+. Omega has a fine catalog of sensors, that is well worth requesting.
12232 Data Acquisition & Control System	Starbuck Data Company PO Box 24 Newton Lower Falls, MA 02162 (617) 237-7695	Microprocessor controlled, 8 analog inputs of 12-bit resolution, 8 digital inputs, 8 digital outputs. RAM to store 2000 data points of burst data.	$690. Analog 8-bit 8032 version is now $390.
Hard Wire System, Wireless System	Jance Assoc., Inc. PO Box 234 East Texas, PA 18046 (215) 398-0434	Compute security systems for use with the Commodore computers. Digital I/O board, software, and security equipment. Uses the computer to supervise a system that includes hardwired or wireless (BRS-type) hookup.	$195 (hard wire) $349 (wireless)

means to that end. Since the computer can be programmed in a familiar language such as BASIC, it's easy to design IF . . . THEN conditions that will monitor a variety of sensors and initiate appropriate actions.

Energy control comes to mind as a prime use for an ADC at home. The most basic application, of course, is to provide a super-smart thermostat that can turn temperatures up and down during the day, according to whether anyone's home. Using an array of setpoints for each day of the week, the program can provide the right temperature all the time, and waste a minimum of energy. An electric water heater can be shut off all night and during the day when no one's home, but turned on automatically before morning showers, or evening dishwashing.

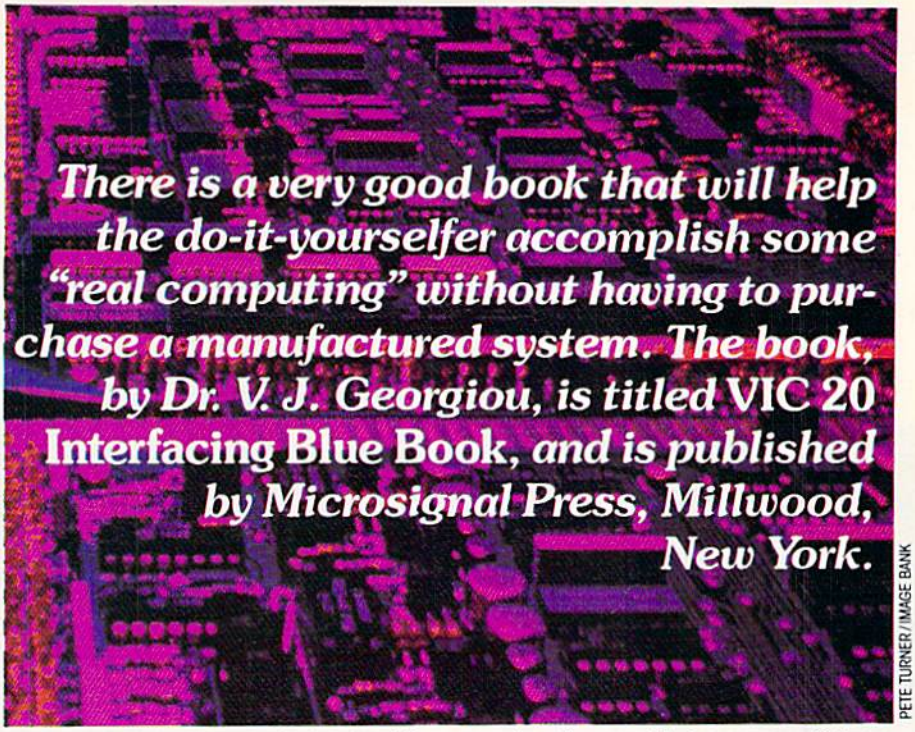
Since the computer can respond to other parameters, such as light levels, a motor can be used to open drapes when the light is bright, and close them in the evening to prevent heat loss. In the summer, the process can be reversed, to minimize air-conditioning costs. In addition, inexpensive sensors are available that clamp a coil around a power line and monitor the current flowing through it. Thus, total electrical consumption can be monitored, or the user can test individual appliances to see how much energy they use.

People with hobbies have also found ways to automate the most boring parts using a computer. A home greenhouse is a pleasure when the plants are automatically watered when they need it. With a soil-moisture sensor, a Commodore computer can also determine when the lawn should be watered, then do so *if* it's the proper time of day *and* it's not currently raining!

Equipment Considerations

Three factors usually determine the suitability of any ADC for a particular project. These factors also determine the ADC's cost.

The first of these factors is *resolution,* which identifies the number of individual increments that can be differentiated, or resolved, over the total measured range. Because digital values are expressed in the form of bits that are turned on or off, resolution of ADC devices is usually expressed in terms of the number of bits used to hold the information. An ADC with eight-bit resolution is able to resolve 256 increments over its entire range. This would mean that a thermometer, for example, that had readings from 0 degrees to 100 degrees would be divided into 256 increments. Each increment would represent .39 degrees. In other words, the accuracy of the reading would be +/− .39 degrees.



PETE TURNER/IMAGE BANK

Increasing resolution to 12 bits divides the range into 4,096 increments, a much higher resolution that allows significantly more accuracy in readings. A thermometer measuring 0 degrees to 100 degrees would have its scale divided into 4,096 units, in this case, which allows readings with an accuracy of ± .0244. A few ADC's that are designed for 16-bit resolution divide the measurable range into 65,536 increments!

A second factor is sampling speed, and is related inversely to resolution. Eight-bit ADC's often are capable of thousands of analog-to-digital conversions per second, 12-bit devices of similar design about 15 per second or less. An ADC's sampling speed must be appropriate for the kind of data being monitored. Someone measuring plant growth might need several measurements a day, while someone monitoring a faster process may need several thousand a second.

The last factor is the number of inputs, outputs, and other special features of the ADC. Some ADC's are limited to a few analog inputs and a couple of controlled outputs. Others come with many inputs, both analog and digital, analog outputs, and even a controller for BSR-type remote-control modules. Some ADC's even contain a microprocessor and RAM memory, and can run without being supervised by a computer.

Do It Yourself!

There is a very good book that will help the do-it-yourselfer accomplish some "real computing" without having to purchase a manufactured system. The book, by Dr. V. J. Georgiou, is titled *VIC 20 Interfacing Blue Book,* and is published by Microsignal Press, Millwood, New York. Most of the projects described in this book are also applicable to the Commodore 64 by taking into consideration the memory differences between the two computers. For example, instead of setting aside 100 bytes of memory at the top of BASIC for a machine-language driver (in the VIC), one could put the same routine at $C000 in the 64. The two computers are identical with respect to the interfacing ports, insuring the success of most of these projects.

One of the projects mentioned in the *Blue Book* is an eight-bit analog-to-digital converter. Although it works with one channel only, and requires building another of the projects to provide a control function, it provides a taste of "real computing"

with a Commodore. Another project is a "true" RS-232 interface for the Commodore user port, which might be needed to connect the computer to ADC equipment that does not support communication at TTL levels.

In the equipment listing in Table 1, I've included some representative ADC models that are all suitable for use with Commodore computers, ranging from relatively inexpensive models to some that are well suited for critical industrial control or research data-acquisition tasks. Entries are listed in alphabetical order according to product name, and at the end is listed a special product—a digital, not analog, interface for home security.

In most cases, these models are *not* the only ones that the companies make, and what must be a brief description here cannot explain all their features fully. In every case, the manufacturers will gladly respond to your inquiry with product literature, and to your detailed request with information and recommendations.

Jerry Houston has degrees in Business Administration and in Data Processing, and is the author of many articles on real-world programming. He is Marketing Director for Remote Measurement Systems, Inc., and teaches computer courses at Griffin College in Seattle and at Central Seattle Community College. C

Morning Coffee

DON CARROLL / IMAGE BANK

The program "Morning Coffee" shows how simple it really is to accomplish useful tasks with a Commodore 64, using an ADC to extend its reach into the world around it. This program uses only one feature of an ADC—the ability to send commands to the popular BSR series of AC line-carrier remote control modules. These modules, sold at many stores, are ordinarily used with a timer or controller to operate lights and appliances around the house by sending radio-frequency signals over the AC lines.

In this case, we have a program that orders a coffee maker to start at a given time in the morning, then turns on a light in the bedroom when the coffee's ready! If you're as hard to get up as I am, you'll also want your clock radio to turn on just before the lights. Struggle out to the kitchen and help yourself to a steaming cup while you try to remember what comes next.

Line 120 opens the Commodore RS-232 channel as channel 2, all set to communicate with a peripheral at 1200 BPS.

Line 140 establishes the values to be sent over the AC lines by the BSR transmitter. These values are defined by BSR as the location codes required to turn on a certain module number. For those familiar with BSR devices, we're setting COFFEE to Module 1 and LIGHTS to Module 2, both using House Code A.

Line 160 assigns C1 (Control 1) to 197 and C2 (Control 2) to 199, BSR's commands to turn a module ON and OFF, respectively. It would be more descriptive to call these commands "ON" and "OFF," but avoid the temptation. The 64's editor won't let you get away with a variable name that is, or contains, a keyword.

Lines 180 and 190 send OFF commands to both modules, making sure that the program starts out with the coffee pot turned off and likewise the lights.

Lines 210-340 simply print a screen for prompts and input three times from the user—the current time (when the program's started), the time for the coffee maker to be turned on in the morning, and the time for lights-on. Only hours and minutes are used here. In each case, enter the hour, then a comma, then the minute.

The main timing and control logic is from 360-470. 360 checks the current time in minutes, multiplying hours times 60 and adding minutes. 370 checks to see whether the current time is equal to the time that the coffee should come on, and returns control to line 360 if not.

Once the time is right, subroutine 490 is executed, which sends the appropriate module number and command to turn on the coffee maker.

The same logic occurs in lines 410 and 420, but this time the coffee is turned off before the lights are turned on. If your coffee maker takes care of this by itself, you would want to delete line 440.

The subroutine at 490 is the only section that would need to be changed to run this program with different ADC systems. The code shown will operate the BSR transmitter of the ADC-1 system from Remote Measurement Systems (see Table 1), the unit with which the author is most familiar. In any event, it's a simple matter. With the ADC-1, for example, line 500 sends the RS-232 channel a byte that identifies the module to be operated, and line 520 sends the byte that indicates whether it is supposed to turn on or turn off. After each byte is sent, a short pause is provided to keep from sending a successive command to the ADC before the BSR transmitter has finished sending the last byte. Because of the way BSR commands are sent through the house wiring, each command takes about 1/3 second to send.

This program is, of course, a very simple example. It should show, however, that real-world computer programming is well within the capability of anyone with a Commodore computer and an ADC system. Though not all use the same commands and some are easier than others to use, in all cases the required code can be written into subroutines that can be tested once, then used forever.

Once you've written a subroutine to read an analog channel, for example, save it as a small program that starts at a convenient line number, like 5000. To read analog input channel number 5 in any of your programs, then, all you'd need to do is something like:

```
200 Channel = 5
210 GOSUB 5000
```

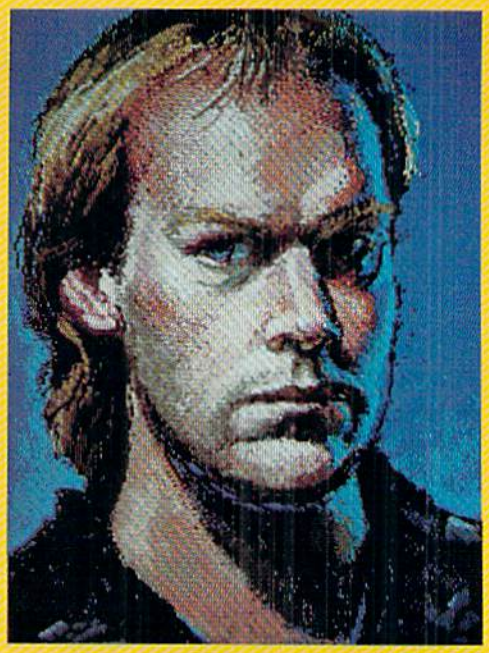
and the value read from that channel will be placed into the variable you've set aside for it.

Similar subroutines can be written to read the condition of the digital inputs, send a BSR command, or operate the controlled outputs. Once these routines are tested, you'll never have to re-invent them, and programming will be a joy. C

Morning Coffee

```
120 OPEN 2,2,0,CHR$(136)+CHR$(0)
    : REM - OPEN RS-232 AT 1200
    BPS'FKPI
130 :'ABHX
140 COFFEE = 204: LIGHTS = 220
    : REM - ASSIGN BSR MODULE 1 AND 2
    ADDRESSES'DASO
150 :'ABHA
160 C1 = 197: C2 = 199: REM - ASSIGN
    COMMANDS FOR BSR CONTROL'DPXN
170 :'ABHC
180 CHANNEL = COFFEE: CTRL = C2
    : GOSUB 500: REM - MAKE SURE
    COFFEE IS OFF'EUHR
190 CHANNEL = LIGHTS: CTRL = C2
    : GOSUB 500:REM - MAKE SURE
    LIGHTS ARE OFF'EVGT
200 :'ABHV
210 PRINT "[CLEAR,DOWN]
    MORNING COFFEE - PROGRAM
    INITIALIZATION[DOWN3]"'BAQJ
220 INPUT " ENTER TIME NOW,
     SUCH AS HH,MM ";HOURS$,
    MINUTES$'BQDK
230 TI$ = HOURS$ + MINUTES$ +
    "00"'DRNG
240 :'ABHA
250 INPUT "[DOWN2] ENTER TIME TO TURN
    ON COFFEE[SPACE2]";HOURS$,
    MINUTES$'BQBO
260 TC$ = HOURS$ + MINUTES$'CRVI
270 T1 = 60 * VAL(HOURS$) +
    VAL(MINUTES$)'FWDM
280 INPUT "[DOWN2] ENTER TIME TO TURN
    ON LIGHTS[SPACE2]";HOURS$,
    MINUTES$'BQKR
290 TL$ = HOURS$ + MINUTES$'CREL
300 T2 = 60 * VAL(HOURS$) +
    VAL(MINUTES$)'FWEG
310 PRINT "[DOWN2,SPACE10]
    HAVE A GOOD NIGHT!"'BAUE
320 PRINT "[DOWN2,SPACE5]
    COFFEE WILL START AT ";LEFT$(TC$,
    2)+":"+RIGHT$(TC$,2)'FPEM
330 PRINT "[SPACE11]LIGHTS ON AT ";
    LEFT$(TL$,2)+":"+RIGHT$(TL$,
    2)'FPKL
340 PRINT "[DOWN2,SPACE4]
    PLEASE TURN OFF MONITOR
    NOW..."'BANK
350 :'ABHC
360 T = 60 * VAL(LEFT$(TI$,
    2)) + VAL(MID$(TI$,3,2))'HXPM
370 IF T <> T1 THEN 360'EGII
380 :'ABHF
390 CHANNEL = COFFEE: CTRL = C1
    : GOSUB 490: REM - TURN ON COFFEE
    MAKER'ESNU
400 :'ABHX
410 T = 60 * VAL(LEFT$(TI$,
    2)) + VAL(MID$(TI$,3,2))'HXPI
420 IF T <> T2 THEN 410'EGFE
430 :'ABHB
440 CHANNEL = COFFEE: CTRL = C2
    : GOSUB 490: REM - TURN OFF
    COFFEE MAKER'ETBQ
450 CHANNEL = LIGHTS: CTRL = C1
    : GOSUB 490: REM - TURN ON
    LIGHTS'ENWQ
460 CLOSE 2: PRINT"[CLEAR]"'CCGG
470 END: REM - END OF PROGRAM,
     SUBROUTINE TO CONTROLLE BSR
    MODILES FOLLOWS'CDOV
480 :'ABHG
490 REM - SUBROUTINE TO CONTROL A BSR
    MODULE:'BGIQ
500 PRINT#2,CHR$(CHANNEL)'CLHD
510 FOR PAUS = 1 TO 1000: NEXT PAUS
    : REM - SHORT DELAY'FBKJ
520 PRINT#2,CHR$(CTRL)'CIQE
530 FOR PAUS = 1 TO 1000: NEXT PAUS
    : REM - SHORT DELAY'FBKL
540 RETURN'BAQD
```

END



"Self-Portrait"

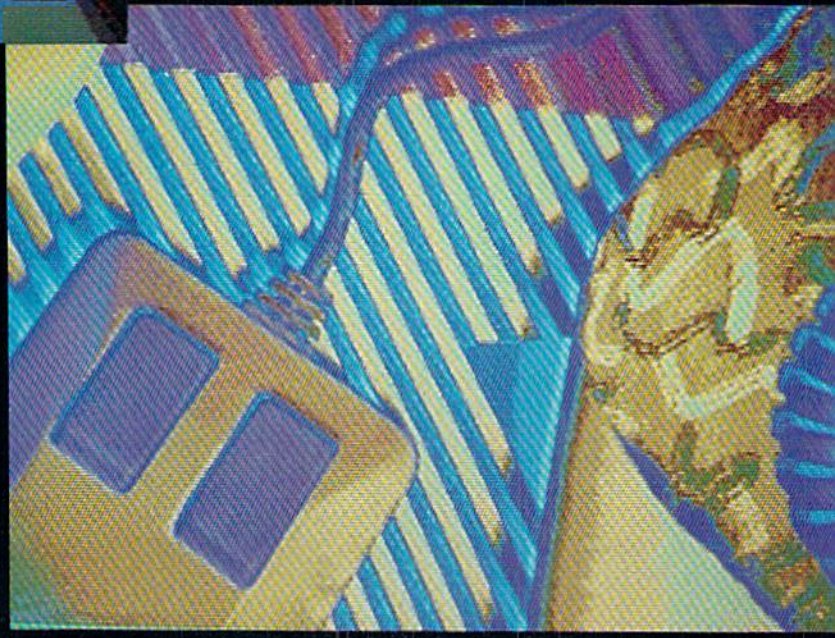
Amiga Update:

Art by Jack Haeger

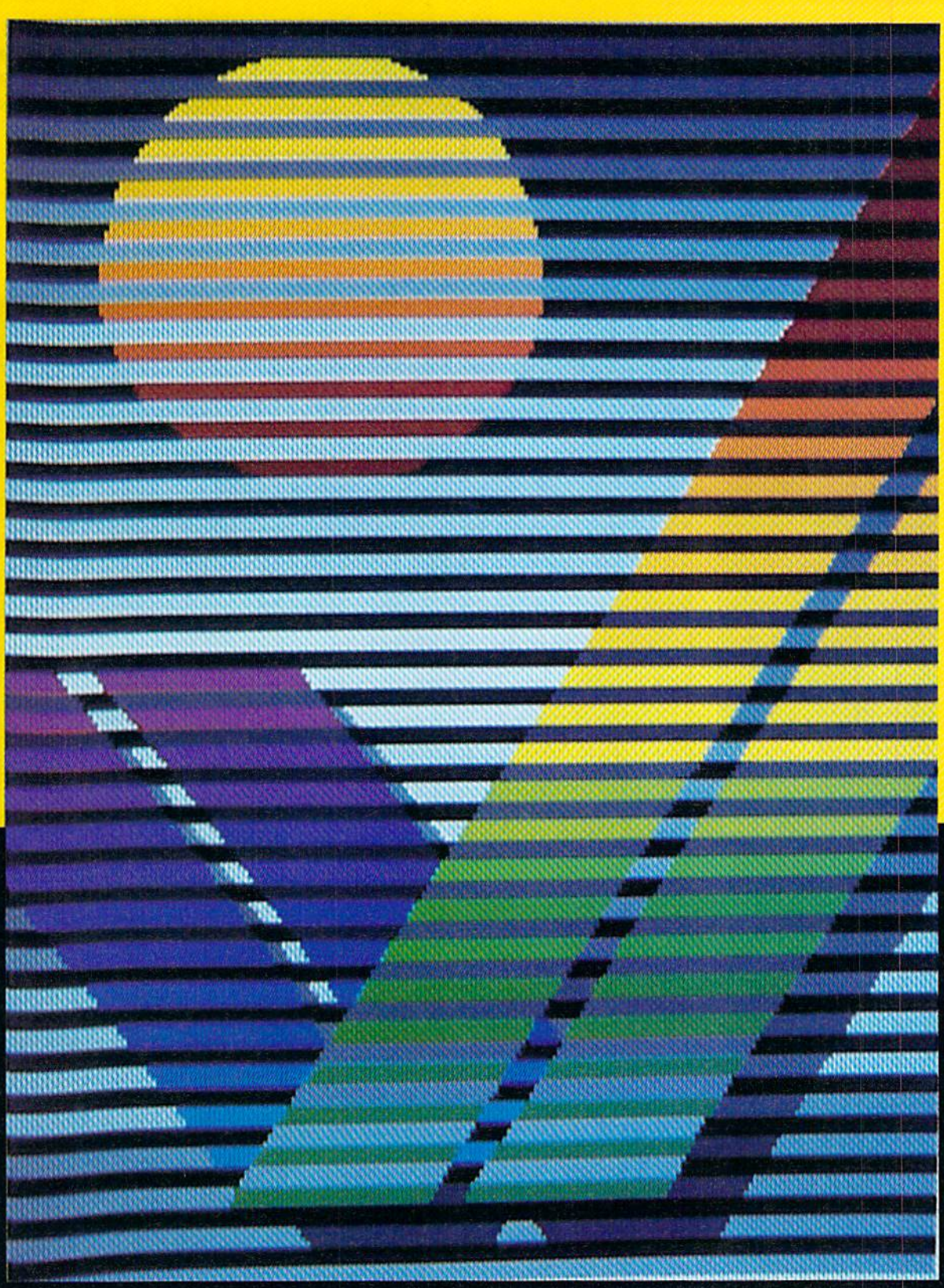
Commodore-Amiga's art director, Jack Haeger, is the creator of many of the Amiga screens you have been seeing over the past few months in various publications. Trained as a painter at the Art Institute of Chicago, Jack characterizes himself as "an artist who uses computers," rather than as a computer scientist. Those of you who frequent video arcades may be familiar with some of his early video work: Before coming to Amiga two years ago, he designed the graphics for Williams Electronics' *Sinistar* and *Star Rider* games.

The screens shown here represent some of Jack's recent work on the Amiga.

"Graphicraft"



"Electric Canvas"



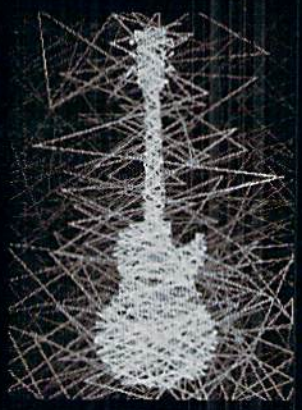
"Checkmark with ball"

Amiga Update:

Art by Jack Haeger

"Keyboard Instrument"

"His Own Thing"



"Power Chord"

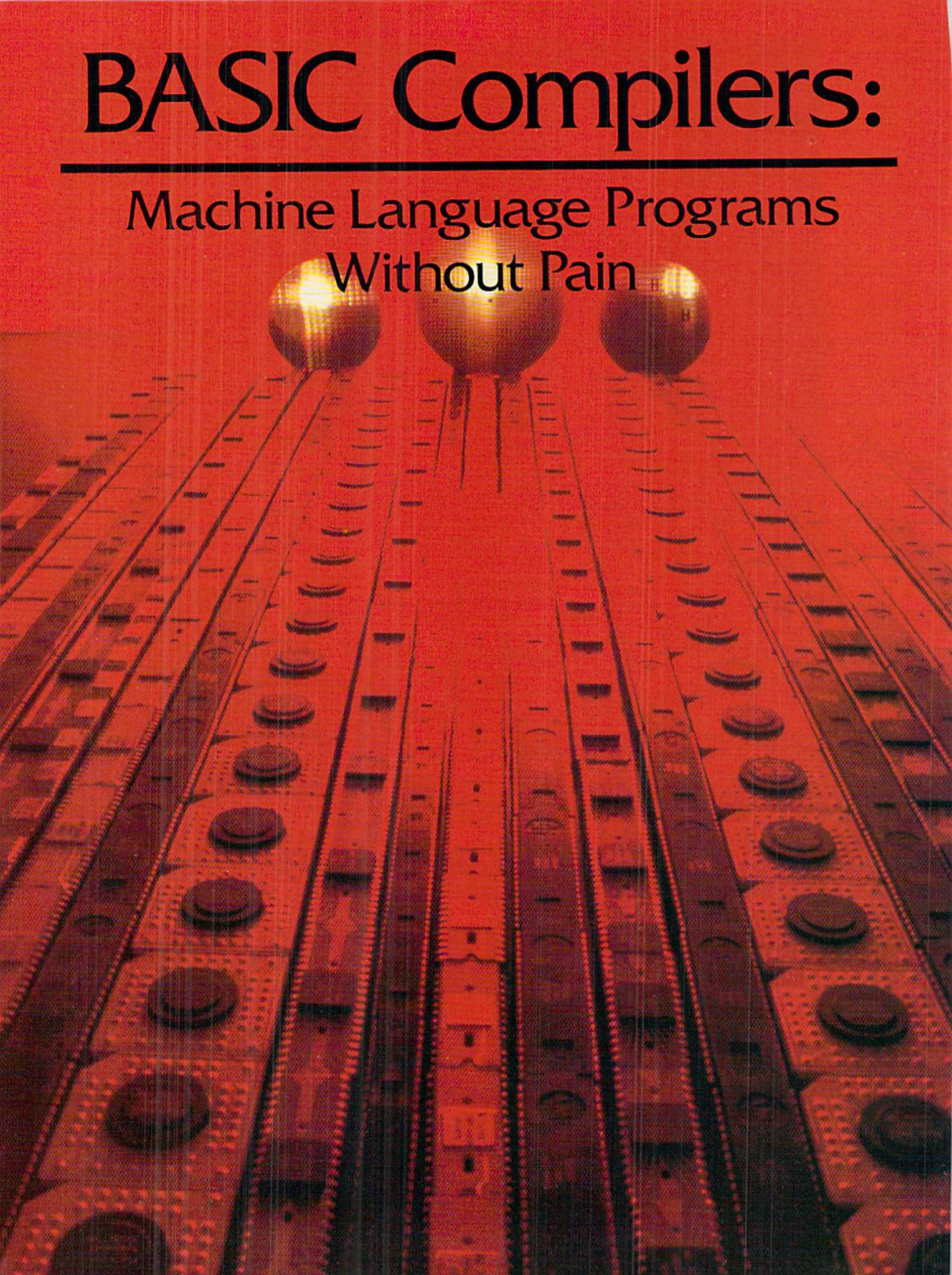
"Custom Colors and Brushes"

"Four Byte Burger"

"Amiga in Stereo"

BASIC Compilers:

Machine Language Programs Without Pain



Here we take a look at what compilers are and what they do, so you can decide for yourself if one is for you.

by Tom Benford

BASIC compilers have been called the greatest programming aid since the home computer. While praise of that order may be overdoing it a bit, compilers are a terrific means of increasing the speed and efficiency of your programs. There are other nice side benefits to compiling BASIC programs, too. For instance, a compiled program cannot be listed.

However, compilers are not a universal panacea. In some applications, they are responsible for tremendous gains in speed. In other applications, where the programming is relatively straightforward, their gains may be minimal. In some instances, they won't make any perceptible difference in execution time and may even add considerably to the size and load time of the program itself.

In order to decide whether your programming efforts will benefit by using a compiler, it is first necessary to understand what compilers are, what they do and what they won't do.

What is a Compiler?

A compiler is a program utility that converts a program from its source form into a more efficient form that the computer can run faster. Compilers are available for virtually every "high level" programming language, in addition to BASIC. Compilers are also frequently incorporated as a part of the programming language's environment, as in Kyan Pascal, Forth, Promal, and others.

What Does a Compiler Do?

In very simple terms, a compiler reads your source (in this case, BASIC) program, digests the commands, statements and variables, finds more direct routes for executing the program by "collecting the garbage," and produces a new program that performs the same as the original source program, but runs much faster. Often (but not always) the compiled program will require less memory and disk space.

How Can Compilers Help Me?

BASIC is an excellent programming language for many reasons. It is easy to learn, it uses English-language commands like "load" and "save," and it is very flexible in its applications and capabilities. On the downside, however, it is slow in program execution, particularly where sprites and graphics are used extensively, because it must first be "translated" into the language the computer understands before it can be executed.

Machine language programs are very fast because they speak directly to the computer without going through an interpreter. The only downside to machine language is that you have to know the language, which is considerably less "friendly" than BASIC.

Compilers are helpful in that they give you the speed and efficiency of machine language programming while requiring only BASIC program code to start with. This is particularly desirable in any program that uses graphics and sprites, such as action games. However, keep in mind that timing may have to be adjusted once a program is compiled. You don't want your sprites, for instance, speeding across the screen so fast you can hardly see them. Compilers, in short, allow the convenience of programming in BASIC while providing the speed of machine language.

Why Use a Compiler?

There are several reasons for using a compiler. First, compiled programs run faster. Some BASIC compilers boast execution-speed increases of between 5 and 300 percent. While this covers quite a bit of latitude in the speed department, this is basically a true claim. The variations in the execution speed are directly attributable to the content of the program itself. As a general rule of thumb, "straight" BASIC programs that don't incorporate SYS calls, repeated looping, or excessive variable strings demonstrate the most impressive speed gains. Programs that do contain these three items will still run faster than in the non-compiled form, but the difference in speed may not be significant.

Second, compiling protects the source code. Since the compiled program is a machine-generated version of the original, the source code in BASIC need not be present to run the compiled version. Compiled programs can't be listed, so the integrity of the BASIC code is protected.

Memory is also more efficiently managed by compiling. During compilation, a "garbage collection" routine is performed. "Garbage" in this sense means REM statements, spaces, line numbers and other such data that is not crucial to the proper operation of the program. The compiler sorts through and eliminates all the garbage when it produces the new version of the program. Since there are fewer bytes in the program without all the garbage, less memory is used.

And, finally, BASIC keywords are eliminated. While the keywords of BASIC (like load or save) make the language easy for humans to understand and work with, they aren't necessary for the computer, which understands very simple instructions. BASIC commands are passed through the computer's interpreter which does a translation into the machine's native binary tongue, machine language. A compiled program is transformed into either pure machine language code or a pseudo-code that is very close to machine language in speed and compactness. This contributes to smaller compiled program size and faster execution.

I Don't Know Machine Language. Can I Still Use a Compiler?

Compilers were designed to aid programmers who don't know machine language, and no machine language programming knowledge is required for effectively using any of the compilers covered here. The documentation that comes with all of these programs provides tips and useful information for creating better source programs that are more "compiler-friendly," but this data is intended for the more serious hacker. The real beauty of compilers is that they handle the machine language code, so you don't have to know anything about it to use them successfully.

Can I Give Compiled Programs to My Friends?

The "transportability" of a compiled program depends on which compiler you've used. Some compilers require that the RTL (run-time library) or symbol tables be present as a file on the disk along with the compiled BASIC program. Others incor-

TOM TRACY / IMAGE BANK

porate such required programs or data into the actual compiled program itself, which accounts for the increased size of some compiled programs. Still others require that a encryption device or "dongle" be inserted into a joystick port for the program to run.

As I stated earlier, nothing is perfect. Make sure you understand the trade-offs involved before you decide to purchase a compiler.

Will My Program Load Faster After Compiling?

NO! This is a commonly misunderstood point about compilers. Compilers will not decrease the loading time of a program in most instances. They will decrease the program execution or "run" time of the program, however, especially in longer and more complex programs using graphics and sprites.

I mentioned earlier that some compilers require the use of RTL and SYM (symbol) files. These files must also be loaded into computer memory to run your compiled programs. Since more code has to be loaded into memory, the load time may be increased, even though the program execution time is decreased.

With some compilers, the RTL and SYM files only have to be loaded once, and they are retained in memory as long as the computer is on. Subsequent compiled program loads are usually very fast using this arrangement. Others require loading these files prior to running each compiled program. Once these program files are in memory, however, subsequent loads of compiled programs are faster.

What Are "P-code," "Speed-code" and "Pseudo-code"?

These three terms all refer to the same symbolic token code for BASIC that results after passing through the computer's interpreter. "P-", "speed-" and "pseudo-" are all the same thing, although the name varies from compiler to compiler. Since this code bypasses the interpretive step in running a program, it executes faster than a BASIC program.

BASIC-64 Compiler

Abacus Software
P.O. Box 7211
Grand Rapids, MI 49510
(616) 241-5510
$39.95 suggested retail price

This offering from Abacus is everything a good compiler should be: It is easy to use, it is efficient, it offers a good range of optional features, it comes with excellent documentation, and it is inexpensive.

One of the nicest features about *BASIC-64* is that it produces compiled programs that are fully transportable by themselves—it is not necessary to supply an RTL (run-time library) or symbol table with the program. During the second compiling pass, the run-time module is merged into the program and the data is inserted. Since the compiled program already has the run-time module incorporated into it, separate support files aren't necessary.

A unique feature of *BASIC-64* is that you're given a choice of which code you wish to generate: speed-code, 6502/6510 machine code, or a combination of the two.

The opening screen provides you with these choices: Compiler/Optimizer I; Compiler/Optimizer II; Advanced Development Package; and Overlay. Other options are presented on sub-menus that depend on your initial selection.

The Compiler/Optimizer I option is totally compatible with the BASIC interpreter of the 64. All calculations

Compilers allow the convenience of programming in BASIC while providing the speed of machine language.

are performed as whole number operations, provided the whole number falls within the normal integer value range (−32768 to +32767). Numbers outside of this range are changed to floating-point numbers.

The Optimizer I option doesn't affect program behavior but it does increase execution speed. Since the program uses integer calculations for its speed gains, it helps considerably if all numeric variables in the source program are converted to integer variables (by adding a "%" to each variable name) prior to compiling with this option. This option is useful for virtually all BASIC compilation, since it produces a faster compiled program with a minimum of fuss and bother.

Compiler/Optimizer II functions differently from Optimizer I and the BASIC interpreter. It treats all variables (except string variables) as integers by placing the "%" after each. Dividing two integers is performed in whole-number operations rather than the normal floating-point division procedures. Optimizer II will also ignore decimal places and automatically convert numbers into integer data. Optimizer II is ideally suited for compiling programs that require mixed variables or which normally don't allow the use of integer variables in BASIC 2.0.

The Advanced Development Features option moves you into the serious application arena. By selecting this option, it is possible to compile BASIC programs into pure 6502/6510 machine code. Programs compiled using this option have an "M-" prefix in the disk directory, designating them as machine language programs.

Advanced Development also allows you to input the name of a symbol table to be loaded before compiling, which is desirable in some programming applications. A symbol table will

One of the nicest features about *Basic-64* is that it produces compiled programs that are fully transportable by themselves—it is not necessary to supply a run-time or symbol table with the program.

automatically retain all variables and memory addresses, which is useful when several programs will require the same variable data.

Also provided with this option is the ability to save a symbol table. Symbol tables may be listed to the screen or printer using the included SYMBOL program on the disk.

An address list may also be generated from within the Advanced Development section. This optional address listing displays the memory addresses on the left while the BASIC line numbers of the original source code are shown on the right. This feature is a blessing when you need to find errors in a program section that starts with a SYS call.

The end-of-memory address may also be changed and the starting address raised or lowered on a compiled program using the Advanced Development section. You may also remove the run-time module and load it as a separate program with this option. Controlling the connection of the run-time module with the compiled program, as when merging a series of programs, is also facilitated using this option.

Source programs written using BASIC extensions such as *Simons' BASIC, VICtree, BASIC 4.0* and others may also be compiled using the Advanced Development section. In addition, the ELSE command can be located and adjusted using this option.

BASIC-64 contains a lot of little convenience features that can only be fully appreciated by using a compiler that doesn't have them. One such feature is the option of switching the run-time error-handling facility on or off. When the error handling is on, the compiler will halt when an error is encountered; turning it off allows compilation to continue uninterrupted. The error-handling toggle is turned on by simply adding a "line 0" in the BASIC source program.

The Overlay Feature may be suppressed from within the Advanced Development section, and disk commands may be enabled. This is very useful for deleting a source code program after it has been successfully compiled, in order to save disk space.

The Overlay Feature allows you to compile a number of successive programs which share the same set of variables, but you need to observe certain restrictions to successfully use this feature. The internal load commands must contain the correct form of the program name. For instance, you must enter LOAD "P-NEXT-PART,"8 instead of LOAD "NEXT-PART",8. The first program must also be longer than all subsequent programs in the overlay group. If these kinds of requirements are not met, your compiled programs won't load and execute properly.

Compiler directives are used to inform the compiler of any changes during compilation. *BASIC-64* permits directives to arrange variable addresses, switch the error handling, declare integer variables, switch either the machine code or speedcode generators, switch optimizers, declare floating point variables, and free the cassette-buffer memory. Uses for these directives are fully described in the manual, which also includes examples of their use.

Blitz! may be configured for a single drive, a dual drive, or two single drives with different device numbers, for optimum compiling efficiency.

Of the three compilers covered here, I found *BASIC-64* to be the best in terms of operation, features, options and price.

DTL 64 Compiler

The Drive Technology Ltd. (DTL) *64 Compiler* by David Hughes is an improved and upgraded version of the *DTL BASIC Compiler* originally developed for the PET series. It is being marketed in the U.S. by two companies under two different names that I'm aware of: Microsci as *InstaSpeed* and by Codewriter as *SpeedWriter*). Since this is essentially the same program with only minor differences in features, I'll cover the main portion of this compiler here and attend to the particular nuances of each later.

DTL Common Features

The following features are common to both *InstaSpeed* from Microsci and *SpeedWriter* from Codewriter.

The run-time library (RTL) file must be in memory in order for a compiled program to run. It is therefore necessary for this RTL file to be present on the same disk as any compiled programs. Without the RTL file, the compiled program is not transportable, so if you wish to make duplicates of your compiled programs, you must duplicate the RTL file also.

Compiling a program is a two-pass operation. The first pass reads the source code one line at a time and a work file is opened to hold a semi-compiled version of the program. This work file resides in a memory area not used by BASIC. The work file is read back during the second pass, data is added and edited, and an object code file is created. The object code file is written in speedcode.

The compiler is totally compatible with the 64's BASIC interpreter, and does error checking while compiling and running. It is far more convenient to correct your errors as they occur in the source code program, however. BASIC extensions are supported by the compiler and it can handle most of the popular extensions such as *Simons' BASIC* and *VICtree.*

Control files may be used to control the compilation of several programs in sequence. This feature is very useful in doing "layered" programs that load and run other programs from within themselves.

A special integer mode is available by using an SI compiler directive. This mode converts all numeric data into integer from by placing a "%" after the data automatically. Both integer and floating-point arithmetic modes are supported, although integer is much faster.

This compiler allows variable list

Continued on pg. 122

PsychoS

The High-Tech Way to

for the Commodore 64

The following is an actual conversation that took place between Dan Gutman (the author of this article) and *Eliza* (the computer psychiatrist).

Eliza: How do you do? Please tell me your problem.
Dan: I don't have a first sentence for this article.
Eliza: I am not sure I understand you fully.
Dan: I don't know how to start the article.
Eliza: Don't you really know how to start the article?
Dan: No, I don't.
Eliza: Aren't you saying "no" just to be negative?
Dan: I'm saying no because I can't think of an opening sentence.
Eliza: Is it because you can't think of an opening sentence that you came to me?
Dan: Yes.
Eliza: You seem quite positive.
Dan: Can you help me start off the article?
Eliza: You believe I can help you start off the article, don't you?
Dan: I hope so.
Eliza: Why do you need to tell me you hope so?
Dan: Because I do.
Eliza: Is that the real reason?
Dan: No.
Eliza: You are being a bit negative.
Dan: Yes, I am being negative.
Eliza: Are you sure?
Dan: No, I'm not sure.
Eliza: How long have you been not sure?
Dan: The hell with it. I'll start the article by myself.
Eliza: Can you elaborate on that?

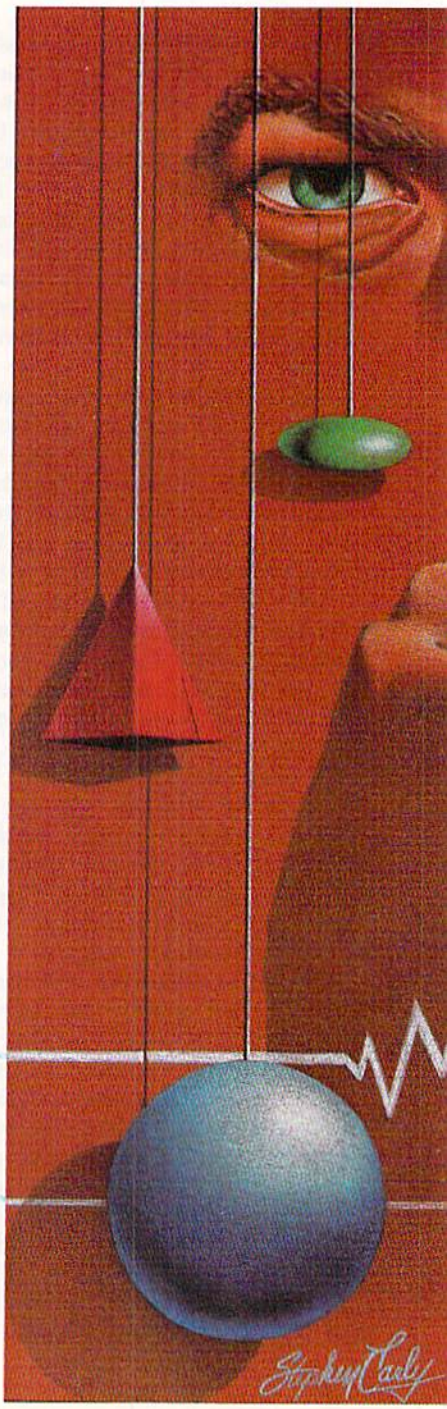
Eliza is a gag program, of course. The funny thing is that *Eliza* is totally stupid, but she sounds almost like a real psychiatrist. When M.I.T. psychologist Joseph Weizenbaum created the program in 1966, he just wanted to show how computers could be used to converse in English. Weizenbaum was shocked to hear that the program was being praised by professional psychologists as a breakthrough in the mental health field. "They told me that with *Eliza* we could treat hundreds of patients an hour at state hospitals," he said in Jack Rochester's book, *The Naked Computer* (1983).

Eliza never helped anybody with their problems. But it did suggest that there was some potential for computers in this area. Today, "self-enrichment software" is a legitimate category. Buying a computer program is a lot cheaper than conventional therapy or psychological testing, and you don't have to discuss your personal problems with another human being. Best of all, you can conduct your computer consultation from the comfort of your keyboard.

Get Smart

I.Q. (intelligence quotient) is probably the most controversial topic in education. Experts don't even agree on what intelligence *means,* much less how it should be tested. Factors like upbringing, motivation, fatigue, and anxiety have all been found to influence I.Q. scores. The only thing known for sure is that people who score high on I.Q. tests usually do well in school and at work.

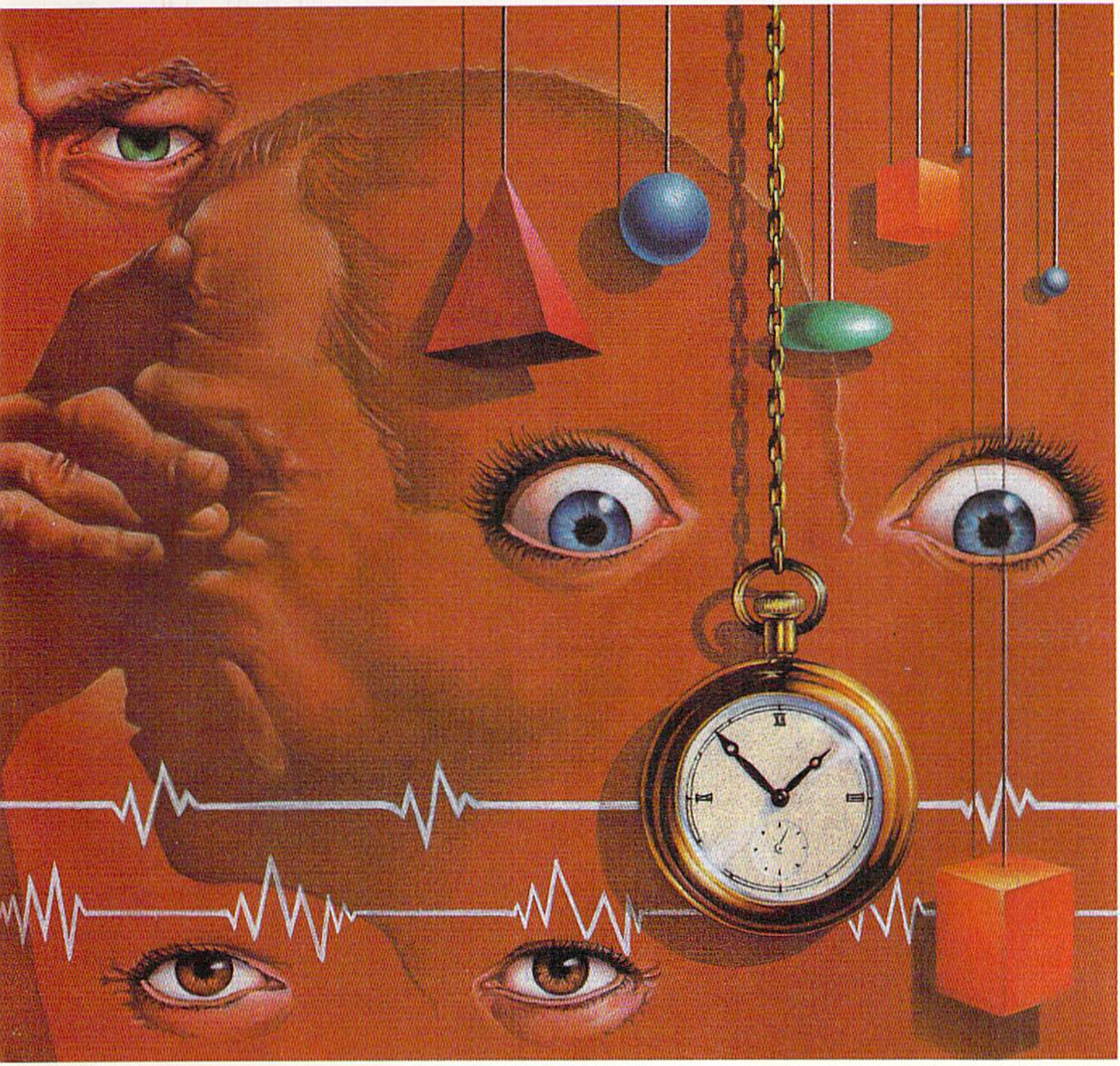
Bantam's *Know Your Own I.Q.* was developed by Drs. H.J. Eysenck and Glen Wilson—world famous experts in intelligence—so if you believe in



oftware

Get Your Head Together

By Dan Gutman



I.Q. testing at all, you can probably trust this one.

Like most I.Q. tests, *Know Your Own I.Q.* is difficult, frustrating, baffling, and seemingly nonsensical. If you think you're pretty bright upstairs, this may cut you down to size. I reluctantly confess that my score was 102.5, and I think I was lucky to do *that* well. (100 is considered to be average intelligence, 180 is a genius.) Just what I need—a high-tech way to confirm how dumb I am. For some of us, it may not be a good idea to know your own I.Q.

The program contains four I.Q. Tests, which can be taken over and over again. Each test has 40 questions, which must be completed in 30 minutes (there's a time clock on-screen). If you score higher than I did, you may choose to print out the results. If you *really* want to gloat, you can frame the printout and hang it on a wall.

I.Q. scores can be a dangerous weapon, and Dr. Eysenck is careful to note that his test is designed for entertainment, not as a method of classifying people. In fact, you only receive a specific score if your I.Q. falls between 100 and 130. The computer will inform you if you've scored below 100 or above 130, but it won't tell you the exact number. In other words, it will tell you that you're smart or dumb, but it won't tell you *how* smart or *how* dumb.

Bantam never mentions this, but the *real* power of their I.Q. test is that it can be used to practice for future I.Q. tests. Unlike most of these tests, this one tells you the correct answer to each question *and* explains why that answer is correct. You may find the questions to be totally incoherent—but when you read the explanations and understand what the test-makers were looking for, it can help prepare you for an I.Q. test that counts. I think this program could probably be used to raise your score on future I.Q. tests.

I didn't say "raise your I.Q." That's impossible, according to psychologists. But if you learn the tricks of taking the test, I believe you can inflate your score to make it *seem* like you have a higher I.Q.

Know Thyself—And Thy Friends

More interesting than the I.Q. test is *Know Your Own Personality,* which is on the same disk. The computer asks you 210 yes/no questions (like, "Would you like to watch an execution if given the opportunity?") to test for three personality traits: extraversion/introversion, emotional stability/adjustment, and tough-/tender-mindedness. At the end of the test you're presented with bar charts comparing your various traits, and a paragraph that describes your personality.

Bantam never mentions this, but the *real* power of their I.Q. test is that it can be used to practice for future I.Q. tests.

Taking the test can be fun, as long as your head is screwed on reasonably well. Bantam suggests inviting close friends over and throwing a "personality party." I wouldn't recommend it. Your close friend might not appreciate having everyone hear, "You have a low opinion of yourself. You believe you are an unattractive failure. Your low score could be an indication of an inferiority complex. You are characteristically pessimistic, gloomy and depressed. You are disappointed with your existence and at odds with the world." That can cool off a party *real* fast.

Naturally, you don't need a computer to take an I.Q. or personality test. The only advantage of the computer is instant feedback. You get your results about a second after completing the last question.

This program can help you understand yourself or another person better, and it can be a lot of fun—*if* you're intelligent and charming. The catch is that the computer also makes it easy for you to find out that you're dumb and you have a bad personality, which can be a drag.

There are a few other personality programs you might want to check out. Psycom Software's *Personality Analyzer* is based on the theories of Swiss psychologist Carl Jung. This program differs from *Know Your Own Personality* mostly in that it gives a more detailed report. *Personality Analyzer* will tell you which of 16 personality types you fit in, occupations that would be most suited to your personality, what you might be like as a mate, and your potential weaknesses.

Another evaluation program worth checking out is *The Self-Analysis Program,* from the International Self-Help Institute. This program asks you 220 true/false questions and ranks your *maturity* level from 0-100. You can even get your handwriting analyzed with Franklin Software's *The Handwriting Analyst.*

It's also possible to analyze somebody *else's* mind. Human Edge Software's *Mind Prober* was a very controversial program in 1984, mostly because it was cleverly advertised as being a manifestation of George Orwell's classic *1984. Mind Prober* doesn't actually read minds, but it lets you get to know a person very well after knowing him or her very little. Here's how it works: The program presents you with a list of 60 adjectives that might describe another person ("cultured, carefree, goal-oriented, silly", etc.). You decide whether or not each adjective fits the person you're trying to "read."

When you're done, the computer analyzes how the various personality traits interact. You'll be presented with seven screens of text describing the person's attitudes about work, stress, and sex, their likes and dislikes, fantasies, personal values, and personality traits. It's almost as if you have broken into the office of that person's psychiatrist and peeked at the files.

Before the 1984 presidential election, the editors of *InfoWorld* magazine used *Mind Prober* to probe Ronald Reagan's mind. The program responded, "This person would love to be an actor, a politician, or something else that would give him center stage and national prominence." Not bad, huh?

Like *Know Your Own Personality, Mind Prober* is no party game. The profiles aren't always flattering. In fact, a husband and wife probably shouldn't sit down and try it on each other unless the marriage is very solid. In general, the program is pretty accurate, and it gets more accurate the

better you know your subject. But then, if you already know your subject, who needs *Mind Prober*? It's a useful tool, but don't worry about Orwell's Thought Police busting down your doors just yet.

Grant . . . Hayes . . . Garfield

Does this mean anything to you: "I saw a ton of wash floating in a big basket next to a dam. On top of the dam a chef and his son were throwing pots and spoons at a man rowing a boat." It's a mnemonic device to learn the names of our first four presidents—Washington, Adams, Jefferson, and Monroe.

Another device would be to get DesignWare's *Remember!,* which is a handy tool designed to help high school and college students memorize course material. The program is fairly easy to use. You type the material to be memorized into the computer in the form of questions and answers. For example, if you're trying to learn the capitals of the 50 states, you'd type NEW JERSEY as a question and TRENTON as its answer. When you're finished, the computer will test you by presenting NEW JERSEY. You then have to type in the correct capital city. The program organizes the list of questions and answers, sort of like flashcards.

To help you form mental associations, the computer lets you create hints for each question. The program contains a simple art and music program, so you can make a drawing or write a tune to accompany a question. If you have trouble remembering that the chemical symbol for iron is Fe, for instance, it might help to draw a simple picture of a ferris wheel carrying irons.

Remember! lets you test yourself in several ways. You can respond to your original question, you can receive the answer first and respond with the question (like on the T.V. quiz show *Jeopardy*), you can receive multiple-choice questions, or the computer can put the answers in a list format. You may also have the computer ask you the questions in the same order you typed them or in random order. When you've finished the test, the computer tells you how many questions you got right, and which ones you missed.

Remember! has one limitation that prevents it from being useful for serious memorization. You can only ask yourself 20 questions per lesson. So in the example of memorizing the Presidents of the United States in chronological order, the program reaches its limit by the time you reach James Garfield. That would be fine if it were 1881, but in 1985 a student needs to memorize the names of *40* Presidents.

Mind Prober is a useful tool, but don't worry about Orwell's Thought Police busting down your doors just yet.

Still, *Remember!* can be very useful for memorizing vocabulary words (in English, French, German, Spanish, and Italian), dates, telephone numbers, short sections of text, chemical symbols, states and capitals, and short lists. The act of typing the material into the computer, reviewing it, and taking a test on it certainly promotes memorization. Another memory program you might want to check out is Avant Garde's *The Einstein Memory Trainer.*

Plug Yourself In

In the sixties, psychotropic substances promised salvation, happiness, and bliss. It didn't happen. In the seventies, Transcendental Meditation was supposed to help us find Utopia. It didn't happen. Now it's the eighties, and we've finally found the one true answer to coping with the stress of everyday life—computers.

Well, maybe. *Relax* is an offbeat device that has you strap on an electronic headband and plug the other end into your computer. Three small sensors on the headband record the tension in your forehead by measuring electrical activity in the muscles. When you're tense, electrical activity increases.

As you stare at the screen, the headband sends the information it is receiving to your computer. You'll see a continuous graph before you registering your muscle-tension level. It's a sensitive gauge—when you tighten your muscles on purpose, the simulated needle jumps up instantly. This is called biofeedback. The idea is that we are usually not aware of slight changes in our body functions. But if we can see graphs of them right up there on the screen, it's possible for us to manipulate them. Some people can use biofeedback to change their heart rate or even their brain waves at will. With other people, it doesn't work at all.

Relax doesn't just have you sit there and watch your muscles tense up. At the same time, you wear headphones and listen to an audio tape with a soothing voice that guides you in deep relaxation exercises. The tape takes you through three games. In one of these, kaleidoscopic patterns and colors change according to how relaxed you are. Far out! And it's legal.

Relax was developed with the help of Dr. Martha David, a clinical psychologist at California's Kaiser Permanente Medical Center. I'm not sure if this is the thing that will finally bring us eternal happiness, but it will do until the nineties, when we'll probably be able to insert electrodes directly into our brains. For now, stressed biofeedback fans might also want to check out Thought Technology's *Calmpute.*

Another—slightly wackier—biofeedback program is Psycom Software's *The Hypnotist.* Hypnotism used to be associated with the occult and stage magicians who would pull people out of the audience and make then cluck like chickens. Now, of course, it's a legitimate form of therapy. But instead of paying a professional hypnotist $50 an hour, why not hypnotize yourself at home?

The Hypnotist combines biofeedback and behavior modification techniques. The disk comes with a "PSI Biofeedback Device" that straps on your wrist and pinky finger. It calculates the elapsed time between your heart beats. In theory, lowering your heart rate makes the subconscious mind more receptive to hypnotic suggestion.

"Welcome to the land of virtual time and space," says Kurian, an Egyptian fellow whose soothing words appear on the bottom of the screen. Kurian will encourage you to let the tension drain from your head,

Continued on pg. 127

HOW TO
TO A HIGHER

THE COMMODORE 128.

The first step is buying the Commodore 128™ Personal Computer. The smartest computer available for the price. It's like getting three computers for less than one usually costs, because the 128 operates in three separate modes. You can run sophisticated CP/M® business software and the new programs written for the 128. Plus over 3,000 Commodore 64® programs. You start out with more software and real life uses than most machines give you after years on the market.

THE COMMODORE 128 WORKS FASTER.

To run all that software and run it faster, you'll want the 1571 Disk Drive. You can't find a faster drive at the price. It transfers nearly 1,000 words a second (5200 cps), so you can load most programs instantly. And you'll save space as well as time because the 1571 holds up to 410K of data, the equivalent of 200 typewritten pages.

THE COMMODORE 128 GETS SMARTER.

Now try improving your memory. Plug in our 1750 RAM Expansion Module and your 128 moves up to a powerful 512K. That's enough to handle just about anything you can dish out, from complicated business forecasting to giant data bases. In fact our expansion module will be the only memory you'll need for many years to come.

EVOLVE NTELLIGENCE.

THE COMMODORE 128 LEARNS TO COMMUNICATE.

There's no real intelligence without the ability to communicate. So you'll want our 1670 Modem/1200. Just plug in your telephone jack and the modem automatically answers, dials and selects the appropriate mode. It puts you in touch with a new world of shopping, banking, communications and information over your telephone line. And it operates at a lightning-fast 1200 baud to save on your telephone bill.

THE COMMODORE 128 LEARNS TO WRITE.

Looking good in print could be your next move with the MPS 1000 Printer. It's a new dot matrix printer designed to make the most of the 128's high-resolution graphics because sometimes pictures speak louder than words. But it's no slouch when it comes to words. The MPS turns out about 1200 words a minute (100 cps) of draft-quality printing, or gives you near-letter-quality printing at nearly 240 words a minute (20 cps). And you can choose printing styles, use international characters, even make up your own symbols.

THE COMMODORE 128 IMPROVES YOUR VISION.

Brains aren't enough without good looks, so improve your vision with Commodore's new 1902 RGB Color Monitor. The high-resolution screen gives you a sharper image and better color than your standard TV, so you can really appreciate the 128's great graphics. And the 80-column display lets you see more of what you're doing while you're doing it.

All these evolutionary steps ahead won't set you back when it comes to paying for them. Additions to your Commodore 128 are available at a store near you and are as affordable as the 128 itself. We think that's a smart way to help you build a computer system.

COMMODORE 128 PERSONAL COMPUTER

A Higher Intelligence

Sound Processing for the Computer Musician

These sound-processing devices can greatly enhance the quality of music on the Commodore 64. You can buy them from musical instrument dealers, or build them yourself from kits.

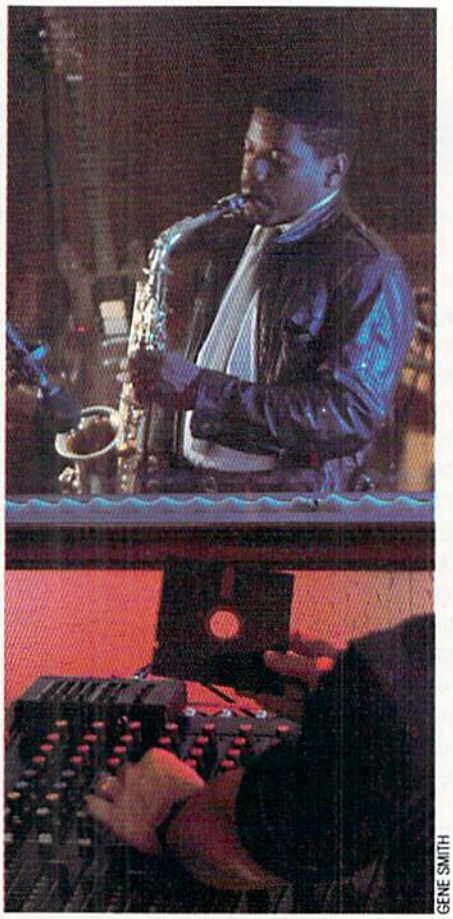
There's a growing addiction out there in music land, and you're probably hooked on it without even knowing it. I'm talking about sound processing. These days, it's hardly possible to hear live or recorded music that hasn't been deliberately enhanced in some way.

In the past few years, the sound-processing tools of professional sound studios and performers have become more accessible to the amateur musician. There's no better way to get acquainted with this topic than by looking at what can be done with the Commodore 64's SID chip. However, most of what I have to say in this article applies to sound from any source, computer or otherwise.

The basic goal of audio signal processing is simply to change the way music sounds, hopefully for the better. An audio signal initially consists of what you want to keep—and improve—plus a noise component that you want to get rid of. Let's talk about the noise component first, because it can be troublesome with the SID chip.

I indicated before that there's no one right way for electronic music to sound. Nonetheless, just about everybody recognizes noise when they hear it. Noise is easy to find; just hook up your Commodore 64's audio output to a good sound system. (Listening through your monitor's built-in audio circuitry will be less satisfactory, as this is typically not a very good sound system.) The hums, hisses, and buzzes you hear are the electronic noise generated both within the SID chip and externally by other electrical activity in your computer.

If you have a program executing while you're listening to the audio output, that program may itself be the cause of some of the noise you hear. I was made acutely aware of this while using a music program that allows you to play its music files either from within the program or in the immediate mode (*Master Composer* from Access Software). When the program is running, a high-resolution graphics screen is displayed while music is playing. In the immediate mode, music files are played by loading them and SYSing to a machine language routine. In this second case, nothing is happening on the screen while the music is playing.



GENE SMITH

The difference between the noise levels in these two modes is quite remarkable, and it led me to conclude that the most important piece of sound processing equipment for computers is the music software itself. If SID sound is being generated while a lot of other operating system or program activity is going on, the noise level is going to be much higher than that inherent in the SID chip itself. If music files must be played from within a program where, for example, high-resolution graphics are being generated at the same time, then a screen-blanking function should be provided to eliminate the extra electrical interference caused by this activity.

There's also a noise signal present at the audio output, even when all SID voices are off. One goal of signal processing is to remove this signal. The problem has its counterpart in other sound systems incorporating several processing devices at once; the cumulative audio noise may be quite noticeable, even though you wouldn't notice any noise from the devices used individually.

The first place to attack this "end of the line" noise on the Commodore 64 is, again, in the software. Proper programming will ensure that all SID voices are gated off when they're not being used. This is necessary because the volume of SID voices doesn't really drop to zero at the end of the release cycle.

Once the SID is really turned off, the residual noise can often be completely eliminated with a device called a noise gate. It works by completely blocking any signal below a selected threshold level. When a larger signal comes along (hopefully the music), the gate opens up and everything, including the noise, passes through unchanged. Although the noise is still present, it is largely masked by the sound.

The adjustment of a noise gate is critical, especially if it is used with devices that have high noise levels to begin with. Otherwise, when the audio signal has a long attack or release time (that is, when the volume increases or

decreases slowly with time), the sound may "pop" in or out in an undesirable way. However, the effect of a properly adjusted noise gate on an appropriate signal is stunning.

Several additional steps can be taken to remove noise from an audio signal. The Dolby and dbx systems for audio cassettes are well known examples of active noise-reduction circuitry. These are both encoded systems, which means that when you apply them in the recording mode, the resulting signal must be decoded by similar circuitry during playback.

Other Devices for Improving Sound

There are other approaches to noise reduction that do not use encoding. They operate by examining the audio signal in real time, deciding what's noise, and throwing that part away. This sounds simple enough, but it is difficult in practice because the circuit may have a hard time defining noise, even when your ears don't have any trouble hearing it.

One kind of active non-encoding noise reduction is called dynamic filtering. It starts with the basic premise that a lot of what we perceive as noise is in the higher part of the audio frequency spectrum. (Tape hiss is a good example.) A dynamic filter listens to an audio signal, isolates its high-frequency components, compares that signal to the total signal, and decides whether to keep the high-frequency part or throw it away.

If, for example, the signal contains lots of low-frequency components, the filter might decide (rightly or wrongly) that the high-frequency component is noise. The dynamic part of the filter continuously adjusts the level and frequency at which signals are kept or rejected so that it acts like a smart, continuously variable low-pass filter. (See the SID section of your *Commodore 64 Programmer's Reference Guide* for a basic discussion of filters.)

Dynamic filtering is not without its drawbacks. It may result in a signal that sounds dull, because too many high-frequency components have been removed. This can be overcome with an active brilliance controller, which analyzes the dynamically filtered signal in real time and puts back some high-frequency components. Does this sound a bit circular? It's not, because now the high-frequency components are harmonically (that is, musically) related to the filtered signal and don't contain the random components our ears would perceive as noise.

If SID sound is being generated while a lot of other operating-system activity is going on, the noise level is much higher than that inherent in the SID chip itself.

Does it work? This is an interesting question in psychoacoustics because there's no fundamental reason why our processed signal should be any better than the original one. However, I think most people would agree that this kind of noise reduction and restructuring of an audio signal is important, because our ears seem quite sensitive to the random signals that we interpret as noise, especially when they're superimposed on the orderly signal that we interpret as music.

A cautionary note is in order here. The noise signal from the SID chip is not all random high-frequency noise. It contains some low-frequency components as well, in the form of a hum or buzz that will look to a noise-reduction filter like a perfectly acceptable audio signal. It won't magically go away just because you know you don't want it!

There are many other kinds of signal processing that aren't related to noise reduction. The graphic equalizers common in stereo systems are one familiar example. Such devices can be used to minimize low-frequency hum from the SID output or to add brilliance to the treble. Note, however, that the latter is an example of passive, as opposed to active, processing that won't perform the specialized function of the brilliance controller I discussed before.

A parametric equalizer makes possible more sophisticated sound shaping. It allows you to tune the frequency at which sound is given a "boost" or "cut," and alter the sharpness of the circuit's response. On graphic equalizers, the amount of boost or cut is variable, but the frequencies and sharpness are fixed. Parametric equalizers are often used to give instrumental or vocal sounds a more natural "presence," and this is also useful for electronic instruments. Judicious frequency equalization is a good way to improve the clarity of multi-part music.

A delay or "reverb" circuit is another useful device. Electronic music can sometimes sound dry because it lacks the acoustic ambience of natural sound. Recent advances in microchip technology have made it possible for digital delay circuits to replace the expensive and unwieldy analog delay devices previously required for this job. When applied to SID output, reverb tends to smooth out the jagged sound that can occur during long attacks and releases.

If you plan to do any recording, you will soon be interested in a limiter. Synthetically generated sound tends to contain a lot of high-frequency voltage spikes (called transients) that can easily overdrive the input circuitry of recording equipment. Often these transients are too short for a recording-level meter to respond to them; you won't know it's happening, but the input signal will still produce significant distortion in the recorded sound. You might ask how you can tell what's distorted and what's not without comparing the recorded signal to the original. Don't worry. Your ears will know the difference because of the non-harmonic relationship between the basic signal and its distorted components.

A limiter acts like a safety valve. It's designed to respond very quickly to signals and cut them off when they exceed a preset level. (Actually, it's not quite that easy, because simply clipping off a signal can introduce its own kind of distortion.) If you listen to the signal going into a well designed limiter and compare it to what comes out, you might (and should) conclude that this particular sound processor isn't doing anything at all!

However, you'll find it can increase the quality of your recordings by allowing you to record at a generally higher level without paying the penalty of distortion. This is one of the biggest differences between studio-quality and amateur recordings, and is one of the easiest improvements to make in your own system.

Of course the list of sound processors goes on and on. The last one I'll mention is a stereo synthesizer. This will take a monophonic output, such as that from the SID chip, and divide it into a stereo signal, with bass on one side and treble on the other. Recording studios have been doing this for years, but it's only recently that good stereo synthesizers have been available at a reasonable price.

Here's a bit of good news for those of you who are accustomed to worrying about computer hardware and software compatibility. While it's possible in principle to mismatch the sound-processing devices I've been talking about, basically you can hook several of them up, even if they're from different manufacturers, to just about any sound source you can think of. There's nothing that restricts their use with the SID chip, Commodore computers, or any other specific sound source. So, you can acquire and use them with the confidence that they won't outlive their usefulness as your own skills and interests expand to MIDI interfaces and keyboard synthesizers.

Build Your Own Devices and Save

All the equipment I've discussed is available through musical instrument dealers, but you may not be pleased when you see the price tags. My own approach has been to build what I wanted from kits. All the equipment I've discussed is available and relatively inexpensive in this form. However, a word of warning: This equipment is not for the novice kit builder. The circuits themselves aren't particularly complicated, but the kit manufacturers serving this still rather obscure market often assume that their customers possess substantial electronic and mechanical skills. For example, most of the individual units I've built don't include cases, and some of them don't even include front panels. If you don't know a mylar capacitor from an LED, and you're all thumbs with a drill, you will have trouble building this equipment!

Now, to balance the words of warning, here are some words of encouragement. There's no reason that electronic novices can't learn the required skills. They can get up to speed by acquiring *Electronic Projects for Musicians* by Craig Anderton (Guitar Player Books, NY, 1980), from PAIA Electronics (see below for addresses). This is an excellent introduction to electronic construction skills, and contains many inexpensive and useful projects, including the noise gate I use myself. Parts kits for the projects are available from PAIA Electronics. Finally, don't forget that your local Commodore user's group probably includes several people who are used to working with electronic equipment

and will be glad to help!

Sources

Catalogs are available from the following sources of sound processing kits:

PAIA Electronics, Inc.
1020 W. Wilshire Blvd.
Oklahoma City, OK 73116
Typical products:
- Power Supply (No. 7700, see note below, $59.95)
- Noise Gate (Project #27, $32.95)
- Parametric Equalizer (No. 6760, $59.95)
- Dual Limiter (No. 6790, $49.95)
- Reverb (No. 6740, $59.95 or No. 4712, $26.95)

RODCAR Electronic Sales
10290 Monroe Drive, Suite 202
Dallas, TX 75229
Typical products:
- Motherboard (No. 1560, see note below, $29.95)
- Dynamic Noise Reducer (No. 1580, $24.95)
- Brilliance Controller (No. 1550, $19.95)
- Parametric Equalizer (No 1500 A-C, $24.95-$54.95)
- Limiter (No. 1570, $29.95)
- Stereo Synthesizer (No. 1400, $119.95, or $149.95 for complete kit including case and power supply)

Note: A separate power supply is required. One power supply will be sufficient for several modules from either company. The RODCAR Motherboard kit includes a power supply and plug-in connectors for its other kits. A power transformer is extra, and costs about $7. The RODCAR kits do not include a front panel, mounting hardware, or cases (except for the stereo synthesizer, as noted). The PAIA kits include panels but not cases; appropriate cases, sufficient for several modules, are available separately.

One more note: Both the PAIA reverb units listed above use spring-type analog delay lines. The No. 4712 uses a single-delay line and produces the muddy sound quality associated with this kind of technology. The No. 6740 uses a double-delay line and claims to be of much higher quality, although I have not tried it. Inexpensive digital-delay units, either monophonic or stereo, are available from Radio Shack stores.

Both PAIA and RODCAR will provide knowledgeable technical advice for kit builders and users of their equipment either by telephone (no toll-free numbers) or by mail. I must report that PAIA Electronics has an annoying order-taking policy. If you order items by phone or mail and the items are out of stock (not an unusual occurrence), your account will still be charged at the time of the order. My experience is that some kits may be out of stock for as long as several months and that promised delivery dates are not always met. So, unless you don't mind paying well in advance for equipment you don't have yet, check carefully on the availability of any item you order and insist on being kept informed about realistic delivery times. C

BY LOUIS F. SANDER

Spriteseer 2.0

for the Commodore 128

This sprite program is a sophisticated yet easy-to-use system for creating, examining and altering sprites on the Commodore 128.

Last issue, we described the Commodore 128's exceptional sprite-handling capabilities. All the sprite-related commands were illustrated in depth, and a few "tools" were included, to make it easier for you to view, change and save the sprites you created.

This month, you'll see what happens when a BASIC 7.0-crazed programmer gets hold of those tools. I spent several weekends working with the tool programs, and the result was Spriteseer 2.0. This much longer program combines the features of its predecessors, and adds significant extra capabilities. When you type it in, you'll have a sophisticated yet easy-to-use system for creating, examining and altering sprites, and for working with your library of sprites. Specifically, Spriteseer 2.0 lets you do these things:

- Create sprites with the 128's SPRDEF command.
- Put text into sprites, from one of three different fonts.
- Display sprites in an orderly way, in any color on any background color.
- Observe sprites as they move at various speeds and angles.
- Save sprites to disk, and load them back again.

Unlike some sprite editor programs, Spriteseer 2.0 lacks facilities for rotating, inverting and other activities useful in creating animations. Perhaps we'll add them at a later date.

When you type in the program, be careful not to renumber any lines. Much of Spriteseer's operation depends on their being numbered exactly as shown. You can add lines almost to your heart's content, but you must not renumber the lines shown here. The only restriction on adding lines is for line numbers between 32 and 255, which are reserved for data for the text characters. (We use BASIC 7.0's selective RESTORE statement to select the character to be printed. The line number must equal the character code.) Lines below 400 are numbered in rather non-standard ways—be careful when you type them. Everything above 400 is numbered in increments of 10, so you can use the AUTO command when entering them. Be careful with line 1150—the characters between the quotes are shifted spaces.

When you run the program, eight sprites will be displayed at the bottom of the screen, and seven lines of text will appear at the top. The cursor will be flashing, indicating that the program has reached its end. At this point you are in direct mode, and you can enter any commands you'd like. The purpose of the text on the screen is to automate the entering of certain commands, and the changing of one critical program line.

That line, number 410, is listed at the top of the screen. It controls many aspects of the sprite display. The line has already been executed, of course, giving the sprite display you see right now. To change the display for subsequent runs, you can change line 410 as follows:

- E controls sprite expansion. Set it to 1 for expanded sprites, 0 for unexpanded sprites.
- M sets multicolor mode off (0) or on (1).
- The number above the arrow marked COLOR establishes the color of the sprites. It's 15 right now, but you can set it to any number between 1 and 16. To give each sprite a different color, change it to the letter K.
- SPRCOLOR controls the multicolor mode. See your manual and our previous article for details.
- The number above the arrow marked PRIORITY controls whether sprites pass in front of (0) other material on the screen, or behind it (1).

If you change line 410, of course, you must re-run the program to execute the changes. Do that by placing your cursor on the RUN301 : VIEW SPRITES line, then pressing RETURN. (Isn't it nice—when you hit RETURN to change line 410, your cursor ends up right on RUN301! Just press RETURN again to execute it.)

There's a reason to RUN301 instead of just RUN. Line 300 sets the screen, character and border colors to white, green and green. If those colors don't suit your fancy, you can change them while in direct mode by pressing the color keys or by executing a COLOR0 or COLOR4 statement. If you subsequently RUN301, your new colors won't be lost. Convenient, eh?

If you put your cursor on the RUN302 : SAVE SPRITES line and

Computer Wizard regularly presents elementary topics of interest to Commodore computerists. It emphasizes the needs of beginners, but is of use to advanced computerists as well. The column is written to be easily understood by all, and to be of lasting value to its readers. If you have comments or suggestions for Computer Wizard, please write to Louis F. Sander, in care of this magazine.

press RETURN, you'll be prompted for saving your sprites to disk. You can save all eight sprites, or any lesser number, just by selecting the range of sprite numbers to save. If you don't enter a file name, nothing will be saved, and you'll return to the main display.

RUN303 : MAKE SPRITES gives you prompts for making sprites. To try it, put your cursor there and press RETURN. One selection lets you use SPRDEF, the 128's built-in sprite editor. (For information on using SPRDEF, see your manual and our previous article.) To terminate SPRDEF, just press RETURN in response to the SPRITE NUMBER? prompt. You'll return to Spriteseer, so you can see and save your creations. If you choose the Lettersprites option, you can make up sprites that contain up to three lines of text. You can use the 128's regular character set in your sprites, which gives you three characters per line, or you can use one of Spriteseer's fonts, which give you either four or six characters per line. The fonts are not too shabby, if I do say so myself. If you use them, you'll see the individual letters being created on the screen.

RUN304 : LOAD SPRITES lets you load sprites from the disk, just by entering their file names. If you want to see a directory, type a dollar sign in response to the prompt. To see a selective directory, type a dollar sign followed by your selection. Typing $SPR.* will give a directory of all files whose names begin with "SPR." If you fail to enter a file name, nothing will be loaded.

And finally, RUN305 : MOVE SPRITES animates your sprites on the screen. They move at various speeds, and every few seconds their directions change at random. You can use this feature to see how your sprites will look in action, and to check out the PRIORITY feature. Depending on the sprites, it can also be entertaining on its own. Clear the text from the screen, and it's sort of like watching fish in a tank. C

Before typing this program, read "How to Enter Programs."

Spriteseer 2.0

```
30 REM ** LINES 32-255 ARE RESERVED! **
31 :
32 DATA SP,00,00,00,00,00,00,00,00,00,00
33 DATA !,40,40,40,00,40,80,80,80,00,80
45 DATA -,00,00,70,00,00,00,00,E0,00,00
46 DATA .,00,00,00,00,80,00,00,00,00,80
48 DATA 0,F8,98,A8,C8,F8,E0,A0,A0,A0,E0
49 DATA 1,60,20,20,20,F8,C0,40,40,40,E0
50 DATA 2,F8,08,F8,80,F8,E0,20,E0,80,E0
51 DATA 3,F8,08,78,08,F8,E0,20,60,20,E0
52 DATA 4,88,88,F8,08,08,A0,A0,E0,20,20
53 DATA 5,F8,80,F8,08,F8,E0,80,E0,20,E0
54 DATA 6,F8,80,F8,88,F8,E0,80,E0,A0,E0
55 DATA 7,F8,10,20,40,80,E0,20,40,40,40
56 DATA 8,F8,88,F8,88,F8,E0,A0,E0,A0,E0
57 DATA 9,F8,88,F8,08,F8,E0,A0,E0,20,E0
65 DATA A,F8,88,F8,88,88,E0,A0,E0,A0,A0
66 DATA B,F8,88,F0,88,F8,E0,A0,E0,A0,E0
67 DATA C,F8,80,80,80,F8,E0,80,80,80,E0
68 DATA D,F0,88,88,88,F0,C0,A0,A0,A0,C0
69 DATA E,F8,80,E0,80,F8,E0,80,C0,80,E0
70 DATA F,F8,80,E0,80,80,E0,80,C0,80,80
71 DATA G,F8,80,98,88,F8,E0,80,A0,A0,E0
72 DATA H,88,88,F8,88,88,A0,A0,E0,A0,A0
73 DATA I,F8,20,20,20,F8,E0,40,40,40,E0
74 DATA J,08,08,08,88,F8,20,20,20,A0,E0
75 DATA K,88,90,A0,D0,88,A0,A0,C0,A0,A0
76 DATA L,80,80,80,80,F8,80,80,80,80,E0
77 DATA M,88,D8,A8,88,88,A0,E0,A0,A0,A0
78 DATA N,88,C8,A8,98,88,90,D0,B0,90,90
79 DATA O,F8,88,88,88,F8,E0,A0,A0,A0,E0
80 DATA P,F8,88,F8,80,80,E0,A0,E0,80,80
81 DATA Q,F8,88,88,A8,F8,E0,A0,A0,E0,40
82 DATA R,F8,88,F8,90,98,E0,A0,C0,A0,A0
83 DATA S,F8,80,F8,08,F8,E0,80,E0,20,E0
84 DATA T,F8,20,20,20,20,E0,40,40,40,40
85 DATA U,88,88,88,88,F8,A0,A0,A0,A0,E0
86 DATA V,88,88,88,50,20,A0,A0,A0,A0,40
87 DATA W,88,88,A8,D8,88,A0,A0,A0,E0,A0
88 DATA X,88,50,20,50,88,A0,A0,40,A0,A0
89 DATA Y,88,88,F8,20,20,A0,A0,E0,40,40
90 DATA Z,F8,10,20,40,F8,E0,20,40,80,E0
160 DATA SP,00,00,00,00,00,00,00,00,00,
    00
256 :
257 REM ** LINES 32-255 ARE RESERVED! **
258 :
300 COLOR 0,02:COLOR 1,16:COLOR 4,14
    :COLOR 5,14:GRAPHIC 0
301 GOTO 400 : REM THIS MUST BE LINE 301
302 GOTO 550 : REM THIS MUST BE LINE 302
303 GOTO 660 : REM THIS MUST BE LINE 303
304 GOTO 730 : REM THIS MUST BE LINE 304
305 GOTO 820 : REM THIS MUST BE LINE 305
390 REM VIEW SPRITES
400 PRINT"[CLEAR]":FOR K=1 TO 8
    :MOVSPR K,0#0
410 E=0:M=0:SPRITE K,1,15,1,E,E,M
    :SPRCOLOR 01,03:REM........COLOR^
    ^PRIORITY
420 A=RCLR(0):B=A:IF B>8 THEN B=B-16
430 FOR J=1 TO 8:IF RSPRITE(J,1)=A THEN
    :SPRITE J,1,B+8
440 NEXT
450 MOVSPR K,72*K+288*(K>4)-24,
    146-(56*(K>4)):NEXT K
460 PRINT"[CLEAR]";:LIST 410
    :CHAR 1,0,24,"[DOWN,HOME,DOWN2]"
470 PRINT"RUN301 : VIEW SPRITES
480 PRINT"RUN302 : SAVE SPRITES
490 PRINT"RUN303 : MAKE SPRITES
500 PRINT"RUN304 : LOAD SPRITES
510 PRINT"RUN305 : MOVE SPRITES
520 END
530 :
540 REM SAVE SPRITES
550 GOSUB 920 : REM PRINT HEADING
560 F=0:INPUT"FIRST SPRITE TO SAVE
    (1-8)[SPACE2]1[LEFT3]";F
    :IF F<1 THEN RUN 400
570 PRINT
580 L=9:INPUT" LAST SPRITE TO SAVE
    (1-8)[SPACE2]8[LEFT3]";L
    :IF L>8 THEN RUN 400
590 FL=3584+64*(F-1):LL=3584+64*L
```

```
600 P$="FILENAME TO USE":PRINT
610 GOSUB 950 : REM INPUT F$ OR GET DIR
620 Q$=CHR$(34):PRINT"[CLEAR,DOWN3]
    BSAVE";Q$F$Q$;",B0,P";FL;"TOP";LL
630 GOTO 770 : REM DYNAMIC KEYBOARD
640 :
650 REM MAKE SPRITES
660 GOSUB 920 : REM PRINT HEADING
670 INPUT"SPRDEF OR LETTERSPRITES (S OR
    L)[SPACE3]S[LEFT3]";A$
680 IF LEFT$(A$,1)="L"THEN 1030
690 IF LEFT$(A$,1)="S"THEN:FOR J=1 TO 8
    :SPRITE J,0:NEXT:SPRDEF
700 RUN 400
710 :
720 REM LOAD SPRITES
730 GOSUB 920 : REM PRINT HEADING
740 P$="[SPACE2]FILE TO BLOAD"
750 GOSUB 950 : REM INPUT F$ OR GET DIR
760 Q$=CHR$(34):PRINT"[CLEAR,DOWN3]
    BLOAD";Q$F$Q$
770 X$="RUN301[HOME]":PRINT"[DOWN4]
    PRINTCHR$(147);DS$;TAB(80);X$;[HOME]
    "
780 POKE 842,13:POKE 843,13:POKE 844,13
    :POKE 208,3
790 END
800 :
810 REM MOVE SPRITES
820 GOSUB 920:TRAP 890:K%=0
    :PRINT" PRESS THE STOP KEY TO STOP
    THE MOTION"
830 PRINT"[DOWN] PRESS CLR TO CLEAR THE
    SCREEN OF TEXT"
840 FOR J=1 TO 8:MOVSPR J,15*K%#J:NEXT
850 FOR I=1 TO 200:IF PEEK(213)=51 THEN
    PRINT"[CLEAR]"
860 NEXT
870 K%=RND(0)*22+1:SOUND 1,K%*2000,5
880 GOTO 840
890 TRAP:RUN 400
900 :
910 REM PRINT THE HEADING
920 PRINT"[CLEAR,DOWN,SPACE2]
    SPRITESEER - LOUIS F. SANDER - 1/86
    [DOWN]":RETURN
930 :
940 REM INPUT FILENAME OR GET DIRECTORY
950 PRINT TAB(17)"[DOWN,CMDR T16,UP2]"
960 F$="":PRINT P$;:INPUT F$
    :IF F$=""THEN RUN 400
970 IF LEN(F$)>16 THEN:PRINT:GOTO 950
980 A$=LEFT$(F$,1):B$=MID$(F$,2)
    :IF B$=""THEN B$="*"
990 IF A$="$"THEN:FOR J=1 TO 8
    :SPRITE J,0:NEXT:PRINT:DIRECTORY(B$)
    :PRINT:GOTO 950
1000 RETURN
1010 :
1020 REM CREATE LETTERSPRITES
1030 FOR J=1 TO 8:SPRITE J,0:NEXT
1040 GRAPHIC 2,1,3:PRINT"[CLEAR,DOWN4]
     THIS PUTS LINES OF TEXT CHARACTERS
     INTO"
1050 PRINT"ANY SPRITE. THERE ARE 3
     FORMATS:"
1060 PRINT"[DOWN] A. 2 LINES OF 3
     CHARACTERS EACH
1070 PRINT"[DOWN] B. 3 LINES OF 4
     CHARACTERS EACH
1080 PRINT"[DOWN] C. 3 LINES OF 6
     CHARACTERS EACH
1090 INPUT"[DOWN]WHICH FORMAT (A,
     B OR C)";FO$:IF FO$<"A"OR
     FO$>"C"THEN 1090
1100 IF FO$="A"THEN NL=2:CL=3:CW=8
1110 IF FO$="B"THEN NL=3:CL=4:CW=5
1120 IF FO$="C"THEN NL=3:CL=6:CW=3
     :PRINT"[DOWN]5 CHARS MAX FOR LINES
     CONTAINING 'N'"
1130 TRAP 1450:DIM A%(64)
     :UL$="[DOWN,CMDR T6]"
1140 FOR LN=1 TO NL:PRINT
     :PRINT TAB(11);LEFT$(UL$,1+CL)"
     [UP2]";CHR$(141);"LINE#";LN;
1150 POKE 842,34:POKE 208,1:INPUT L$(LN)
     :L$(LN)=L$(LN)+"[SHFT SPACE8]"
     :REM SHIFTED SPC!
1160 NEXT LN
1170 IF FO$="A"THEN CHAR 1,0,0,L$(1)
     :CHAR 1,0,1,L$(2):GOTO 1240
1180 PRINT"[DOWN]WORKING - 40 SECONDS"
1190 FOR LN=1 TO NL:XX=0
1200 FOR CH=1 TO CL
1210 A$=MID$(L$(LN),CH,1)
1220 RESTORE(ASC(A$)):GOSUB 1310
     :REM DRAW IT
1230 NEXT CH:NEXT LN
1240 S=0:INPUT"[DOWN]WHICH SPRITE TO
     PUT IT IN (1-8)";S:IF S=0 THEN 1270
1250 SSHAPE A$,0,0,23,20
1260 SPRSAV A$,S
1270 GRAPHIC 0
1280 RUN 400
1290 :
1300 REM DRAW A CHARACTER
1310 READ L$:K=0:IF CL=6 THEN FOR J=1
     TO 5:READ A$:NEXT
1320 IF L$="SP"THEN 1430
1330 FOR J=1 TO 5
1340 READ BY$:BY=DEC(BY$)
1350 FOR BI=7 TO 0 STEP-1
     :A%(K)=SGN(BY AND 2^BI):K=K+1
     :NEXT BI
1360 NEXT J
1370 FOR Y=0 TO 7
1380 FOR X=0 TO 7
1390 Z=A%(8*Y+X):IF Z THEN DRAW 1,
     X+XX+(CW+1)*(CH-1),Y+7*(LN-1)
1400 NEXT X
1410 NEXT Y
1420 IF CL=6 AND L$="N"THEN XX=XX+1
1430 RETURN
1440 :
1450 PRINT"UNDEFINED CHARACTER! TRY
     AGAIN.":END:REM ERROR TRAP
```

END

BY DICK BARNES

SuperPET Potpourri

As promised in the last *PotPourri* column, this one is devoted to sorts. I won't cover any theories of sorting, because most computer books and many articles in this and other magazines do so in depth. But I do give you three structured sorts of different types, each useful for a purpose, and each adaptable to all languages in SuperPET except APL.

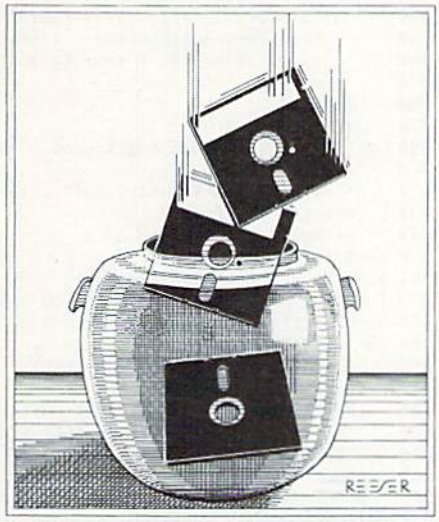
There is, of course, no such thing as a "best" sort. The listings this issue show three different types: a bubble, a shell, and a version of Hoare's justly famed "quicksort." Your first reaction, I'd guess, is that nobody in his right mind ever uses a slow bubble sort—which can be dead wrong.

In Table 1, I compare the performance of a double-bubble, shell, and quicksort on randomly arranged (disordered) and an almost ordered (nearly sorted) lists. The "ordered" list in Table 1 was in alphabetical order except for six names I added at the top of their alphabetical place; i.e., at the top of the B's I added "Blucher" and "Buxford"; at the top of the D's, "Daxon" and "Dydford," and so on—just as you might stuff new names into an existing list.

Hoare's quicksort obviously is best at sorting random lists, but is miserable when it attempts to handle a list which is almost ordered. The double-bubble, conversely, is miserable when it sorts a random list, but performs superbly on an almost ordered one. The shell sort can't beat the other two at their best, but it wins over the bubble at sorting random lists and whips quicksort hollow on almost ordered lists. You should therefore use each sort for the job it does best. When in doubt, the shell is the best compromise.

I've included complete listings of each sort because you can easily get into trouble on the proper option base (do arrays start with zero or one?), and because most published sorts are either in BASIC 4.0 (obscured in a maze of GOSUB's and GOTO's) or in Pascal, which many cannot follow.

There is no such thing as a "best" sort. It depends on what kind of list you're working with.



Whatever the SuperPET language you may choose to adapt the sorts to, be sure to use integer values for all variables (done %, for example, indicates an integer variable in mBASIC). If you use real variables, the speed of the sorts will be cut down by fifty percent or thereabouts. In all SuperPET languages, integer variables are handled in two memory bytes; real variables in five. Enough said.

To save space, only the shell sort in Listing 1 outputs the sorted list to disk. The other sorts may be modified easily to do so. In addition, the shell converts any name beginning in lower case, such as "de Smet" to "De Smet," and sorts it as such. At output, the name is converted back to "de Smet." Without this feature, such names would appear on a sorted list after the Z's.

Every list brought into these sorts should contain one item or name for each line, as in the line: "Williams, G. Penny." Don't expect to sort a list in which items are separated by commas, as in: "Grumpy, Addled, Woofer, Bates, Zebra . . ." We use "linput" to get a line at a gulp; a line is defined as a series of characters ended by a carriage return.

The double-bubble is a bit faster than the usual bubble sort because it alternately bubbles a light item (such as a name beginning with "A") upward, and then bubbles a heavy item down.

All the sorts listed are useful for light and occasional sorting, but if you must sort often, you need one in machine language. For information on such sorts for the 6809-side of SuperPET, write me at P.O. Box 411, Hatteras, N.C. 27943. C

Table 1: Comparison of Times to Sort for Three Sorts (in Seconds)

Array Size	Quicksort		Double Bubble		Shell Sort	
Sorted:	List Random	List Ordered	List Random	List Ordered	List Random	List Ordered
24	4	5	14	2	7	3
75	14	33	98	6	28	12
150	33	147	407	10	75	28
300	79	749	1503	16	256	55
600	205	. . .	. . .	. . .	. . .	. . .

Listing 1. Shell Sort

```
100 ! shell-bd. Adapted from published shell sorts.
110 option base 1 : one%=1 : two%=2 : zero%=0 : thirtytwo%=32 : CS$=chr$(12)
120 !                    INPUT SECTION
130 print CS$; : input 'Enter number of items to be sorted: ', num_items%
140 print : input 'Enter name of disk file to be sorted: ', file$
150 open #30, file$, input : dim list$(num_items%)
160 on eof ignore : print 'Opening files and loading...'
170 for i% = one% to num_items%
180   linput #30, list$(i%)
```

Continued on pg. 94

Computers For The Blind

Talking computers give blind and visually impaired people access to electronic information. The question is how and how much?

The answers can be found in "The Second Beginner's Guide to Personal Computers for the Blind and Visually Impaired" published by the National Braille Press. This comprehensive book contains a Buyer's Guide to talking microcomputers and large print display processors. More importantly it includes reviews, written by blind users, of software that works with speech.

This invaluable resource book offers details on training programs in computer applications for the blind, and other useful information on how to buy and use special equipment.

Send orders to:

National Braille Press Inc.
88 St. Stephen Street
Boston, MA 02115
(617) 266-6160

$12.95 for braille or cassette, $14.95 for print. ($3 extra for UPS shipping)

NBP is a nonprofit braille printing and publishing house.

```
190   if io_status() 0 then quit
200   if list$(i%) = '' then i% = i%-one%    ! Remove blank lines.
210   first% = ord(list$(i%)(one%:one%))
220   if first% ) 96                         ! Any lower case at string start?
230     long% = len(list$(i%)) + one%
240     list$(i%)(one%:one%) = chr$(first%-thirtytwo%) ! Change leading lower
250     list$(i%)(long%:long%)=chr$(255)               ! to caps; mark string.
260   endif
270 next i%
280 num_items% = i%-one% : close #30
290 !                          THE SHELL SORT
300 print CS$; 'Sorting...' : t1=time
310 half% = num_items%/two% : halflist% = num_items%-half%
320 for j% = one% to num_items%
330   for i% = one% to halflist%
340     if list$(i%) )= list$(i%+half%)
350       transfer$ = list$(i%) : list$(i%) = list$(i%+half%)
360       list$(i%+half%)·= transfer$ : exchange% = i%
370     endif
380    next i%
390    if exchange%
400      halflist% = exchange%-one% : exchange%=zero%
410    else
420      half% = half%/two% : halflist% = num_items%-half%
430    endif
440    if half% ( one% then quit
450 next j%
460 t2=time : print 'End of sort'
470 !               PRINT RESULTS TO DISK AND SCREEN.
480 open #40,file$+'.sorted', output
490 for i%=one% to num_items%
500   marker%=idx(list$(i%),chr$(255))
510   if marker%                             ! Substitute lower case.
520     first%=ord(list$(i%)(one%:one%))
530     list$(i%)(one%:one%)=chr$(first%+thirtytwo%)
540     list$(i%)(marker%:marker%)=''
550   endif
560   print #40, list$(i%) : print list$(i%)
570 next i%
580 print t2-t1;'seconds to sort.'
590 stop                                   ! STOP closes all files.
```

END

Listing 2. Double-Bubble

```
100 ! double_bubble:bd. From Robert Dray. Version outputs only to screen.
110 print chr$(12); : option base 1
120 input "Enter filename of file to be sorted: ", file$
130 print : input "Enter number of items to be sorted: ", num_items%
140 dim list$(num_items%) : open #12, file$, input
150 on eof ignore
160 for i%=1 to num_items%             ! The list on disk must be formed
170   linput #12, list$(i%)            ! with a single entry on each line.
180   if io_status then quit
190 next i%
200 num_items%=i%-1
210 reset : print "Sorting ..." : t1=time : call double_bubble
220 t2=time                            ! "reset" closes all open files
230 mat print list$
240 print "Time to sort:";t2-t1;"seconds"
250 stop
260
270 proc double_bubble
280 upper_bound%=num_items% : lower_bound%=2
290 last_swap_up%=upper_bound% : last_swap_down%=lower_bound%
300 loop
310   for j%=upper_bound% to lower_bound% step -1
320     if list$(j%-1) ) list$(j%)
330       exchange$=list$(j%) : list$(j%)=list$(j%-1)
340       list$(j%-1)=exchange$ : last_swap_down%=j%
350     endif
360   next j%
370   if lower_bound%=last_swap_down%+1 then quit
380   lower_bound%=last_swap_down%+1
390   for k%=lower_bound%-1 to upper_bound%
400     if list$(k%-1) ) list$(k%)
410       exchange$=list$(k%) : list$(k%)=list$(k%-1)
420       list$(k%-1)=exchange$ : last_swap_up%=k%
430     endif
440   next k%
450   upper_bound%=last_swap_up%
460 endloop
470 endproc
```

END

Listing 3. Quicksort

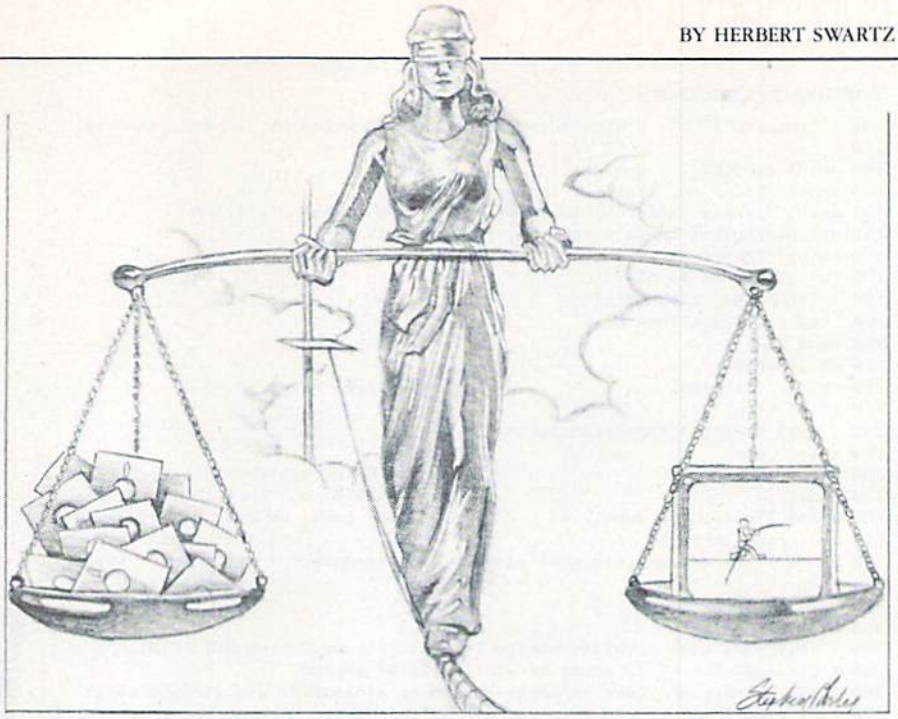
```
100 ! "quicksort:bd". A structured Quicksort. Revision by Jerry W. Carroll.
110
120 print chr$(12); : one%=1
130 input "Enter no. of items to sort: ", n%
140 print : input "Enter filename of list to be sorted: ", file$
150 dim string$(n%+one%) : open #4, file$, input
160 on eof ignore
170 for i% = one% to n%
180   linput #4, string$(i%)
190   if io_status then quit
200 next i%
210 n%=i%-one%
220 reset : t1=time                    ! Reset closes files in V1.1
230 if n% (= 9
240   call straight_insertion_sort
250 else
260   call quicksort                   ! The MAT PRINT statement (line 290) will
270 endif                              ! print chr$(0) and chr$(255) to screen
280 print "Sorting is done ..."        ! at end of sort. We used MAT PRINT to
290 mat print string$                                  ! save space. A loop
300 print "Time to complete sort was:";t2-t1;"seconds." ! will remove these
310 stop                                               ! characters.
320
330 proc quicksort
340 ! This sort will print to screen in five columns, using MAT PRINT.
350 ! string$(n% + 1) is array of strings to be sorted.
360 ! n% = number of items in array excluding string$(0) and string$(n%+1)
370 ! m% = max. no. of items in largest partition for straight insertion sort.
380  print "Sorting has started ..."
390  string$(0)=chr$(0)                ! Lower boundary of array
400  string$(n%+one%)=chr$(255)        ! Upper boundary of array
410  m%=9 : p%=one% : r%=n%            ! Stage Q1
420  loop                              ! Start of new stage Q2
430    loop
440      done%=one% : i%=p% : j%=r%+one%  ! Boolean argument
450      key$=string$(p%)
460      loop                          ! Stage Q3
470        i%=i%+one%
480        while string$(i%) ( key$ and i% (= j%
490          i%=i%+one%
500        endloop
510        j%=j%-one%                  ! Stage Q4
520        while key$ ( string$(j%) and j% )= i%-one%
530          j%=j%-one%
540        endloop
550        if j% (= i%                 ! Stage Q5
560          hold$=string$(p%) : string$(p%)=string$(j%) : string$(j%)=hold$
570        else                        ! Stage Q6
580          hold$=string$(i%) : string$(i%)=string$(j%) : string$(j%)=hold$
590        endif
600      until j% (= i%
610      if r%-j% )= j%-p% and j%-p% ) m%   ! Stage Q7
620        s%=s%+one% : stack%(s%,one%)=j%+one% : stack%(s%,2)=r% : r%=j%-one%
630        r%=j%-one% : done%=0
640      elseif j%-p% ) r%-j% and r%-j% ) m%
650        s%=s%+one% : stack%(s%,one%)=p% : stack%(s%,2)=j%-one%
660        p%=j%+one% : done%=0

670      elseif r%-j% ) m% and m% )= j%-p%
680        p%=j%+one% : done%=0
690      elseif j%-p% ) m% and m% )= r%-j%
700        r%=j%-one% : done%=0
710      endif
720    until done%
730    if s% ) 0
740      p%=stack%(s%,one%) : r%=stack%(s%,2) : s%=s%-one% : done%=0
750    endif
760  until done%
770  call straight_insertion_sort
780 endproc
790
800 proc straight_insertion_sort
810   for j%=2 to n%
820     i%=j%-one% : key$=string$(j%)
830     while key$ ( string$(i%)
840       string$(i%+one%)=string$(i%) : i%=i%-one%
850     until i%=0
860     string$(i%+one%)=key$
870   next j%
880   t2=time
890 endproc
```

END

BY HERBERT SWARTZ

The "Shrink-Wrap" Software License: Protection or Rip-off?



When you buy a piece of software, you often find something called a "license agreement," either in the front of the manual or on a separate card in the package. The agreement usually says something like, "Such-and-such Software Company grants to the user a license to use this piece of software under such-and-such terms and conditions . . ." It probably spells out under what conditions you can make backup copies, and what the limits are for letting other people use the software. It also may include a limited warranty, which may say something like, "This software is sold 'as is' and we aren't responsible if it doesn't work the way you expect it to." (The companies do, however, usually agree to replace the disk if it's damaged. They just don't warranty the operation of the program.)

These so-called "shrink-wrap licenses" are, among other things, an attempt to prevent at least some of the illegal copying that costs software manufacturers hundreds of thousands of dollars in lost revenues each year. Yet some end users contend that these licenses are not legal. If you are a "licensee," rather than an "owner," for instance, you are denied certain rights to use of a product. And, since you don't have access to the license agreement until you open the package—but opening the package constitutes accepting the agreement—the licenses create a real "Catch-22" situation in some people's eyes.

The courts haven't been able to clarify the issue, because, even though suits have been brought against alleged violators of these licenses, they have been settled out of court. Therefore, a case that actually tests the validity of the shrink-wrap license has not yet come to trial.

Here, lawyer Herbert Swartz considers what the shrink-wrap licenses mean for you, the end user.

Since you don't have access to the license agreement until you open the package—but opening the package constitutes accepting the agreement—the licenses create a real "Catch-22" situation.

As the new year begins, questions about the validity of the shrink-wrap license continue to hang over the heads of both software manufacturers and end users. Is that license, as the manufacturers claim, a bonafide legal agreement? Or, as attorneys for users assert, is it a legal nullity, waiting for the first federal judge who gets hold of it to declare it as such?

Two major cases were brought in 1985 that squarely challenged the validity of the licenses, but neither has helped resolve the issue yet. And, as the wait for a judicial decision goes on, one thing is clear: Time is on the side of the software manufacturers and against the end users. For unless the license is declared invalid, users are denied certain legal rights to use of the licensed software.

Rights of Ownership Denied

Despite manufacturers' assertions to the contrary, the shrink-wrap license is designed to avoid two sections of the Copyright Act of 1976—section 109(a) and section 117. Both of these sections apply only to "owners" of a copy of a copyrighted work—in this instance, a computer program. If software buyers are "licensees," not "owners," the rights granted by these two sections of the Act are denied to them.

Why You Can't Rent Software

Section 109(a) is general, rather than computer-specific. It permits the "owner" of a copy of a copyrighted work "without the authority of the copyright owner, to sell or otherwise dispose of the possession of that copy. . . ." It is because of this section that bookstores can rent books, video

stores can rent cassettes, and libraries can exist. "Disposal"—renting or lending—is permitted because the stores or libraries are "owners" of the copies in their establishments. Software buyers, however, because they are merely licensees, are not permitted to dispose of their copies in this manner. That's why you cannot rent most commercial software or run a software lending library that includes licensed programs.

Manufacturers claim they are avoiding section 109(a) in order to prevent piracy. Rentals or loans, they say, make it too easy for people to make unauthorized copies of their programs. If users rent software for a few dollars and make copies, they have saved themselves—and cost the manufacturers—hundreds of dollars.

The manufacturers' need to prevent illegal copying is understandable. Yet, the Copyright Act itself already makes it a crime to copy a program under any circumstances, regardless of whether that program is "owned" or "licensed." So manufacturers are already legally protected, without having to resort to the shrink-wrap licenses. Why should they need to deny legal rights to those who buy their products?

There are many reasons, for instance, that end users should have access to software rentals. Relative to other items, such as records or books, software buyers pay a great deal of money for their products. But, although a record buyer has heard the music on the radio, or can listen to the record in the store, and a book buyer can browse through his or her selection before purchase, a software user is buying "blind." And even though software buyers have legitimate concerns about whether a program is right for them, they are forced to guess. At best, a salesperson will demonstrate the product. A demo disk—not always available, anyway—is usually of little help.

The extent of the software buyer's need to test a program before purchase is explained by Professor Michael Scott of the University of Southern California Law School, author of *Computer Law* and publisher of *Software Protection:* "Using (demo disks) is like trying to see if you want to buy a Ferrari by driving a Volkswagen. While they both have four wheels and go forward and backward, there is no way to know if the 'real thing' will meet your needs without 'taking it out for a spin.'"

NEWS FLASH: ILLINOIS CONDONES SHRINK-WRAP LICENSES

According to *Infoworld,* a microcomputer industry weekly tabloid, a Software Licensing Enforcement Act was recently signed into law by Illinois governor James T. Thompson. The act, modeled after Louisiana's "shrink-wrap" law, is meant to prohibit unauthorized use, duplication and distribution of computer software. It specifically allows license agreements ("shrink-wrap" licenses) to be included in copies of computer software, and makes the agreements enforceable once the software package is opened. Whether any end users will challenge this new law is yet to be seen.

Other Limitations Under the License

Section 117 of the Copyright Act, the other section that the shrink-wrap license is designed to avoid, was passed in 1980 specifically as part of the Software Amendments to the Act. This section says that the "owner" of a copy of a computer program can make "a backup copy," and is permitted "to adapt" their copy of the program.

In most cases, shrink-wrap licenses do allow you to make a backup copy. They do not, however, allow you to tinker with the program to make it suit your needs better. As a "licensee," not an "owner," you cannot legally customize your spreadsheet, speed up your game—or even correct a problem within the program. And even though the licenses do grant you the right to make a backup copy, the question remains: Why should you have to receive from the manufacturer what is already your right by law?

The typical shrink-wrap license also sets other limitations on users. For instance, you cannot use the same copy of the program both at home and in the office. You are allowed to use it on only one computer in one location. You also usually cannot take the program out of the country, so forget traveling with your software.

In addition, the licenses are often attached to restrictive warranties. Under most shrink-wrap licenses, users cannot exchange a program that is found to be unsuitable for their needs, even if the package is promptly returned. Only physically damaged disks will be replaced. Almost all software is sold strictly "as is," with all warranties disclaimed. So users have no recourse, and, unable to pre-test a software product, are at the mercy of what may be exaggerated claims and pure advertising hype.

At this time, many computer law scholars view the shrink-wrap license as a legal nullity. Their reasoning goes like this: No one tells users when they put up their money that they are not "purchasing" a piece of software. The word "licensing" never comes up. And nothing in their experience of going to a store and buying goods prepares them to conclude they are anything other than a purchaser—an owner. Only the license in the package—which users don't see until they get home and open it—tells them they are only a licensee.

Yet, if users have won the battle for the minds of computer law scholars concerning the illegality of shrink-wrap licenses, the war still continues in legislative halls throughout the country. Louisiana, for instance, has passed its own shrink-wrap statute and similar bills are pending in Arizona, California, Georgia, Hawaii, Illinois and Washington. In addition, the "Computer Software Protection Act" continues to kick about in the U.S. Congress. This bill would amend section 109 of the Copyright Act to specifically exclude computer programs. Renting or lending programs would thus be prohibited by federal law.

Manufacturers' concerns about software piracy certainly must be considered. Nevertheless, end users must also be guaranteed their rights under the law. At the moment, however, manufacturers are better organized than users, and, until the shrink-wrap license is tested in court, will undoubtedly continue to use the license to limit users' rights to rent, lend and adapt their personal copies of commercial software products. C

COMMODORE 64 COMPUTER

(Order Now)

$139.95

- C128 Disks 79¢ ea.*
- Commodore Graphics Printer $99.95
- 13" Zenith Color Monitor $149.95

CALL BEFORE YOU ORDER

COMMODORE 64 SYSTEM SALE

Deal 1	Deal 2
Commodore 64	Commodore 64
Com. 1541 Disk Drive	Com. 1541 Disk Drive
Com. 803 Printer	13" Zenith Color Monitor
$407	$457

PLUS FREE $49.95 Oil Barons Adventure Program

C128 COMMODORE COMPUTER

NEW

(Order Now)

Call Us

Less $30 FREE Software, your choice from our catalog (See Catalog Page 13)

- 340K 1571 Disk Drive
- Voice Synthesizer $39.95
- 12" Amber Monitor $59.95

PRICES MAY BE LOWER

COMMODORE 64 COMPUTER $139.95

You pay only $139.95 when you order the powerful 84K COMMODORE 64 COMPUTER! LESS the value of the SPECIAL SOFTWARE DISCOUNT COUPON we pack with your computer that allows you to SAVE OVER $250 off software sale prices!! With only $100 of savings applied, your net computer cost is $39.95!!

* C128 DOUBLE SIDED DISKS 79¢ EA.

Get these 5¼" Double Sided Floppy Disks specially designed for the Commodore 128 Computer (1571 Disk Drive). 100% Certified, ***Lifetime Warranty***. Automatic Lint Cleaning Liner included. 1 Box of 10 - $9.90 (99¢ ea.), 5 Boxes of 10 - $44.50 (89¢ ea.), 10 Boxes of 10 - $79.00 (79¢ ea.).

13" ZENITH COLOR MONITOR $149.95

You pay only $149.95 when you order this 13" ZENITH COLOR MONITOR. LESS the value of the SPECIAL SOFTWARE DISCOUNT COUPON we pack with your monitor that allows you to save over $250 off software sale prices!! With only $100 of savings applied, your net color monitor cost is only $49.95. (16 Colors).

80 COLUMN COMMODORE GRAPHICS PRINTER $99.95

You pay only $99.95 when you order the 803 Commodore Graphics Printer, 60 CPS, Dot Matrix, Bi-Directional, Prints 8½" full size paper. Plug in direct interface included! LESS the value of the SPECIAL SOFTWARE DISCOUNT COUPON we pack with your printer that allows you to SAVE OVER $250 off software sale prices!! With only $100 of savings applied your net printer cost is ZERO!

4 SLOT EXPANDER & 80 COLUMN BOARD $59.95

Now you program 80 COLUMNS on the screen at one time! Converts your Commodore 64 to 80 COLUMNS when you plug in the 80 COLUMN EXPANSION BOARD!! PLUS 4 slot expander!

80 COLUMNS IN COLOR PAPERBACK WRITER 64 WORD PROCESSOR $39.95

This PAPERBACK WRITER 64 WORD PROCESSOR is the finest available for the COMMODORE 64 computer! The ULTIMATE FOR PROFESSIONAL Word Processing, DISPLAYS 40 or 80 COLUMNS IN COLOR or black and white! Simple to operate, powerful text editing, complete cursor and insert/delete key controls line and paragraph insertion, automatic deletion, centering, margin settings and output to all printers! List $99.00. **SALE $39.95.** Coupon $29.95.

SPECIAL SOFTWARE COUPON

We pack a SPECIAL SOFTWARE DISCOUNT COUPON with every COMMODORE 64 COMPUTER, DISK DRIVE, PRINTER, or MONITOR we sell! This coupon allows you to SAVE OVER $250 OFF SALE PRICES!!

(Examples)

PROFESSIONAL SOFTWARE COMMODORE 64

Name	List	Sale	Coupon
Paperback Writer 64	$99.00	$39.95	$29.95
Paperback Database 64	$69.00	$34.95	$24.95
Paperback Dictionary	$24.95	$14.95	$10.00
The Print Shop	$44.95	$27.95	$26.95
Halley's Project	$39.95	$25.95	$24.95
Practicalc (spread sheet)	$59.95	$19.95	$14.95
Programmers Reference Guide	$24.95	$16.95	$12.50
Nine Princes in Amber	$32.95	$24.95	$21.95
Super Bowl Sunday	$30.00	$19.95	$17.95
Flip & File Disk Filer	$24.95	$14.95	$12.95
Deluxe Tape Cassete (plus FREE game)	$89.00	$44.95	$34.95
Pro Joystick	$19.95	$12.95	$10.00
Computer Care Kit	$44.95	$29.95	$24.95
Dust Cover	$ 8.95	$ 6.95	$ 4.60
Injured Engine	$39.95	$27.95	$24.95
Pitstop II (Epyx)	$39.95	$22.95	$19.95
Music Calc	$59.95	$14.95	$12.95
File Writer (by Codewriter)	$39.95	$29.95	$24.95

(See over 100 coupon items in our catalog)

Write or call for Sample SPECIAL SOFTWARE COUPON!

ATTENTION Computer Clubs

We Offer Big Volume Discounts

CALL TODAY!

PROTECTO WARRANTY

All Protecto's products carry a minimum 90 day warranty. If anything fails within 90 days from the date of purchase, simply send your product to us via United Parcel Service prepaid. We will IMMEDIATELY send you a replacement at no charge via United Parcel Service prepaid. This warranty proves once again that ***We Love Our Customers.***

C128 COMMODORE COMPUTER

We expect a limited supply for Christmas. We will ship on a first order basis. This all-new revolutionary 128K computer uses all Commodore 64 software and accessories plus all CPM programs formatted for the disk drive. ***Less $30 FREE Software, your choice from our catalog (See Catalog Page 13)***

340K 1571 COMMODORE DISK DRIVE

Double Sided, Single Disk Drive for C-128 allows you to use C-128 mode plus CPM mode. 17 times faster than 1541, plus runs all 1541 formats.

SUPER AUTO DIAL MODEM $37.95

Easy to use. Just plug into your Commodore 64 computer and you're ready to transmit and receive messages. Easier to use than dialing your telephone, just push one key on your computer! Includes exclusive easy to use program for up and down loading to printer and disk drives. ***Best in U.S.A.*** List $129.00. **SALE $37.95.**

VOICE SYNTHESIZER $39.95

For Commodore-64 computers. Just plug it in and you can program words and sentences, adjust volume and pitch, make talking adventure games, sound action games and customized talkies!! FOR ONLY $19.95 you can add TEXT TO SPEECH, just type a word and hear your computer talk — ADD SOUND TO "ZORK", SCOTT ADAMS AND OTHER ADVENTURE GAMES!! (Disk or tape.) List $89.00. **SALE $39.95**

12" AMBER MONITOR $59.95

Your choice of green or amber screen monitor, top quality, 80 columns x 24 lines, easy to read, anti-glare! PLUS $9.95 for connecting cable. Com-64 or VIC-20. List $159.00. **SALE $59.95.**

PRINTER/TYPEWRITER COMBINATION $249.95

"JUKI" Superb letter quality, daisy wheel printer/typewriter combination. Two machines in one — just a flick of the switch. 12" extra large carriage, typewriter keyboard, automatic margin control and relocate key, drop in cassette ribbon! (90 day warranty) centronics parallel or RS232 serial port built in (Specify). List $349.00. **SALE $249.95.** **(Ltd. Qty.)**

13" RGB & COMPOSITE COLOR MONITOR $259.95

Must be used to get 80 columns in color with 80 column computers (C128 - IBM - Apple). (Add $14.50 shipping) List $399.00. **SALE $259.95.**

- LOWEST PRICES • 15 DAY FREE TRIAL
- BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL

PHONE ORDERS
8 a.m. - 8 p.m. Weekdays
9 a.m. - 12 noon Saturdays

- 90 DAY FREE REPLACEMENT WARRANTY
- OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES EXCEPT CANADA. Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery. 2 to 7 days for phone orders, 1 day express mail! VISA — MASTER CARD — C.O.D. No C.O.D. to Canada, APO-FPO

PROTECTO

We Love Our Customers

Box 550, Barrington, Illinois 60010

312/382-5244 to order

GIANT PRINTER SALE!!

List $399.00 **10" Printer**

SALE **$169.00** SALE

1 Year Warranty

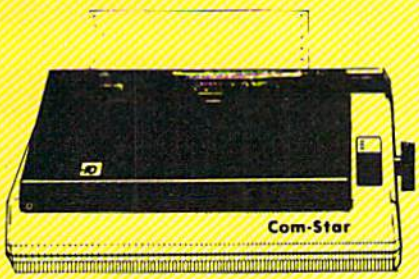
120-140 CPS

Premium Quality

List $599.00 **15½" Printer**

SALE **$239.00** SALE

10" Comstar 10X — This Bi-directional Tractor/Friction Printer prints standard sheet 8½"x11" paper and continuous forms or labels. High resolution bit image graphics, underlining, horizontal tab setting, true lower descenders, with super scripts and subscripts, prints standard pica, compressed, expanded, block graphics, etc. Fantastic value. (Centronics parallel interface.)
List $399.00. **Sale $179.00.**



15½" Comstar 15X — Has all the features of the 10" Comstar 10X plus a wider 15½" carriage and more powerful electronics to handle large ledger business forms! (Better than FX-100). The 15½" Comstar 15X also prints on standard size paper and continuous forms and labels. Fantastic value. (Centronics parallel interface.)
List $599.00. **Sale $239.00.**

List $499.00 **10" Printer**

SALE **$199.00** SALE

1 Year Warranty

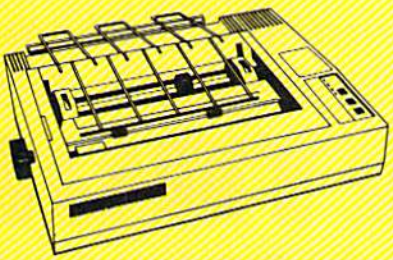
150-170 CPS

High Speed

List $699.00 **15½" Printer**

SALE **$289.00** SALE

10" Comstar 160+ High Speed — This Bi-directional Tractor/Friction Printer combines the above features of the 10" Comstar 10X with speed (150-170 cps) and durability. Plus you get a 2K buffer, 96 user definable characters, super density bit image graphics, and square print pins for clearer, more legible print (near letter quality). This is the best value for a rugged dependable printer. (Centronics parallel interface.)
List $499.00. **Sale $229.00.**



15½" Comstar 160+ High Speed — This Bi-directional Tractor/Friction Printer has all the features of the 10" Comstar 160+ High Speed plus a wider 15½" carriage and the heavy duty electronics required for today's business loads. You can use large ledger business forms as well as standard sheets and continuous forms and labels. This is the best wide carriage printer in the U.S.A. (Centronics parallel interface.)
List $699.00. **Sale $289.00.**

List $599.00 **10" Printer**

SALE **$259.00** SALE

*Lifetime warranty**

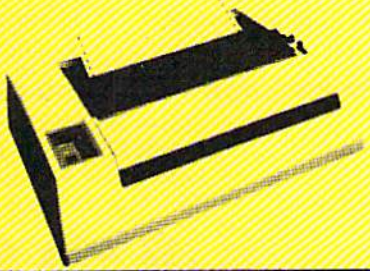
165-185 CPS

High Speed & Letter Quality

List $599.00 **10" Printer**

SALE **$259.00** SALE

10" Comstar 2000 — The ultimate printer has arrived! This Bi-directional Tractor/Friction Printer gives you all the features of the Comstar 160 plus higher speed (165-185 cps), 256 downloadable characters, proportional setting, external dark printing mode and a *lifetime printhead warranty. **PLUS ...**



With the flip of a switch you can go into the letter quality mode which makes all your printing look like it came off a typewriter. Turn in term papers, do articles or just print programs. Have the best of letter quality and speed in one package. Fantastic printer (Centronics parallel interface.)
List $599.00. **Sale $259.00.**

15 Day Free Trial — 1 Year Immediate Replacement Warranty

Parallel Interfaces

Commodore-64, VIC 20 — $39.00 | Atari — $59.00 | Apple II, II+, IIe — $59.00

Add $10.00 ($14.50 for 15½" Printers) for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!

VISA — MASTER CARD — C.O.D. No C.O.D. to Canada, APO-FPO.

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Commodore 64

MODEM

Commodore 64

Sale

SAVE

List $99.00

Sale $29.95

Coupon $24.95

Telecommunications

Lowest Price In the U.S.A.

List $99.00

Sale $37.95

Coupon $32.95

FOR CHILDREN·ADULTS·BUSINESS

Complete Auto Dial Telecommunications Package

"The only telecommunications package you will ever need."

(Exclusive Easy To Use Features)

FREE VIEWTRON DATABASE MEMBERSHIP

• Only Good Color Graphic Database Service in the U.S.A. (C-64)

Viewtron Software Plus First Hour FREE (See the Protecto Software Catalog On-Line) $9.95 value

• 300 Baud Modem • Auto Dial • Auto Answer • Upload & Download

Reach Out and Access Someone

- Educational courses
- Financial Information
- Banking at Home
- Popular Games
- News Updates and Information
- Electronic Shopping
- Research and Reference Materials

The Complete Telecommunications Package offers you all this plus ...

- Auto Log-on
- Dialing from Keyboard
- On-line Clock
- Capture and Display High Resolution Characters
- Download Text, Program or Data Files
- Stores on Disk Downloaded Files
- Reads Files from Disk and Uploads Text or Program Files
- Select Any Protocol (access almost any computer or modem)
- Plus Much, Much More

List $99.00 **Sale $37.95** *Coupon $32.95*

We are so sure this is the only telecommunications package you will need we will give you 15 days Free Trial.

Viewtron Membership sold separately — $9.95.

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!

VISA — MASTER CARD — C.O.D.

No C.O.D. to Canada APO-FPO

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

COLOR MONITOR SALE!!!

Sale

16 COLORS

(Premium Quality)

- **Built in Speaker and Audio**
- **Front Panel Controls**
- **For Video Recorders**
- **For Small Business Computers**
- **Apple - Commodore - Atari - Franklin - etc.**

(Premium Quality)

- **Beautiful Color Contrast**
- **High Resolution**
- **Separated Video**
- **Sharp Clear Text**
- **Anti-Glare Screen**
- **40 Columns x 24 Lines**
- **Supports 80 Columns**

Requires special $19.95 C-128 cable

List $329.00

RGB

IBM, C-128, Apple, Atari ST

13" Color Computer Monitor

*C64/C128 connecting cable $9.95

Sale $159.95*

Add $14.50 Shipping

13" RGB & COMPOSITE COLOR MONITOR

Allows use of C-128 and C64 mode - composite and 80 column RGB mode. Must be used to get 80 columns in color with 80 column computers (IBM, C-128, Apple & Atari ST). (add $14.50 shipping)

List $399.00 **Sale $259.95***

* C128 RGB cable $19.95

13" ZENITH COLOR MONITOR

(add $14.50 shipping)

List $299.00 **Sale $149.95***

15 Day Free Trial - 90 Day Immediate Replacement Warranty

12" XTRON SUPER HI-RESOLUTION MONITOR

80 Columns x 24 Lines, Super Hi-Resolution 800 lines Green or Amber Super-Clear "Easy to Read" text with special anti-glare screen! *(Ltd. Qty.)* List $249.00

Sale $109.95*

12" ZENITH HI-RESOLUTION GREEN OR AMBER TEXT DISPLAY MONITOR

80 Columns x 24 Lines, Hi-Resolution, crisp clear easy to read text with anti-glare screen! A MUST for word processing. *(Ltd. Qty.)* List $199.00

Sale $89.95*

12" AMBER MONITOR

80 Columns x 24 Lines, easy to read up front controls *(Ltd. Qty.)*

List $159.00 **Sale $59.95***

* C64/C128 connecting cable $9.95

• LOWEST PRICES • BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL • OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

FLOPPY DISKS SALE *59¢ ea.

Economy Model or C-128 Cadillac Quality

We have the lowest prices!

***ECONOMY DISKS** — For use with Commodore 64, Atari, Apple.

Good quality 5¼" single sided double density with hub rings.

Bulk Pac	100 Qty.	59¢ ea.	Total Price	$59.00
Box w/ sleeves	10 Qty.	79¢ ea.	Total Price	7.90

☆ C-128 Computer Disks ☆

CADILLAC QUALITY (Double Sided, Double Density) ***Specifically designed for use with C-128***

• Each disk certified • Free replacement lifetime warranty • Automatic dust remover • Works with IBM PC

For those who want cadillac quality we have the C-128 Floppy Disk. Used by professionals because they can rely on C-128 Disks to store important data and programs without fear of loss! Each C-128 disk is 100% certified (an exclusive process) plus each disk carries an exclusive FREE REPLACEMENT LIFETIME WARRANTY. With C-128 disks you can have the peace of mind without the frustration of program loss after hours spent in program development.

100% CERTIFICATION TEST

Some floppy disk manufactures only sample test on a batch basis the disks they sell, and then claim they are certified. Each C-128 disk is individually checked so you will never experience data or program loss during your lifetime!

FREE REPLACEMENT LIFETIME WARRANTY

We are so sure of C-128 Disks that we give you a free replacement warranty against failure to perform due to faulty materials or workmanship for as long as you own your C-128 disk.

AUTOMATIC DUST REMOVER

Just like a record needle, disk drive heads must travel hundreds of miles over disk surfaces. Unlike other floppy disks the C-128 smooth surface finish saves disk drive head wear during the life of the disk. (A rough surface will grind your disk drive head like sandpaper). The lint free automatic CLEANING LINER makes sure the disk-killers (dust & dirt) are being constantly cleaned while the disk is being operated.

C-128 Disks are definitely the Cadillac disk in the world

Just to prove it even further, we are offering these super LOW INTRODUCTORY PRICES

1 Box of 10 — $9.90 (99¢ ea.) 5 Boxes of 10 — $44.50 (89¢ ea.) 10 Boxes of 10 — $79.00 (79¢ ea.)

All disks come with hub rings and sleeves in an attractive package.

DISK DRIVE CLEANER $19.95

Everyone Needs A Disk Drive Doctor

FACTS

- 60% of all drive downtime is directly related to poorly maintained drives.
- Drives should be cleaned each week regardless of use.
- Drives are sensitive to smoke, dust & all micro particles.
- Systematic operator performed maintenance is the best way of ensuring error free use of your computer system.

NEW RSI-SMARTCARE HEAD CLEANING KIT WITH FREE LIBRARY CASE!

Improper maintenance can cause costly read/write errors on your disk drive. Regular cleaning of heads is essential for maintaining high quality operation and eliminating expensive downtime and repairs. The RSI-Smartcare Dry Process Head Cleaning Kit provides proper maintenance without the mess. Simply insert the cleaning disk into the drive and activate for 30 seconds. This kit contains 1-open ended disk jacket and 2 cleaning pads, good for a total of 30 cleanings. Applicable for both single and double sided drives.

List $29.95. **Sale $19.95.** *Coupon $12.95.*

RSI-COMPLETE PERSONAL COMPUTER CARE KIT!

The RSI Complete Personal Computer Care Kit contains everything you need to properly maintain your computer system. This easy to use kit makes necessary computer care quick and inexpensive. In one convenient package you will get all the necessary products: 5¼" Dry Process Head Cleaning Disk (Good for 15 thorough cleanings); Anti-Static Spray (Eliminates static build-up. For use on carpets, clothes, paper, glass, hard surfaces, and other material); Screen-Terminal Cleaner (Removes smoke, dust, and other contaminants on computer screens and keyboards); Lint-Free Cloths (Non-scratch, non-abrasive, used for cleaning and applying sprays to computers); Foam-Covered Swabs (use to clean those hard to reach places such as keyboards, etc.). GREAT VALUE FOR THE MONEY!! List $44.95. **Sale $29.95.** *Coupon $24.95.*

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery. 2 to 7 days for phone orders, 1 day express mail!

VISA — MASTER CARD — C.O.D.

No C.O.D. to Canada, APO-FPO.

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Commodore 64 COMPUTER

$139.95*

COMMODORE 64 SYSTEM SALE

SAVE

Deal 1
Commodore 64
Com. 1541 Disk Drive
Com. 803 Printer
$407
PLUS FREE $49.95 Oil Barrons Adventure

Deal 2
Commodore 64
1541 Disk Drive
13" Zenith Color Monitor
$457
PLUS FREE $49.95 Oil Barrons Adventure

$30 **$30**

FREE SOFTWARE COUPON CM45

SUPER OFFER (Expires 1-1-86) ***SUPER OFFER***

To introduce you to the C128 computer, we are offering you **YOUR FIRST $30 OF SOFTWARE** from the enclosed catalog ***FREE*** when you buy a ☐ C128 Computer from Protecto. Just send this coupon along with your order. Choose software worth $30 or more at sale prices from this catalog and subtract $30 from the total.

$30 Complete software order must accompany C128 purchase. (1 Coupon per family) Software From This Catalog Only **$30**

* **Commodore C128 Computer $289.00.** This all-new revolutionary 128K computer uses all Commodore 64 computer software, CPM Software, plus new advanced C-128 software. You pay only $289 for the C128 computer! **Less** the value of the **Special Software Discount Coupon** (see page 14) we pack with your computer that allows you to **Save Over $250** off software sale prices!! With only $100 of savings applied your net computer cost is $189.00. ***PLUS $30.00 Your Choice FREE Software.*****

* **340K 1571 Commodore Disk Drive $259.00.** Double Sided, Single Disk Drive for C-128 allows you to use C-128 mode plus CPM mode. 17 times faster than the 1541, plus runs all 1541 formats. You pay only $259.00 for the 340K 1571 Commodore Disk Drive. **Less** the value of the **Special Software Discount Coupon** (see page 14) we pack with your Disk Drive that allows you to **Save Over $250** off software sale prices! With only $100 of savings applied your net Disk Drive cost is only $159.00.

*** Less Coupon Discounts**
(See page 14)

More Coupon Items (Ltd. Qty.)	List	Sale	Coupon
1. Disk Drive Cleaner	$29.95	**$19.95**	***$14.95***
2. HES Games (disk)	$29.95	**$19.95**	***$14.95***
3. Pogo Joe (tape or disk)	$29.95	**$19.95**	***$14.95***

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery. 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

USER GROUPS

Commodore user groups nationwide and around the world provide invaluable assistance to Commodore computerists. If you are looking for people who share your computing interests, or if you need help getting started with your computer, contact the group near you.

This list is compiled from groups who responded to a survey conducted by Pete Baczor, Commodore's user group coordinator. If you would like your group to appear here, or if you need information about Commodore's user group support, contact Pete at Commodore Business Machines, 1200 Wilson Drive, West Chester, PA 19380.

ALABAMA

Birmingham
Birmingham Commodore Computer Club
Harry Jones
4845 Avenue V Lot 7-B—35208

Enterprise
Wiregrass Micro-Computer Society
William H. Brown
109 Key Bend Rd.—36330

Mobile
Commodore Club of Mobile
Tom Wyatt
3868-H Rue Maison—36608

Pinson
Commodore Club—South
William J. Freeman
P.O. Box 324—35126

Scottsboro
Scottsboro C-64 Users' Group
Richard Radon
Rt. 5, Box 255—35768

ALASKA

Calgary
Calgary Commodore Users Group
Lloyd Norman
810 Canfield Way SW—T2W 1K1

Anchorage
Alaska 84 Commodore Computer Club
Doug Gannon
P.O. Box 6043—99502

Ketchikan
First City Users Group
Jim Llanos
P.O. Box 6002—99901

Sitka
Sitka Commodore Users Group
Dennis Lanham
P.O. Box 2204—99835

ARIZONA

Chandler
Arizona VIC & 64 Users
Thomas Monson
904 W. Marlboro Cir.—85224

Globe
Gila Hackers
Paul R. Machula
Rt. 1, Box 34—85501

Phoenix
Phoenix Arizona Commodore Club
Leslie A. Roberts
P.O. Box 34905—85067

Sierra Vista
Thunder Mountain Commodore Users Gp.
Jeff Tyler
P.O. Box 1796—85636

Tucson
Catalina Commodore Computer Club
Mike LaPaglia
3371 S. Mormon Dr.—85730

ARKANSAS

Conway
Commodore/PET Users Club
Geneva Bowlin
Davis Street—72032

Russellville
Russellville Commodore Users Group
Bob Brazeal
401 S. Arlington Dr.—72801

CALIFORNIA

Altaloma
Cactus Communicator
P.O. Box 1277—91701

Arcadia
Pasadena Commodore Computer Club
Ernie McDonald
P.O. Box 1163—91006

Auburn
Auburn Commodore Computer Club
Pat Strub
11210 Mira Loma Dr.—95603

Baldwin Park
Cal Poly Commodore Users Group
David Dietzel
14617½ Ramona Blvd.—91706

Bellflower
Club 64
Robert Johnson
14944 Bayou Ave.—90706

Chatsworth
San Fernando Valley Commodore Users Group
Tom Lynch
21208 Nashville—91311

Clovis
Fresno 64 Users
Greg Edwards
091 W. 9th #203—93612

Concord
Diablo Valley Commodore Users Group
Dean Steichen
P.O. Box 27155—94520

Costa Mesa
C Tech Users of Orange County
R. Stevenson
P.O. Box 1497—92626

Cupertino
PUG of Silicon Valley
Marvin Vander Kool
22355 Rancho Ventura—95014

Danville
Danville Users Group
Kent E. Davis
185 Front St., Suite 106—94526

Hollywood
Hollywood Commodore Users Group
Martin Blackwell
733 N. Ridgewood Blvd.—90038

Livermore
PALS
Jo Johnson
886 South K—94550

Los Osos
San Luis Obispo Commodore Comp. Cl.
Gary Bissell
1766 9th Street—93402

Nevada City
C64 Users Group
Gerry Friederici
11614 Red Dog Road—95959

Ojai
64/20 Club
Greg Burns
215 N. Fulton St.—93023

Oxnard
CIVIC 64
Nathan Okum
120 W. Magnolia Ave.—93030

Pasadena
P.C.A.C.C.
Brian Klotz
1135 Coronet Ave.—91107

San Carlos
Peninsula Commodore Users Group
Timothy Avery
543 Old County Rd.—94070

San Francisco
PET on the AIR
Bola Lowe
525 Crestluine Dr.—94132

San Ysidro
South Bay Commodore 64 Users Group
Jose M. Gallego
P.O. Box 3193—92073

Santa Maria
Central Coast Commodore User Group
Gilbert F. Vela
4237 Plumeria Ct.—93455

Santa Monica
Commodore 64 West
Don Campbell
P.O. Box 406—90406-0406

Santa Rosa
Santa Rosa Commodore 64 Users Group
Garry Palmer
333 E. Robles Ave.—95407-7925

Stockton
Stockton Commodore Users Group
Rita McCann
2929 Calariva Dr.—95204

COLORADO

Aspen
Rocky Mountain Commodore Club
Ray Brooks
P.O. Box 377—81612

Denver
Colorado Commodore Computer Club
Royce Mitchell
9400 Elm Ct. #6078—80221

Grand Junction
Western Slope Commodore Users Group
Ralph D'Andrea
P.O. Box 4142—81502

CONNECTICUT

Colchester
Computer Users Group
Liz Rafolowski
Halls Hill Road—06415

Danbury
Fairfield County Commodore User Gp.
Kenneth H. Hottes
P.O. Box 212—06810

East Hartford
Hartford County Commodore User Gp.
R. Curtiss
P.O. Box 8553—06018

Enfield
Northern CT Commodore Club
Mike Levesque
3 Marble St.—06082

Hampton
Commodore East Computer Users Group
Ernest Julian
165 B.S. Bigelow Rd.—06247

Mystic
C64 Users Group of New London
A. Estes
57 Fishtown Ln.—06355

Stamford
Stamford Commodore Users Group
Stan Czernel
P.O. Box 1337—06902

Stratford
Commodore User Group of Stratford
Dan Kern-Ekins
P.O. Box 1213—06497

West Haven
U of New Haven Microcomputer Club
M. McConeghy
U of New Haven, 300 Orange Ave.—06516

DELAWARE

Felton
Diamond State Users Group
Michael Butler
P.O. Box 892—19943

Newark
Newark Commodore Users Group
Bob Taylor
210 Durso Dr.—19711

FLORIDA

Avon Park
CHIPS Users Group
Gerald Klimczak
P.O. Box 3063—33825

Bradenton
Manasota Commodore Users Group
Bob Bronson
916 E. 35th Ave. Dr.—33508

Brandon
Brandon Users Group
Paul A. Daugherty
108 Anglewood Dr.—33511

Clearwater
Clearwater Commodore Club
Gary B. Gould
1532 Lemon Street—33516

Cocoa
El-shift-oh
Mike Schnoke
P.O. Box 548—32922-0548

Eaton Park
VIC/64 Heartland User Group—Lakeland
Randall Lipham
P.O. Box 912—33840

Gainsville
Gainsville Commodore Users Group
M. Drew Hurley
P.O. Box 14716—32604

Jacksonville
UNF Commodore Computer Club
Jay S. Huebner
University of North Florida—32216

Lutz
VIC/64 Heartland Users—Tampa
Tom Keough
14806 Morning Dr.—33549

MacDill, AFB
South Tampa Commodore 64 Users Group
Ronald S. Clement
736F Second Dr.—33621

Miami
Miami 2064
Jim Luftman
12911 SW 49th St.—33175

North Miami Beach
Commodore Educators User Group—North
Robert Figueroa
16330 NE 2nd Ave.—33162

Ocala
Tri-County Commodore Users Group
Don Vanderventer
P.O. Box 1151—32671

Orlando
Central Florida Commodore Users Group
Thurman Lawson
P.O. Box 7326—32854

Palm Harbor
Suncoast 64s
Curtis J. Miller
2419 US 19 North—33563

Panama City
Bay Commodore Users Group
Doug Schipall
P.O. Box 3187—32401

Pensacola
Commodore Advantage
Deanna Owens
P.O. Box 18490—32523

Pensacola
Commodore Users Group of Pensacola
Debbie Johnston
P.O. Box 3533—32506

Plantation
Suburban 64 Users Group
Ken Partridge
1360 SW 82nd Ter. Apt. #625—33324

Sanibel
Sanibel Commodore VIC
Phil Belanger
1119 Periwinkle Box 73—33957

Shalimer
Ft. Walton Beach Commodore User Group
Jim Moore
P.O. Box 3—32549

Tamarac
Broward Commodore Users Group
Lewis Horn
P.O. Box 25794—33320

Tavares
Triangle Educational Users Group
John Zeigler
P.O. Box 326—32778

Zephyrhill
East Pasco Commodore Users Group
P.J. Bryden
1836 Sandy Road—34248

GEORGIA

Albany
Albany CBM Amatuer Computerist
David L. Via
P.O. Box 5461—31706

Atlanta
VIC-tims
Earl Dobney
P.O. Box 467052—32346

Brunswick
Golden Isles Commodore Users
Richard L. Young
135 Sherwood Forest Cir.—31520

College Park
Commodore Format
Gregory Young
2804 Camp Creek Pkwy.—30337

Morrow
CCC 64 UG
David Padett
P.O. Box 842—30260

Oakwood
NE Georgia Commodore Users Group
Randy Shuler
Route #2, Box 226—30566

USER GROUPS

Riverdale
Clayton County C64 Users Group
Joyce Jay
527 Wanelyn Way—30274

Stowe Mountain
Atlanta C64 Users Group
Ronald Lisoski
1767 Big Valley Lane—30083

IDAHO

Coeur d'Alene
Coeur d'Alene Computer Club
Gary Edwards
506 Lunceford Ln.—83814

Grangeville
Grangeville HS Computer Club
Don Kissinger
910 S. 'D' Street—83530

Idaho Falls
Eagle Rock Commodore Computer Club
Jeanie Larson
P.O. Box 3884—83403-3884

ILLINOIS

Belvidere
Heuristics Users Group
Chip Downey
1303 Nettle #2—61008

Bloomington
Bloomington-Normal Commodore ug
Carl Burress
P.O. Box 1058—61702-1058

Canton
Canton Area Commodore Users Group
Robert S. Smolich
Spoon River College, RR #1—61520

Champaign
Champaign Urbana Commodore Users Group
Steve Gast
2006 Crescent Dr.—61821

Elgin
Fox Valley Commodore Users Group
Herb Gross
833 Prospect—60120

Galesburg
Knox Commodore Club
Randy Fox
195 Olive St.—61401

Harvey
South Suburban V20 Users Club
Nick Forte
16409 Emerald—60426

Kankakee
Kankakee Hackers
Bill Brouillet
R.R. #2, Box 228-H—60901

Lincolnwood
C-64 Users Group, Inc.
David W. Tamkin
P.O. Box 46464—60646

Monmouth
Western Illinois Commodore Users Group
Robert Cokel
906 W. 6th Ave.—61462

Peoria
Commodore Users of Peoria
Jeff Martin
1600 W. Devereux—61614

Rockford
Rockford Area Commodore Computer Club
Kathe Heyer
2507 Pelham Rd.—61107

Scott AFB
Scott Computer Users' Group
Gilbert D. Helland
P.O. Box 397—62225

Springfield
Springfield PET Users Group
Bill Eardley
3116 Concord—62704

INDIANA

Columbus
Columbus Commodore Club
Frank Nicholas
11210 W. Georgetown Rd.—47201

Indianapolis
VIC Indy Club
Fred Imhausen
Box 11543, 4812 E. Michigan St.—46201

N. Manchester
Manchester Commodore Users Group
Richard M. Bellows
6060 E. 9th St.—46902

Terra Haute
Western Indiana Commodore Users Group
Dennis C. Graham
912 E. Brown Ave.—47803

W. Lafayette
Commodore Owners of Lafayette
Ross Indelicato
20 Patrick Lane—47906

IOWA

Des Moines
Commodore Computer Users of Iowa
Curtis L. Shiffer
P.O. Box 3140—50316

Hartford
Capital Complex Commodore Computer Club
Doren Hulet
P.O. Box 58—50118

Waterloo
Waterloo Area Commodore Club
Rick Volker
945 Lowel—50702

KANSAS

Hutchinson
Salt City Commodore Club
Wendell D. Hinkson
P.O. Box 2644—67504

Viola
Commodore Users Group of Wichita, Inc.
Walt Lounsbery
Rt. #1,Box 115—67149

KENTUCKY

Bardstown
Commodore Bardstown Users Group
Patrick Kirtley
P.O. Box 165—40004

Bowling Green
Bowling Green Commodore Users Group
Doug Skinner
P.O. Box 20214—42102

Glasgow
Glasgow Commodore Users Group
Steve England
P.O. Box 154—42141

Henderson
Commodore Connection
James Kemp
1010 S.Elm—42420

LOUISIANA

Metairie
Sixty-Four 'Um Users' Group
Elizabeth S. Hoffman
4317 Stockton St.—70001

New Orleans
Control Data Institute
Kent H. Courtney II
6600 Plaza Dr.—70127

Shreveport
Ark-La-Tex Commodore 64 Club
Bill Walker
5515 Fairfax Ave.—71108

Sulphur
Southwest LA Users' Group
Fred White
99 Ann—70663

Swartz
Ouachita Commodore Club
Beckie Walker
P.O. Box 175—71281

MAINE

Brunswick
Y.U.G. (Brunswick)
Peter O'Brien
20 Columbia Ave.—04011

Orano
Computer Society of Penobscot Art Pete
101 Crosby Lab, UMO—04469

Portland
Southern Maine 20/64
Ed Moore
10 Walker Street—04092

Scarborough
Y.U.G. (Scarborough)
George Caswell
16 Westwood—04074

Westbrook
Your Commodore Users Group
Mike Procise
P.O. Box 611—04092

MARYLAND

Aberdeen
COMPUCATS
Betty Scheuler
680 W. BelAir Ave.—21001

Baltimore
Baltimore Area Commodore Uses Group
Michael M. Broumberg
4605 Vogt Ave.—21206

Westinghouse BWI Commodore User Group
Lee D. Barron
P.O. Box 1693—21203

Woodlawn Commodore Club
Mike Long
1712 Aberdeen Rd.—21234

Fallston
Harford County Commodore Users Group
Kim Lloyd
P.O. Box 209—21047

Frederick
Frederick Functioneers
Stephen M.Jacobson
108 Key Pkwy.—21701

Hagerstown
Hagerstown Users Group
Joseph F. Rutkowski
23 Coventry Ln.—21740

Leonardtown
PAX/COM/64
David J. Hamblin
240 Jefferson St.—20650

Potomac
Western Montgomery C64 Users Group
Jorge Montalvan
11209 Tack House Ct.—20854

Rockville
RockvilleVIC/64 Users Group
Tom Pounds
P.O. Box 8805—20856

Silver Springs
Montgomery County Commodore Soc.
Meryle B. Pounds
P.O. Box 6444—20906

Suitland
Edison Commodore Users Group
William H.Harr
4314 Oxford Dr.—20746

MASSACHUSETTS

Bedford
Raytheon Commodore Users Group
John Rudy
Raytheon Co.—GRA6—017730

Foxboro
Foxboro Area Commodore User Group
Samuel Knight
154 Green St.—02035

Rutland
The Polyhedral Wizard
Rob Hodgon Jr.
422 Countryside Est.—01543

Westfield
Pioneer Valley VIC/64 Club
Marvin S. Yale
34 Bates St.—01085

Woburn
EM 20/64 Users Group
John Chaplain
36 Buckman St.—01801

MICHIGAN

Ann Arbor
Commodore User Group—U of Mich.
John J. Gannor
School of Pub. Health—U of Mich.—48109

Bay City
Bay City Users Group
Jim Griffin
1013 N. Johnson St.—48706

Clare
Mid-Michigan Commodore Club
Virgil Graham
—48617

East Detroit
Michigan Commodore 64 User Group, Inc.
Chuck Ciesliga
P.O. Box 539—48021

Madison Heights
Slipped Disk, Inc.
J. Moskow
31044 John R—48071

Midland
Commodore Computer Club
Jack Walley
4106 West Man Rd.—48460

Richmond
COMP
Brian Pringle
7514 Putty Gutt Rd.—48062

Southfield
Commodore Computer Enthusiasts
Steven Lepsetz
20050 Winchester—48076

Watervliet
DAB Computer Club
Dennis Burlington
P.O. Box 542—49098

MINNESOTA

Little Falls
Heartland Area Computer Co-op
Sam Walz
2401 Riverwood Dr.—56345

Redwood Falls
Interface
Ron Schilling
243 E. 2nd St.—56283

MISSISSIPPI

Biloxi
Gulf Coast Commodore Club
Mark W. Harvey
4550 W. Beach Blvd.—39531

Hattiesburg
Commodore Computer Club
Len J. Mathias
Dept. of Poli. Sci.—U of Miss.—39406-0076

MISSOURI

Cape Girardeau
Heartland Users Group
Mason Emerson
129 S. Lorimer #7—64701

Golden
MOARK Commodore Users Group
Marshall B. Turner
P.O. Box 504—65658

Joplin
Joplin Commodore Computer Users
R.D. Connely
422 S. Florida Ave.—64801

Kansas City
Commodore Users Group of Kansas City
Salvador Cerda
P.O. Box 36492—64111

Liberty
Northland Amateur Radio Association
Alan Boyer
811 Lindenwood Lane—64068

Maryville
Commodore PAC
Patricia Lucido
Horace Mann Learning Center—64468

Warrensburg
Commodore User Group of Warrensburg
Buck Sommerkamp
P.O. Box 893—64093

MONTANA

Missoula
Western Montana Commodore User Group
Carl White
800 Kensington Ave.—59801

NEBRASKA

Alliance
Alliance Commodore Computer Club
M. Seller
1629 Boise—69301

Gering
Platte Valley Commodore Users Group
Jim Parks
1720 'O' St.—69341

Lincoln
Computer Power Unlimited
Robert Howard
416 N.27—68503

Omaha
Greater Omaha C64 Users Group
Ken Jirele
2932 Leawood Dr.—68123

NEVADA

Las Vegas
Las Vegas Area PET Users Group
5130 E. Charleston Blvd.—89122

Las Vegas
SOG Commodore Users Group
Lee Schram
4011 W. Charleston Blvd.—89102

NEW HAMPSHIRE

Rochester
C-64 U.S.E.R.S.
P.O. Box 4022—03867

Winchester
Monadnock Commodore 64 Users Group
Paul Rule III
RFD #1 Route 10—03470

NEW JERSEY

Belle Mead
ACGNJ PET/VIC/CBM Users Group
Joseph M. Pylka
30 Riverview Ter.—08502

Cherry Hill
South Jersey Commodore Users Group
Fred Herman
P.O. Box 4205—08034

Freehold
F.T.C.U.
John Rinaldi
150 Starling Ave.—07728

Morris Plains
Morris Area Commodore Users Group
Robert J. Searing
51 Ferncliff Rd.—07950

USER GROUPS

Morris Plains
N. American Commodore Teachers Users Group
Robert J. Searing
51 Fernciff Rd.—07950

Mount Holly
RCA Commodore Users Group
William Rowe
432 Hemlock Ln.—08060

Ocean View
Cape-Atlantic Commodore Users Group
B.J. Chadwick
1440 Old Stage Coach Rd.—08230

Ridgewood
INFO 64
Pete Nissely
16 W. Ridgewood Ave.—07432

Union City
Hudson County Commodore User Group
Dave Westphalen
308 Palisade Ave.—07087

Wall Township
Jersey Shore Commodore Users Group
Bob McKinley
1905 Monmouth Blvd.—07728

Wayne
Club 64
G. Cipolletti
Hamburg Tpke.—07470

Westwood
Commodore 64 Beginners Club
Thomas Martin
680 Leigh Ter.—076765

NEW YORK

APO New York
Zweibruecken Commodore Computer Club
Henry White
200th TAMMC, Box 114—09052
06332-6347

Bethpage
Commodore Computer Club
N. Threuisen
1111 Stewart Ave.—11714

Bronx
Commodore Microchips
Anthony Sanchez
340 E. Mosholu Pkwy.—10458

Brooklyn
Brooklyn Commodore Users Group
Malcolm Jay Gottesman
1735 E. 13th St. Apt. #7N—11229

East Setauket
Long Island VIC Society
Lawrence V. Stefani
20 Spyglass Lane—11733

Great Neck
Great Neck Commodore Users Group
Adam Korn
30 Grace Ave.—11021

Great Neck
LIVE
A. Friedman
17 Picadilly Road—11023

Hammondsport
Finger Lakes Area Komputer Experts
Terry Lynne Narby
86 W. Lake Rd.—14840

Jaberg
Commodore Freaks
Mike West
9111 Railroad St.—13471

Mt. Vernon
Folklife Terminal Club
Ted Corales
Box 2222-C—10551

New Hampton
C-64 Users Group of Orange County
Stephen Gerstl
Box 238 RD #1—10958

New Platz
Computer Metropolis
Eric Dean
P.O. Box 39—171 Main St.—12561

New York
New York Commodore Users Group
Ben Tunkelang
380 Riverside Dr. 7Q—10025

New York City V20/C64 Users Group
Ralph Lees
103 Waverly Place—10011

Newark
Finger Lakes Commodore Users Group
Todd Andrews
229 W. Union St.—14513

Oswego
Oswego Commodore Users Group
John R. Boronkay
208 Park Hall—SUNY—13126

Rochester
Commodore Users Group of Rochester
P.O. Box 26514—14626

Saratoga Springs
Adirondack C64 Users Group
Paul W. Klompas
208 Woodlawn Ave.—12866

Syracuse
Commodore Computer Club of Syracuse
P.O. Box 2232—13220

Tallman
Computer Club of Rockland
Peter Bellin
P.O. Box 233—10982

Tonawanda
64 Southtowns Users Group
Michael Leskow
504 Somerville—14150

Tonawanda
Commodore Buffalo Users Group
David Newton
199 Dale Dr.—14150

Tribes Hill
Mohawk Valley Commodore Users Group
William A. Nowak
3818 Stinson Ave., P.O. Box 343—12177

NORTH CAROLINA

Charlotte
Charlotte C64 Users Group
H. S. Hanks
3717 Flowerfield Rd.—28210

Havelock
Down East Commodore Users Group
Jim Worthington
302 Belltown Rd.—28532

Tryon
Tryon Commodore 64 Club
Robin Michael
P.O. Box 1016—28782

NORTH DAKOTA

Bismarck
Commodore Club of North Dakota
James G. Allen
16067 Reno Dr.—58501

OHIO

Brook Park
C.P.U. Connection
Burke Frank
P.O. Box 42032—44142

Cincinnati
Commodore Users of Blue Chip
Ted Stalets
816 Beecher St.—45206

Maple Heights
SE Cleveland Commodore Crazies Users Group
Carl Skala
18873 Harlan Dr.—44137

Marion
Marion Ohio Commodore Users Group
Richard Munro
775 Wolfinger Rd.—43302

Mentor
NE Ohio Commodore Users Group
Ross Black
P.O. Box 718—44062

Milford
Clercom-64
Dave Skaggs
5771 Observation Ct.—45150

New Philadelphia
Commodore 64 Users Group
Jeff Eklund
702 Park Ave. NW—44663

Toledo
Commodore Computer Club of Toledo
Jim Cychlen
P.O. Box 8909—43623

Xenia
Dayton Area Commodore Users Group
Charles Tobin
679 Murray Hill Dr.—45385

OKLAHOMA

Muskogee
Muskogee Computer Soc. Commodore Users Group
Steve Ford
202 S. 12th Street—74401

Oklahoma City
Greater Oklahoma Commodore Club
Richard P. Hill
1401 N. Rockwell—73127

OREGON

Albany
Albany Corvallis C64 Users Group
Al Rasmus
2138 Chicago St. SE—97321

Portland
NE Portland Commodore User Group
Gary A. Thompason
8759 N.Calvert Ave.—97217

Portland
PGE Commodore Users Group
Richard Turnock
121 SW Salmon St.—97005

Prineville
Central Oregon C64 Users Group
Marvin McCleary
499 N. Harwood Ave.—97754

Sheridan
Yamhill County Commodore Users Group
Dan Linscheid
Route 2, Box 246—97378

PENNSYLVANIA

Altoona
Altoona & Martinsburg Commodore Users Group
Ralph Osmolinski Jr.
R.D. #4 Box 258—16601

Altoona
B.A.S.I.C.
Dimitri N. Dantos
1433 13th Ave.—16603

Apollo
Hackers Elite Club
Joe Moore
305 Route 356—15613

Berwyn
GE Com64
Edward Davidson
507 Westwind Dr.—19312

Blue Bell
Worldwide Commodore Users Group
Dave Walter
P.O. Box 337—19422

Clarks Summit
Scranton Commodore Users Group
Mark A. Davis
P.O. Box 211—18411

Clifton Heights
C.H.U.G.
Kevin J. Daly
P.O. Box 235—19018

Greensburg
Westmoreland Computer Users Club
Gary Means
3021 Ben Venus Dr.—15601

Morrisville
Lower Bucks Users Group
Don Moyer
252 Mansfield Rd.—19067

Mount Holly Springs
Eight Squared
Andy Skelton
P.O. Box 76—17065

New Kensington
A-K 64 Users Group
Alton Glubish
1762 Fairmount St.—15068

Philadelphia
Boeing Emp. Personal Computer Club
Alina McNichol
P.O. Box 16858 P32-31—19142

Pittsburgh
Bettis Commodore Users Group
Bill Campbell
592 Arbor Lane—15236

Pittsburgh Commodore Group
Joel A. Casar
2015 Garrick Dr.—15235

V.U.G.O.P.
Davin Flateau
1931 Rockledge—15212

Salona
Central PA User Group for Commodore
Joseph W. Coffman
Box 102—17767

West Chester
Main Line Commodore Users Group
Emil J. Volcheck Jr.
1046 General Allen Ln.—19382

West Mifflin
South Pittsburgh Commodore Group
Charles W. Groves III
2407 Pennsylvania Ave.—15122

RHODE ISLAND

Harrisville
Burrillville Commodore Users Club
David R. Migneault
28 Cherry Farm Rd.—02850

Narragansett
Narragansett Commodore Users Group
Robert R. Payne
South Ferry Rd.—02882

Pawtucket
R.I.C.E.
Michael Skeldon
198 Morris Ave.—02860

SOUTH CAROLINA

Cayce/W. Columbia
Commodore Computer Club of Columbia
Buster White
P.O. Box 2775—29171

Myrtle Beach
Commodore Update
Brad Bannon
230 Tarpon Bay—29577

North Charleston
Charleston Computer Society
Jack A. Furr Jr.
P.O. Box 5264—29406

Rock Hill
Rock Hill Commodore User Group
Smarzik Robbie
565 Scaleybark Cir.—29730

Spartanburg
SPARCUG
James B. Pasley
385 S. Spring St.—29302

TENNESSEE

Bartlett
Memphis Commodore Users Club
Steven A. Gaines
P.O. Box 38095—38134-0095

Estill Springs
Commodore Computer Club
Marty Garner
P.O. Box 96—37330

Knoxville
ET 64 Users Group
Rick McCall
P.O. Box 495—37901

Metro Knoxville Commodore Users Group
Edward Pritchard
7405 Oxmoor Rd.—37931

Soddy-Daisy
C64/VIC 20/+4 Club
Aaron Kennedy
2414 Blue Ridge Dr.—37379

TEXAS

Austin
Commodore Computer Club of Austin
Roy Holmess
P.O. Box 49138—78765

Corpus Christi
Corpus Christi Commodores
Jim O'Rear
P.O. Box 6541—78411

Dallas
64 Users Group, Inc.
Stan Gordin
P.O. Box 801828
13604 Midway Rd.—75380

Hurst
Mid Cities Commodore Club
Diane Dews
413 Chisolm Trail—76053

Irving
Irving Commodore Users Group
Bill Marshall
P.O. Box 165034—75016

Longview
Longview Users Group
Dorothy Metzler
P.O. Box 9284—75608

Lubbock
Lubbock Commodore Users Group
Robert Massengale
3817 64th Drive—79413

Mexia
BiStone Users Club
R.G. Gore
P.O. Box 386—76667

Victoria
Crossroads Commodore Users Group
Jerry Guy
417 Irma Dr.—77901

Waco
Computas
Burl A. Hays
Route 4 Box 214—76705

UTAH

Ogden
Wasatch Commodore Users Group
Mike Murphy
P.O. Box 4028—84402

VERMONT

S. Burlington
Champlain Valley Commodore Users Group
Steve Lippert
6 Mayfair St.—05401

USER GROUPS

VIRGINIA

Alexandria
Alexandria Users Group
Jeffrey K. Hendrickson
1206 Westgrove Blvd —22307

Franconia Commodore
Users Group
Mark Sowash
5924 Dovee Dr —22310

Arlington
Arlington VICtims (20/64)
Clifton Gladley
4501 Arlington Blvd —22204

Crystal City Commodore 64 Club
Greg Selezynski
1235 Jefferson Davis Hwy
Ste 1200—22202

Dale City
Dale City Commodore
Users Group
Jack B. Doyle
P.O. Box 2004—22193-0058

Fairfax
PENTAF
Ron Thomas
9912 Colony Rd —22030

Franklin
Commodore Users of Franklin
Bruce Powell
1201 N. High St.—23851

Hardy
Roanoke Computer Society
Larry Washburn
Rt. 1, Box 40—24101

Hurt
Central Virginia Users Group
Dale Moore
Route 3, Box 451—24563

McLean
C64 Users Group of McLean
John Mueller
7426 Eldorado St.—22102

Roanoke
R.A.C.E. Commodore
Users Group
Larry Rackow
4726 Horseman Dr.—24019

WASHINGTON

Arlington
Arlington Commodore
Users Group
Jane M. King
18204 67th Ave. NE—98223

Centralia
CBM Users Group
Rick Beaber
803 Euclid—98531

Ft. Lewis
64 PCS
Jim Litchfield
Quarters 2821-A—98433

Marysville
Bunch 'A' Bytes
Joey Miller
4916 121st Place NE—98272

Oak Harbor
Whidbey Island Commodore
Computer Club
Bob Hardy
P.O. Box 1471, #203—98277

Puyallup
PNB Users Group
Rod Small
9105 158th St. E—98373

Richland
Tri City Commodore
Computer Club
Jack Garvin
1926 Pine St.—99352

Seattle
C.O.M.P.U.T.E.R.
Art Witbeck
5303 Shilshale Ave. NW—98107

NNW Commodore Users Group
Richard Ball
2565 Dexter N. #203—98109

University 64 Users Group
Allen M. Grown
Pathology SM-30
Univ. of Washington—98195

Tacoma
World Wide Users Group
R. Smith
P.O. Box 98682—98498

WISCONSIN

Caledonia
C.L.U.B. 84
Jack White
P.O. Box 72—53108

Eau Claire
Eau Claire Area CBM 64
Users Group
John F. Slasky Jr.
Rte. 5 Box 179A—54703

Green Bay
Comm Bay 64
Richard F. Luxton
P.O. Box 1152—54305

Madison
Madison Area Commodore
Users Group
Napolian Smith
1850 Beld #11—53713

Menomonie
Menomonie Area Commodore
Users Gp.
Mike Williams
510 12th St.—54725

Milwaukee
M.A.C.E.
John Postman
6332 W. Manitoba St.—53219

Racine
C.U.S.S.H.
Tim Trammel
3614 Soverign Dr.—53406

WYOMING

Cheyenne
Cheyenne Assn. of
Comp. Enthusiasts
William Holden
505 Williams St. #260—82007

Lander
ZYMEC Users Group
Tim Struna
Rte 63, Box 357
11 Birchfield Ln.—82520

CANADA

Belleville
Baden Computer Club
Ben Brash
P.O. Box 1219—CFPO 5056—
K0K 3R0
07229-3791

Fargus
CWDHS Users Group
Jerry Rogerson
155 Belsyde St. East—N1M 1Y7

Hamilton
TRACK 64
Wayne Chapman
491 Kenilworth Ave. N.—
H4H 4T6

Kimberley
Kootenay Computer Club
Tom Lang
348 Archibald St.—V1A 1M9

Montreal
C64 Users Group of Montreal
Robert Adler
Snowdon P.O. Box 792—
H3X 3X9

St. Catherines
Niagara Commodore
Users Group
Ian Kerry
44 Queenston St. Unit 12—
L2R 2Y9

Stoney Creek
Eastdale Commodore
Users Group
R.W. Harasen
275 Lincoln Rd.—L8E 1Z4

Toronto
Toronto PET Users Group
Chris Bebbett
1912A Avenue Rd. Suite 1—
M5M 4A1

Victoriaville
C.P. 884
Michael Laganiere
—G6P 8Y1

MEXICO

Mexico City D.F.
Club Herra*Tec C64
Alain Bojmal
Vicente Suarez 25—06140

Commodore 128 Memory Map

This is a preliminary version of the Commodore 128 memory map, for those of you who have been clamoring for the information. A more complete version will appear in the *Commodore 128 Programmer's Reference Guide* by Larry Greenly, soon to be available from Bantam Books. Look for it in your local bookstore.

```
0000    D6510       ;6510 DATA DIRECTION REGISTER
0001    R6510       ;6510 DATA REGISTER

0002    BANK        ;TOKEN 'SEARCH' LOOKS FOR, OR BANK #
0003    PC_HI       ;FOR SYS, LONG CALL/JUMP ROUTINES
0004    PC_LO       ;ADDRESS, STATUS, A-REG, X-REG, Y-REG
0005    S_REG       ;
0006    A_REG       ;BASIC SYS COMMAND     OR
0007    X_REG       ;MONITOR AND LONG CALL/JUMP
0008    Y_REG       ;
0009    STKPTR
```

```
                    ;BASIC ZERO PAGE STORAGE
0009    INTEGR
        CHARAC      ;SEARCH CHARACTER
000A    ENDCHR      ;FLAG: SCAN FOR QUOTE AT END OF STRING
000B    TRMPOS      ;SCREEN COLUMN FROM LAST TAB
000C    VERCK       ;FLAG: 0=LOAD, 1=VERIFY
000D    COUNT       ;INPUT BUF.PTR / # OF SUBSCRIPTS
000E    DIMFLG      ;FLAG: DEFAULT ARRAY DIMENSION
000F    VALTYP      ;DATA TYPE: $FF=STRING, $00=NUMERIC
0010    INTFLG      ;DATA TYPE: $00=FLOAT.PT, $80=INTEGER
0011    GARBFL      ;FLAG: DATA SCAN / LIST QUOTE / GARB.COLL.
        DORES
0012    SUBFLG      ;FLAG: SUBSCRIPT REF. / USER FUNC. CALL
0013    INPFLG      ;FLAG: $00=INPUT, $40=GET, $98=READ
0014    DOMASK
        TANSGN      ;FLAG: TAN SIGN / COMPARISON RESULT
0015    CHANNL
0016    POKER
        LINNUM      ;TEMP INTEGER VALUE
0018    TEMPPT      ;POINTER: TEMP STRING STACK
0019    LASTPT      ;LAST TEMP STRING ADDRESS
001B    TEMPST      ;STACK FOR TEMP STRINGS
0024    INDEX       ;UTILITY POINTER AREA
        INDEX1
0026    INDEX2
0028    RESHO       ;FLOAT.PT. PRODUCT OF MULTIPLY
0029    RESMOH
002A    ADDEND
        RESMO
002B    RESLO
002D    TXTTAB      ;POINTER: START OF BASIC TEXT
002F    VARTAB      ;POINTER: START OF BASIC VARIABLES
0031    ARYTAB      ;POINTER: START OF BASIC ARRAYS
0033    STREND      ;POINTER: END OF BASIC ARRAYS + 1
0035    FRETOP      ;POINTER: BOTTOM OF STRING STORAGE
0037    FRESPC      ;UTILITY STRING POINTER
0039    MAX_MEM_1   ;TOP OF STRING/VARIABLE BANK (BANK 1)
```

```
                        ;BASIC ZERO PAGE STORAGE

003B    CURLIN          ;CURRENT BASIC LINE NUMBER
003D    TXTPTR          ;POINTER TO BASIC TEXT USED BY CHRGET,ETC.
003F    FORM            ;USED BY PRINT USING
        FNDPNT          ;POINTER TO ITEM FOUND BY SEARCH
0041    DATLIN          ;CURRENT DATA LINE NUMBER
0043    DATPTR          ;CURRENT DATA ITEM ADDRESS
0045    INPPTR          ;VECTOR: INPUT ROUTINE
0047    VARNAM          ;CURRENT BASIC VARIABLE NAME
0049    FDECPT
        VARPNT          ;POINTER: CURRENT BASIC VARIABLE DATA
004B    LSTPNT
        FORPNT          ;POINTER: INDEX VARIABLE FOR FOR/NEXT
        ANDMSK
004C    EORMSK
004D    VARTXT
        OPPTR
004F    OPMASK
0050    GRBPNT
        TEMPF3
        DEFPNT
0052    DSCPNT
0054                    ;?????
0055    HELPER          ;FLAGS 'HELP' OR 'LIST'
0056    JMPER
0057                    ;?????
0058    OLDOV
0059    TEMPF1
        PTARG1          ;MULTIPLY DEFINED FOR INSTR
005B    PTARG2
005D    STR1
0060    STR2
0063    POSITN
0064    MATCH
005A    ARYPNT
        HIGHDS
005C    HIGHTR
005E    TEMPF2
005F    DECCNT          ;NUMBER OF DIGITS AFTER THE DECIMAL POINT
0060    TENEXP
0061    GRBTOP
        DPTFLG          ;DECIMAL POINT FLAG
        LOWTR
0062    EXPSGN
0063    FAC
        DSCTMP
        LEFT_FLAG       ;PAINT-LEFT FLAG
        FACEXP          ;FAC#1 EXPONENT
0064    RIGHT_FLAG      ;PAINT-RIGHT FLAG
        FACHO           ;FAC#1 MANTISSA
0065    FACMOH
0066    INDICE
        FACMO
0067    FACLO
0068    FACSGN          ;FAC#1 SIGN
0069    DEGREE
```

```
         SGNFLG       ;POINTER: SERIES-EVAL. CONSTANT
006A     ARGEXP       ;FAC#2 EXPONENT
006B     ARGHO        ;FAC#2 MANTISSA
006C     ARGMOH
         INIT_AS_0    ;JUST A COUNT FOR INIT
006D     ARGMO
006E     ARGLO
006F     ARGSGN       ;FAC#2 SIGN
0070     STRNG1
         ARISGN       ;SIGN COMPARISON RESULT: FAC#1 VS #2
0071     FACOV        ;FAC#1 LOW-ORDER (ROUNDING)
0072     STRNG2
         POLYPT
         CURTOL
         FBUFPT       ;POINTER: CASSETTE BUFFER
0074     AUTINC       ;INC. VAL FOR AUTO (0=OFF)
0076     MVDFLG       ;FLAG IF 10K HIRES ALLOCATED
0077     Z_P_TEMP_1   ;PRINT USING'S LEADING ZERO COUNTER
                      ;MOVSPR & SPRITE TEMPORARY
                      ;MID$ TEMPORARY
0078     HULP         ;COUNTER
         KEYSIZ
0079     SYNTMP       ;USED AS TEMP FOR INDIRECT LOADS
007A     DSDESC       ;DESCRIPTOR FOR DS$
007D     TOS          ;TOP OF RUN TIME STACK
007F     RUNMOD       ;FLAGS RUN/DIRECT MODE
0080     PARSTS       ;DOS PARSER STATUS WORD
         POINT        ;USING'S POINTER TO DEC.PT
0081     PARSTX
0082     OLDSTK
-------------------------------------------------------------
                      ;BASIC Z-P STORAGE FOR GRAPHIC COMMANDS
0083     COLSEL       ;CURRENT COLOR SELECTED
0084     MULTICOLOR_1
0085     MULTICOLOR_2
0086     FOREGROUND
0087     SCALE_X      ;SCALE FACTOR IN X
0089     SCALE_Y      ;SCALE FACTOR IN Y
008B     STOPNB       ;STOP PAINT IF NOT BACKGROUND/NOT SAME COLOR
008C     GRAPNT
008E     VTEMP1
008F     VTEMP2
-------------------------------------------------------------
                      ;KERNAL/EDITOR STORAGE
0090     STATUS       ;I/O OPERATION STATUS BYTE
0091     STKEY        ;STOP KEY FLAG
0092     SVXT         ;TAPE TEMPORARY
0093     VERCK        ;LOAD OR VERIFY FLAG
0094     C3P0         ;SERIAL BUFFERED CHAR FLAG
0095     BSOUR        ;CHAR BUFFER FOR SERIAL
0096     SYNO         ;CASSETTE SYNC #
0097     XSAV         ;TEMP FOR BASIN
0098     LDTND        ;INDEX TO LOGICAL FILE
0099     DFLTN        ;DEFAULT INPUT DEVICE #
009A     DFLTO        ;DEFAULT OUTPUT DEVICE #
009B     PRTY         ;CASSETTE PARITY
009C     DPSW         ;CASSETTE DIPOLE SWITCH
```

```
009D    MSGFLG      ;OS MESSAGE FLAG
009E    PTR1        ;CASSETTE ERROR PASS1
        T1          ;TEMPORARY 1
009F    PTR2        ;CASSETTE ERROR PASS2
        T2          ;TEMPORARY 2
00A0    TIME        ;24 HOUR CLOCK IN 1/60TH SECONDS
00A3    R2D2        ;SERIAL BUS USAGE
        PCNTR       ;CASSETTE STUFF
00A4    BSOUR1      ;TEMP USED BY SERIAL ROUTINE
        FIRT
00A5    COUNT       ;TEMP USED BY SERIAL ROUTINE
        CNTDN       ;CASSETTE SYNC COUNTDOWN
00A6    BUFPT       ;CASSETTE BUFFER POINTER
00A7    INBIT       ;RS-232 RCVR INPUT BIT STORAGE
        SHCNL       ;CASSETTE SHORT COUNT
00A8    BITCI       ;RS-232 RCVR BIT COUNT IN
        RER         ;CASSETTE READ ERROR
00A9    RINONE      ;RS-232 RCVR FLAG FOR START BIT CHECK
        REZ         ;CASSETE READING ZEROES
00AA    RIDATA      ;RS-232 RCVR BYTE BUFFER
        RDFLG       ;CASSETTE READ MODE
00AB    RIPRTY      ;RS-232 RCVR PARITY STORAGE
        SHCNH       ;CASSETTE SHORT CNT
00AC    SAL         ;POINTER: TAPE BUFFER / SCREEN SCROLLING
00AD    SAH
00AE    EAL         ;TAPE END ADDRESSES / END OF PROGRAM
00AF    EAH
00B0    CMP0        ;TAPE TIMING CONSTANTS
00B1    TEMP
00B2    TAPE1       ;ADDRESS OF TAPE BUFFER
00B4    BITTS       ;RS-232 TRNS BIT COUNT
        SNSW1
00B5    NXTBIT      ;RS-232 TRNS NEXT BIT TO BE SENT
        DIFF
00B6    RODATA      ;RS-232 TRNS BYTE BUFFER
        PRP
00B7    FNLEN       ;LENGTH CURRENT FILE N STR
00B8    LA          ;CURRENT FILE LOGICAL ADDR
00B9    SA          ;CURRENT FILE 2ND ADDR
00BA    FA          ;CURRENT FILE PRIMARY ADDR
00BB    FNADR       ;ADDR CURRENT FILE NAME STR
00BD    ROPRTY      ;RS-232 TRNS PARITY BUFFER
        OCHAR
00BE    FSBLK       ;CASSETTE READ BLOCK COUNT
00BF    DRIVE
        MYCH        ;SERIAL WORD-BUFFER
00C0    CAS1        ;CASSET MANUAL/CNTRLED SWITCH (UPD'D DURING IRQ)
00C1    TRACK
        STAL        ;I/O START ADDRESS (LO)
00C2    SECTOR
        STAH        ; " " (HI)
00C3    MEMUSS      ;CASSETTE LOAD TEMPS (2 BYTES)
        TMP2
00C5    DATA        ;TAPE READ/WRITE DATA
00C6    BA          ;BANK FOR CURRENT LOAD/SAVE/VERIFY OPERATION
00C7    FNBANK      ;BANK WHERE CURRENT FN IS FOUND (AT 'FNADR')
00C8    RIBUF       ;RS-232 INPUT  BUFFER POINTER
```

```
00CA    ROBUF       ;RS-232 OUTPUT BUFFER POINTER
-------------------------------------------------------------------
                    ;GLOBAL SCREEN EDITOR VARIABLES

00CC    KEYTAB      ;KEYSCAN TABLE POINTER
00CE    IMPARM      ;PRIMM UTILITY STRING POINTER
00D0    NDX         ;INDEX TO KEYBOARD QUEUE
00D1    KYNDX       ;PENDING FUNCTION KEY FLAG
00D2    KEYIDX      ;INDEX INTO PENDING FUNCTION KEY STRING
00D3    SHFLAG      ;KEYSCAN SHIFT KEY STATUS
00D4    SFDX        ;KEYSCAN CURRENT KEY INDEX
00D5    LSTX        ;KEYSCAN LAST    KEY INDEX
00D6    CRSW        ;<CR> INPUT FLAG
00D7    MODE        ;40/80 COLUMN MODE FLAG
00D8    GRAPHM      ;TEXT/GRAPHIC MODE FLAG
00D9    CHAREN      ;RAM/ROM VIC CHARACTER FETCH FLAG (BIT-2)
-------------------------------------------------------------------
                    ;THE FOLLOWING LOCATIONS ARE SHARED
                    ;BY SEVERAL EDITOR ROUTINES.
00DA    SEDSAL      ;POINTERS FOR MOVLIN
00DC    SEDEAL      ;
00DE    SEDT1       ;SAVPOS
00DF    SEDT2       ;

00DA    KEYSIZ      ;PROGRAMMABLE KEY VARIABLES
00DB    KEYLEN      ;
00DC    KEYNUM      ;
00DD    KEYNXT      ;
00DE    KEYBNK      ;
00DF    KEYTMP      ;

00DA    BITMSK      ;TEMPORARY FOR TAB & LINE WRAP ROUTINES
00DB    SAVER       ;ANOTHER TEMPORARY PLACE TO SAVE A REGIS.
-------------------------------------------------------------------
                    ;LOCAL SCREEN EDITOR VARIABLES.  THESE ARE
                    ;SWAPPED OUT TO $0A40 WHEN SCREEN 40/80
                    ;MODE CHANGES.
00E0    PNT         ;POINTER TO CURRENT LINE (TEXT)
00E2    USER        ;POINTER TO CURRENT LINE (ATTRIBUTE)
00E4    SCBOT       ;WINDOW LOWER LIMIT
00E5    SCTOP       ;WINDOW UPPER LIMIT
00E6    SCLF        ;WINDOW LEFT  MARGIN
00E7    SCRT        ;WINDOW RIGHT MARGIN
00E8    LSXP        ;CURRENT INPUT COLUMN START
00E9    LSTP        ;CURRENT INPUT LINE   START
00EA    INDX        ;CURRENT INPUT LINE   END
00EB    TBLX        ;CURRENT CURSOR LINE
00EC    PNTR        ;CURRENT CURSOR COLUMN
00ED    LINES       ;MAXIMUM NUMBER OF SCREEN LINES
00EE    COLUMNS     ;MAXIMUM NUMBER OF SCREEN COLUMNS
00EF    DATAX       ;CURRENT  CHARACTER TO PRINT
00F0    LSTCHR      ;PREVIOUS CHAR PRINTED  (FOR <ESC> TEST)
00F1    COLOR       ;CURR ATTRIBUTE TO PRINT (DEFAULT FGND COLOR)
00F2    TCOLOR      ;SAVED ATTRIB TO PRINT  ('INSERT' & 'DELETE')
00F3    RVS         ;REVERSE MODE FLAG
00F4    QTSW        ;QUOTE MODE FLAG
00F5    INSRT       ;INSERT MODE FLAG
```

```
00F6    INSFLG      ;AUTO-INSERT MODE FLAG
00F7    LOCKS       ;DISABLES  <C=><SHIFT>,   <CTRL>-S
00F8    SCROLL      ;DISABLES  SCREEN SCROLL, LINE LINKER
00F9    BEEPER      ;DISABLES  <CTRL>-G

00FA    FREKZP      ;FREE ZERO PAGE RESERVED FOR
                    ;APPLICATIONS SOFTWARE ($FA-$FE)
00FF    LOFBUF      ;
```

```
                    ;MONITOR ZERO PAGE STORAGE
                    ;IN BASIC AREA

0002    PCB         ;PC TEMP
0003    PCH         ;PC TEMP
0004    PCL         ;PC TEMP
0005    FLGS        ;STATUS TEMP
0006    ACC         ;ACC  TEMP
0007    XR          ;XREG TEMP
0008    YR          ;YREG TEMP
0009    SP          ;STACK POINTER TEMP
0060    T0          ;MONITOR ZP STORAGE IN FAC
0063    T1          ;
0066    T2
007A    TXTPTR
```

```
                    ;BASIC/DOS INTERFACE VARS
0100    BAD         ;TAPE READ ERRORS
        FBUFFR      ;AREA TO BUILD FILENAME IN (16 BYTES)
0110    XCNT        ;DOS LOOP COUNTER
0111    DOSF1L      ;DOS FILENAME 1 LEN
0112    DOSDS1      ;DOS DISK DRIVE 1
0113    DOSF2L      ;DOS FILENAME 2 LEN
0114    DOSDS2      ;DOS DISK DRIVE 2
0115    DOSF2A      ;DOS FILENAME 2 ADDR
0117    DOSOFL      ;BLOAD/BSAVE STARTING ADDRESS
0119    DOSOFH      ;....AND ENDING ADDRESS
011B    DOSLA       ;DOS LOGICAL ADDR
011C    DOSFA       ;DOS PHYS ADDR
011D    DOSSA       ;DOS SEC. ADDR
011E    DOSRCL      ;DOS RECORD LENGTH
011F    DOSBNK      ;
0120    DOSDID      ;DOS DISK ID
0122    DIDCHK      ;DOS DSK ID FLG
```

```
                    ;SPACE USED BY PRINT USING
0123    BNR         ;POINTER TO BEGIN. NO.
0124    ENR         ;POINTER TO END NO.
0125    DOLR        ;DOLLAR FLAG
0126    FLAG        ;COMMA FLAG
0127    SWE         ;COUNTER
0128    USGN        ;SIGN EXPONENT
0129    UEXP        ;POINTER TO EXPONENT
012A    VN          ;# OF DIGITS BEFORE DECIMAL POINT
012B    CHSN        ;JUSTIFY FLAG
012C    VF          ;# OF POS BEFORE DECIMAL POINT (FIELD)
012D    NF          ;# OF POS AFTER DECIMAL POINT (FIELD)
```

```
012E    POSP        ;+/- FLAG (FIELD)
012F    FESP        ;EXPONENT FLAG (FIELD)
0130    ETOF        ;SWITCH
0131    CFORM       ;CHAR COUNTER (FIELD)
0132    SNO         ;SIGN NO
0133    BLFD        ;BLANK/STAR FLAG
0134    BEGFD       ;POINTER TO BEGIN OF FIELD
0135    LFOR        ;LENGTH OF FORMAT
0136    ENDFD       ;POINTER TO END OF FIELD

0137    SYSTK       ;SYSTEM STACK ($0137-$01FF)

0200    BUF         ;INPUT BUFFER: BASIC & MONITOR($0200-A1)

02A2    FETCH       ;LDA(-),Y FROM ANY BANK
02AA    FETVEC      ;
02AF    STASH       ;STA(-),Y TO ANY BANK
02B9    STAVEC      ;
02BE    CMPARE      ;CMP(-),Y TO ANY BANK
02C8    CMPVEC      ;
02CD    JSRFAR      ;JSR XXXX TO ANY BANK & RETURN
02E3    JMPFAR      ;JMP XXXX TO ANY BANK
```

```
                    ;VECTORS
02FC    ESC_FN_VEC  ;VECTOR FOR ADDITIONAL FUNCTION ROUTINES
02FE    BNKVEC      ;VECTOR FOR FUNCTION CART. USERS
0300    IERROR      ;VECTOR FOR PRINT BASIC ERROR (ERR IN .X)
0302    IMAIN       ;VECTOR TO MAIN (SYSTEM DIRECT LOOP)
0304    ICRNCH      ;VECTOR TO CRUNCH (TOKENIZATION ROUTINE)
0306    IQPLOP      ;VECTOR TO LIST BASIC TEXT (CHAR LIST)
0308    IGONE       ;VECTOR TO GONE (BASIC CHAR DISPATCH)
030A    IEVAL       ;VECTOR TO BASIC TOKEN EVALUATION
030C    IESCLK      ;VECTOR TO ESCAPE-TOKEN CRUNCH,
030E    IESCPR      ;                        ...LIST,
0310    IESCEX      ;                ...AND EXECUTE.
0312                ;NOT USED
0314    IIRQ        ;IRQ RAM VECTOR
        CINV
0316    IBRK        ;BRK INSTR RAM VECTOR
        CBINV
0318    INMI        ;NMI VECTOR
031A    IOPEN       ;KERNAL OPEN ROUTINE VECTOR
031C    ICLOSE      ;KERNAL CLOSE ROUTINE VECTOR
031E    ICHKIN      ;KERNAL CHKIN ROUTINE VECTOR
0320    ICKOUT      ;KERNAL CHKOUT ROUTINE VECTOR
0322    ICLRCH      ;KERNAL CLRCHN ROUTINE VECTOR
0324    IBASIN      ;KERNAL CHRIN ROUTINE VECTOR
0326    IBSOUT      ;KERNAL CHROUT ROUTINE VECTOR
0328    ISTOP       ;KERNAL STOP ROUTINE VECTOR
032A    IGETIN      ;KERNAL GETIN ROUTINE VECTOR
032C    ICLALL      ;KERNAL CLALL ROUTINE VECTOR
032E    EXMON       ;MONITOR COMMAND VECTOR
0330    ILOAD       ;KERNAL LOAD ROUTINE VECTOR
0332    ISAVE       ;KERNAL SAVE ROUTINE VECTOR
```

```
                        ;EDITOR INDIRECT VECTORS

0334    CTLVEC          ;EDITOR: PRINT 'CONTRL' INDIRECT
0336    SHFVEC          ;EDITOR: PRINT 'SHIFTD' INDIRECT
0338    ESCVEC          ;EDITOR: PRINT 'ESCAPE' INDIRECT
033A    KEYVEC          ;EDITOR: KEYSCAN LOGIC  INDIRECT
033C    KEYCHK          ;EDITOR: STORE KEY INDIRECT
033E    DECODE          ;VECTORS TO KEYBOARD MATRIX DECODE TABLES
034A    KEYD            ;IRQ KEYBOARD BUFFER (10 BYTES)

                        ;TABMAP AND BITABL GET SWAPPED TO $0A60
                        ;WHEN SCREEN 40/80 MODE IS CHANGED.
0354    TABMAP          ;BITMAP OF TAB STOPS  (10 BYTES, $0354-D)
035E    BITABL          ;BITMAP OF LINE WRAPS

0362    LAT             ;LOGICAL FILE NUMBERS
036C    FAT             ;PRIMARY DEVICE NUMBERS
0376    SAT             ;SECONDARY ADDRESSES
0380    CHRGET          ;
0386    CHRGOT          ;
0390    QNUM            ;

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

                        ;INDIRECT LOAD SUBROUTINE AREA
039F    INDSUB_RAM0     ;SHARED ROM FETCH SUB
03AB    INDSUB_RAM1     ;SHARED ROM FETCH SUB
03B7    INDIN1_RAM1     ;INDEX1 INDIRECT FETCH
03C0    INDIN2          ;INDEX2 INDIRECT FETCH
03C9    INDTXT          ;TXTPTR
03D2    ZERO            ;NUMERIC CONSTANT FOR BASIC
03D5    CURRENT_BANK    ;CONTEXT FOR SYS,POKE,PEEK FROM BANK CMMD
03D6    TMPDES          ;TEMP FOR INSTR
03DA    FIN_BANK        ;BANK POINTER FOR STRING/NUMBER CONVERT RTN
03DB    SAVSIZ          ;TEMP WORK LOCATIONS FOR SSHAPE
03DF    BITS            ;FAC#1 OVERFLOW DIGIT
03E0    SPRTMP_1        ;TEMP FOR SPRSAV
03E1    SPRTMP_2
03E2    FG_BG           ;PACKED FOREGROUND/BACKGROUND COLOR NYBBLES
03E3    FG_MC1          ;PACKED FOREGROUND/MULTICOLOR 1 COLOR NYBBLES

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

                        ;PAGE FOUR & HIGHER  DECLARATIONS
                        ;(BEGINNING OF BANKABLE RAM)

0400    VICSCN          ;VIDEO MATRIX #1: VIC 40-COLUMN TEXT SCREEN
                        ;$0400-$07FF
0800                    ;BASIC RUN-TIME STACK (512 BYTES)
                        ;$0800-$09FF

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

                        ;ABSOLUTE  KERNAL VARIABLES

0A00    SYSTEM_VECTOR   ;VECTOR TO RESTART SYSTEM (BASIC WARM)
0A02    DEJAVU          ;KERNAL WARM/COLD INIT'N STATUS BYTE
0A03    PALNTS          ;PAL/NTSC SYSTEM FLAG
0A04    INIT_STATUS     ;FLAGS RESET VS. NMI STATUS FOR INIT'N RTNS
0A05    MEMSTR          ;PTR TO BOTTOM OF AVAIL. MEMORY IN SYSTEM BANK
0A07    MEMSIZ          ;PTR TO TOP OF AVAILABLE  MEMORY IN SYSTEM BANK
0A09    IRQTMP          ;TAPE HANDLER PRESERVES IRQ INDIRECT HERE
0A0B    CASTON          ;TOD SENSE DURING TAPE OPERATIONS
```

```
0A0C    KIKA26      ;TAPE READ TEMPORARY
0A0D    STUPID      ;TAPE READ D1IRQ INDICATOR
0A0E    TIMOUT      ;FAST SERIAL TIMEOUT FLAG
0A0F    ENABL       ;RS-232 ENABLES
0A10    M51CTR      ;RS-232 CONTROL REGISTER
0A11    M51CDR      ;RS-232 COMMAND REGISTER
0A12    M51AJB      ;RS-232 USER BAUD RATE
0A14    RSSTAT      ;RS-232 STATUS REGISTER
0A15    BITNUM      ;RS-232 NUMBER OF BITS TO SEND
0A16    BAUDOF      ;RS-232 BAUD RATE FULL BIT TIME(CREATED BY OPEN)
0A18    RIDBE       ;RS-232 INPUT BUFFER INDEX TO END
0A19    RIDBS       ;RS-232 INPUT BUFFER INDEX TO START
0A1A    RODBS       ;RS-232 OUTPUT BUFFER INDEX TO START
0A1B    RODBE       ;RS-232 OUTPUT BUFFER INDEX TO END
0A1C    SERIAL      ;FAST SERIAL INTERNAL/EXTERNAL FLAG
0A1D    TIMER       ;DECREMENTING JIFFIE REGISTER
-------------------------------------------------------------------------
                    ;GLOBAL ABSOLUTE SCREEN EDITOR DECLARATIONS

0A20    XMAX        ;KEYBOARD QUEUE MAXIMUM SIZE
0A21    PAUSE       ;<CTRL>-S FLAG
0A22    RPTFLG      ;ENABLE KEY REPEATS
0A23    KOUNT       ;DELAY BETWEEN KEY REPEATS
0A24    DELAY       ;DELAY BEFORE A KEY STARTS REPEATING
0A25    LSTSHF      ;DELAY BETWEEN <C=><SHFT> TOGGLES
0A26    BLNON       ;VIC CURSOR MODE (BLINKING, SOLID)
0A27    BLNSW       ;VIC CURSOR DISABLE
0A28    BLNCT       ;VIC CURSOR BLINK COUNTER
0A29    GDBLN       ;VIC CURSOR CHARACTER BEFORE BLINK
0A2A    GDCOL       ;VIC CURSOR COLOR BEFORE BLINK
0A2B    CURMOD      ;VDC CURSOR MODE (WHEN ENABLED)
0A2C    VM1         ;VIC TEXT SCREEN/CHARACTER BASE POINTER
0A2D    VM2         ;VIC BIT-MAP BASE POINTER
0A2E    VM3         ;VDC TEXT SCREEN BASE
0A2F    VM4         ;VDC ATTRIBUTE BASE
0A30    LINTMP      ;TEMPORARY POINTER TO LAST LINE FOR LOOP4
0A31    SAV80A      ;TEMPORARY FOR 80-COL ROUTINES
0A32    SAV80B      ;TEMPORARY FOR 80-COL ROUTINES

0A33    CURCOL      ;VDC CURSOR COLOR BEFORE BLINK
0A34    SPLIT       ;VIC SPLIT SCREEN RASTER VALUE
0A35    FNADRX      ;SAVE .X DURING BANK OPERATIONS
0A36    PALCNT      ;COUNTER FOR PAL SYSTEMS (JIFFIE ADJUSTMENT)
0A37    SPEED       ;SAVE SYSTEM SPEED DURING TAPE AND SERIAL OPS
0A38    SPRITES     ;SAVE SPRITE ENABLES DURING TAPE AND SERIAL OPS
0A39    BLANKING    ;SAVE BLANKING STATUS DURING TAPE OPS
0A3A    HOLD_OFF    ;FLAG SET BY USER TO RESRV FULL CNTRL OF VIC
0A3B    LDTB1_SA    ;HI BYTE: SA OF VIC SCRN (USE W/VM1 TO MOVE SCRN)
0A3C    CLR_EA_LO   ;????? 8563 BLOCK FILL KLUDGE
0A3D    CLR_EA_HI   ;????? 8563 BLOCK FILL KLUDGE

0A40                ;$0A40-$0A7F RESERVED SWAP AREA FOR SCREEN
                    ;VARIABLES WHEN 40/80 MODE CHANGES
-------------------------------------------------------------------------
                    ;MONITOR'S DOMAIN
0A80    XCNT        ;COMPARE BUFFER (32 BYTES)
0AA0    HULP
0AAA    FORMAT
```

```
0AAB    LENGTH      ;ASM/DIS
0AAC    MSAL        ;FOR ASSEMBLER
0AAF    SXREG       ;1 BYTE TEMP USED ALL OVER
0AB0    SYREG       ;1 BYTE TEMP USED ALL OVER
0AB1    WRAP        ;1 BYTE TEMP FOR ASSEMBLER
0AB2    XSAVE       ;SAVE .X HERE DURING INDIRECT SUBRTINE CALLS
0AB3    DIRECTION   ;DIRECTION INDICATOR FOR 'TRANSFER'
0AB4    COUNT       ;PARSE NUMBER CONVERSION
0AB5    NUMBER      ;PARSE NUMBER CONVERSION
0AB6    SHIFT       ;PARSE NUMBER CONVERSION
0AB7    TEMPS
```

```
                    ;FUNCTION KEY ROM CARD TABLES

0AC0    CURBNK      ;CURRENT FUNCTION KEY ROM BANK BEING POLLED
0AC1    PAT         ;PHYSICAL ADDRESS TABLE(IDS OF LOGGED-IN CARDS)
0AC5    DK_FLAG     ;RESERVED FOR FOREIGN SCREEN EDITORS

0AC6                ;$0AC6-$0AFF RESERVED FOR SYSTEM

0B00    TBUFFR      ;CASSETTE BUFFER (192 BYTES)
                    ;$0B00-$0BC0, THIS PAGE ALSO USED
                    ;AS A BUFFER FOR THE DISK AUTO-BOOT

0C00    RS232I      ;RS-232 INPUT BUFFER

0D00    RS232O      ;RS-232 OUTPUT BUFFER

0E00                ;SPRITE DEFINITION AREA (MUST BE BELOW $1000)
                    ;$0E00-$0FFF, 512 BYTES

1000    PKYBUF      ;PROGRAMMABLE FUNCTION KEY LENGTHS TABLE
                    ;FOR 10 KEYS (F1-F8, <SHFT-RUN>, HELP)
100A    PKYDEF      ;PROGRAMMABLE FUNCTION KEY STRINGS
```

```
                    ;DOS/VSP AREA

1100    DOSSTR      ;DOS OUTPUT STR. BUF
                    ;48 BYTES TO BUILD DOS STRING
1131    VWORK       ;GRAPHICS VARS
1131    XYPOS
1131    XPOS        ;CURRENT X POSITION
1133    YPOS        ;CURRENT Y POSITION
1135    XDEST       ;X-COORDINATE DESTINATION
1137    YDEST       ;Y-COORDINATE DESTINATION

1139    XYABS       ;LINE DRAWING VARIABLES
1139    XABS
113B    YABS
113D    XYSGN
113D    XSGN
113F    YSGN
1141    FCT
1145    ERRVAL
1147    LESSER
1148    GREATR
```

```
                        ;ANGLE ROUTINE VARIABLES
1149    ANGSGN          ;SIGN OF ANGLE
114A    SINVAL          ;SINE OF VALUE OF ANGLE
114C    COSVAL          ;COSINE OF VALUE OF ANGLE
114E    ANGCNT          ;TEMPS FOR ANGLE DISTANCE ROUTINES
-------------------------------------------------------------
                        ;BASIC GRAPHIC VARIABLES.  THE FOLLOWING
                        ;24 BYTES ARE MULTIPLY DEFINED.

                        ;CIRCLE DRAWING VARIABLES
1150    XCIRCL          ;CIRCLE CENTER, X COORDINATE
1152    YCIRCL          ;CIRCLE CENTER, Y COORDINATE
1154    XRADUS          ;X RADIUS
1156    YRADUS          ;Y RADIUS
1158    ROTANG          ;ROTATION ANGLE
115C    ANGBEG          ;ARC ANGLE START
115E    ANGEND          ;ARC ANGLE END
1160    XRCOS           ;X RADIUS * COS(ROTATION ANGLE)
1162    YRSIN           ;Y RADIUS * SIN(ROTATION ANGLE)
1164    XRSIN           ;X RADIUS * SIN(ROTATION ANGLE)
1166    YRCOS           ;Y RADIUS * COS(ROTATION ANGLE)
-------------------------------------------------------------
                        ;BASIC GENERAL USE PARAMETERS
1150    XCENTR
1152    YCENTR
1154    XDIST1
1156    YDIST1
1158    XDIST2
115A    YDIST2
115C    DISEEND         ;PLACEHOLDER
115E    COLCNT          ;CHAR'S COL. COUNTER
115F    ROWCNT
1160    STRCNT
-------------------------------------------------------------
                        ;BOX-DRAWING VARIABLES
1150    XCORD1          ;POINT 1 X-COORD.
1152    YCORD1          ;POINT 1 Y-COORD.
1154    BOXANG          ;ROTATION ANGLE
1156    XCOUNT
1158    YCOUNT
115A    BXLENG          ;LENGTH OF A SIDE
115C    XCORD2
115E    YCORD2
-------------------------------------------------------------
                        ;SHAPE AND MOVE-SHAPE VARIABLES
1151    KEYLEN
1152    KEYNXT
1153    STRSZ           ;STRING LEN
1154    GETTYP          ;REPLACE SHAPE MODE
1155    STRPTR          ;STRING POS'N COUNTER
1156    OLDBYT          ;OLD BIT MAP BYTE
1157    NEWBYT          ;NEW STRING OR BIT MAP BYTE
1158                    ;PLACEHOLDER
1159    XSIZE           ;SHAPE COLUMN LENGTH
115B    YSIZE           ;SHAPE ROW LENGTH
115D    XSAVE           ;TEMP FOR COLUMN LENGTH
115F    STRADR          ;SAVE SHAPE STRING DESCRIPTOR
1161    BITIDX          ;BIT INDEX INTO BYTE
```

```
                        ;BASIC GRAPHIC VARIABLES

1168    CHRPAG          ;HIGH BYTE: ADDR OF CHARROM FOR 'CHAR' CMD.
1169    BITCNT          ;TEMP FOR GSHAPE
116A    SCALEM          ;SCALE MODE FLAG
116B    WIDTH           ;DOUBLE WIDTH FLAG
116C    FILFLG          ;BOX FILL FLAG
116D    BITMSK          ;TEMP FOR BIT MASK
116E    NUMCNT
116F    TRCFLG          ;FLAGS TRACE MODE
1170    RENUM_TMP_1     ;A TEMP FOR RENUMBER
1172    RENUM_TMP_2     ;A TEMP FOR RENUMBER
1174    T3
1175    T4
1177    VTEMP3          ;GRAPHIC TEMP STORAGE
1178    VTEMP4
1179    VTEMP5
117A    ADRAY1          ;PTR TO ROUTINE:CONVERT FLOAT -> INTEGER
117C    ADRAY2          ;PTR TO ROUTINE:CONVERT INTEGER -> FLOAT
117E    SPRITE_DATA     ;SPRITE SPEED/DIRECTION TABLES ($117E-D5)
11D6    VIC_SAVE        ;COPY OF VIC REG'S, USED TO UPDATE CHIP
                        ;DURING RETRACE (21 BYTES, $11D6-EA)
11EB    UPPER_LOWER     ;POINTER TO UPPER/LOWER CHAR SET FOR CHAR
11EC    UPPER_GRAPHIC               UPPER/GRAPHIC
11ED    DOSSA           ;TEMP STORAGE FOR FILE SA DURING RECORD CMD

                        ;BASIC GENERAL NON-ZP STORAGE
1200    OLDLIN          ;PREVIOUS BASIC LINE NUMBER
1202    OLDTXT          ;POINTER: BASIC STATEMENT FOR CONTINUE

                        ;PRINT USING DECLARATIONS
1204    PUCHRS
1204    PUFILL          ;PRINT USING FILL SYMBOL
1205    PUCOMA          ;PRINT USING COMMA SYMBOL
1206    PUDOT           ;PRINT USING D.P. SYMBOL
1207    PUMONY          ;PRINT USING MONETARY SYMBOL

1208    ERRNUM          ;USED BY ERROR TRAPPING ROUTINE-LAST ERR NO
1209    ERRLIN          ;LINE # OF LAST ERROR - FFFF IF NO ERROR
120B    TRAPNO          ;LINE TO GO TO ON ERROR.. FFXX IF NONE SET
120D    TMPTRP          ;HOLD TRAP # TEMPOR.
120E    ERRTXT
1210    TEXT_TOP        ;TOP OF TEXT POINTER
1212    MAX_MEM_0       ;HIGHEST ADDRESS AVAILABLE TO BASIC IN RAM 0
1214    TMPTXT          ;USED BY DO-LOOP. COULD BE MULT. ASSIGNED
1216    TMPLIN
1218    USRPOK
121B    RNDX
1220    CIRCLE_SEGMENT ;DEGREES PER CIRCLE SEGMENT
1221    DEJAVU          ;'COLD' OR 'WARM' RESET STATUS
                        ;(MUST BE IN PAGE 5!)

                        ;BASIC STORAGE FOR MUSIC VECTORS

1222    TEMPO_RATE
1223    VOICES
1229    NTIME
```

128 USERS ONLY

```
122B    OCTAVE
122C    SHARP
122D    PITCH
122F    VOICE
1230    WAVE0
1233    DNOTE
1234    FLTSAV
1238    FLTFLG
1239    NIBBLE
123A    TONNUM
123B    TONVAL
123E    PARCNT
123F    ATKTAB
1249    SUSTAB
1253    WAVTAB
125D    PULSLW
1267    PULSHI
1271    FILTERS
```

```
        ;INTERRUPT VECTORS

1276    INT_TRIP_FLAG
1279    INT_ADR_LO
127C    INT_ADR_HI
127F    INTVAL
1280    COLTYP
```

```
        ;BASIC SOUND COMMAND VARS

1281    SOUND_VOICE
1282    SOUND_TIME_LO
1285    SOUND_TIME_HI
1288    SOUND_MAX_LO
128B    SOUND_MAX_HI
128E    SOUND_MIN_LO
1291    SOUND_MIN_HI
1294    SOUND_DIRECTION
1297    SOUND_STEP_LO
129A    SOUND_STEP_HI
129D    SOUND_FREQ_LO
12A0    SOUND_FREQ_HI
```

```
        ;BASIC SOUND COMMAND VARS

12A3    TEMP_TIME_LO
12A4    TEMP_TIME_HI
12A5    TEMP_MAX_LO
12A6    TEMP_MAX_HI
12A7    TEMP_MIN_LO
12A8    TEMP_MIN_HI
12A9    TEMP_DIRECTION
12AA    TEMP_STEP_LO
12AB    TEMP_STEP_HI
12AC    TEMP_FREQ_LO
12AD    TEMP_FREQ_HI
12AE    TEMP_PULSE_LO
12AF    TEMP_PULSE_HI
```

```
12B0    TEMP_WAVEFORM
12B1    POT_TEMP_1      ;TEMPORARYS FOR 'POT' FUNCTION
12B2    POT_TEMP_2
12B3    WINDOW_TEMP
12B7    SAVRAM          ;USED BY SPRDEF & SAVSPR
12FA    DEFMOD          ;USED BY SPRDEF & SAVSPR
12FB    LINCNT          ;USED BY SPRDEF & SAVSPR
12FC    SPRITE_NUMBER   ;USED BY SPRDEF & SAVSPR
12FD    IRQ_WRAP_FLAG   ;USED BY BASIC IRQ TO BLOCK
                        ;ALL BUT ONE IRQ CALL

1300                ;UNALLOCATED ABSOLUTE RAM-RESERVED FOR
                    ;APPLICATIONS SOFTWARE $1300-$1BFF

1C00    RAMBOT      ;START OF BASIC TEXT (KERNAL SETS MEMBOT HERE)
                    ;OR
1C00                ;VIDEO MATRIX #2 (1KB OF COLORS FOR BITMAP,
                    ;IF ALLOCATED)

2000                ;VIC BITMAP (8KB, IF ALLOCATED)

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

                    ;** BEGINNING OF ROM OVER RAM **
4000                ;C128 BASIC LO ROM

8000                ;C128 BASIC HI ROM (OR LANGUAGE CARD OR BASIC 2.2)
```

Continued from pg. 79

positioning, disabling and enabling the stop key through directives, toggling warning messages on or off, sharing and/or chaining variables, and other useful operations.

The documentation is complete and detailed, although short on programming examples. This results in a certain amount of ambiguity, especially if you're not familiar with advanced programming principles. Since the content of the manual is almost word-for-word in either version, it is a minor setback for both *InstaSpeed* and *SpeedWriter*.

The *DTL 64 Compiler* is relatively easy to use, but programming novices may have a hard time using the advanced functions effectively because the documentation lacks practical examples. The program in either marketed form is menu-driven and supplied on disk. Overall, this compiler is not as easy to use or as full-featured as the *BASIC-64* compiler reviewed

InstaSpeed

Microsci Marketing
2158 Hathaway Street
Santa Ana, CA 92705
(714) 241-5600
$59.95 suggested retail price

InstaSpeed is essentially the device-protected version of the *DTL-64* compiler described above with a few differences. The most noticeable difference is that a "dongle" must be inserted into control port two on the Commodore 64 or the program will not load. You may, however, make backup copies of the disk if you so desire. Be advised that these copies also require using the dongle to operate successfully.

Compiled programs will run without using the dongle, however, but the run-time library (RTL) must also be present on the program disk, since this will have to be loaded to run the compiled program.

SpeedWriter

CodeWriter Corporation
7847 North Caldwell Avenue
Niles, IL 60648
800-621-4109
$50.00 suggested retail price

SpeedWriter does not require the use of a dongle for compilation, and it is supplied on disk.

Disk drive configuration options are provided with this DTL version, allowing the user to select configurations for one or two single drives (i.e., two 1541's) or for a dual-drive system, such as the 4040. Parallel drives using an IEEE interface are also supported with this version.

One annoying feature of *SpeedWriter* is that it automatically terminates itself after compilation. Since the program takes more than two minutes to load, this gets to be a major pain if you wish to compile several individual programs all in the same session. A control file can be used to compile several "layered" programs, but this doesn't help much for compiling stand-alone programs.

BLITZ! Compiler

Skyles Electric Works
231E South Whisman Road
Mountain View, CA 94041
(415) 965-1735
$99.00 suggested retail price

Blitz! is a fully-transportable BASIC compiler that utilizes P-code to increase program execution speed.

Blitz! is menu-driven and very easy to use. A nice feature is that it may be configured for a single drive, a dual drive, or two single drives with difference device numbers, for optimal compiling efficiency.

The compilation process requires two passes, as with the other compilers covered here. The first pass translates the BASIC program into P-code and checks it for syntax and type mismatch errors. Any line numbers in the BASIC program containing such errors will be displayed as they are discovered. *Blitz!* will also allow the use of certain BASIC extensions such as *VICtree* with minimal fuss.

The second compilation pass replaces all variables and line references with exact memory locations. Any lines previously referenced that are still unknown will result in an undefined statement error message along with the line number it occurs in.

If no errors were encountered during the first or second pass, a compiled program is now created and written to disk. The compiled program consists of the appended run-time routines, data statements, P-code and the variables table. Since all of these components are consolidated into the compiled program, the compiled version is fully transportable without having to put additional support files on the same disk.

The compiled program will bear the original program name, but with a "c/" prefix. For example, a BASIC program called "counter" would become "c/counter" in the compiled version. The original BASIC version of the program remains intact on the disk, and can be deleted after compilation.

During compiling, another file is created and written to disk also. This file has a "z/" prefix (e.g., "z/counter"), and it provides a cross-reference to the original BASIC line numbers and their corresponding locations in the *Blitz!* program counter. This file may also be deleted after compiling the BASIC program satisfactorily.

Blitz! also allows the stop key to be enabled or disabled from within the source program, and lets you link programs and share variables with an overlay utility.

The 23-page *Blitz!* user manual is complete and very easy to understand. It contains several helpful tips for writing more "compiler-friendly" source programs and for merging machine language and BASIC programs together. Overall, *Blitz!* is a very good, easy-to-use (albeit expensive) BASIC compiler.

The Final Pass

All compilers require that the source code program be written correctly, since any errors in the BASIC code are likely to cause the compiler to choke up and stop. Knowing what the compiler expects in the form of numeric and string data will allow you to create your source program to conform to these expectations, resulting in a more efficient compiled program. Whether you use a compiler or not, good programming practices will help all your programs run faster and more efficiently.

Now that you know the facts about BASIC compilers, you will have to decide whether or not they will help you in your work. You'll have to evaluate the advantages and disadvantages of compilers and base your decision on these factors. But at least now you know what they are, what they do and what they won't do, so you can make an educated decision. C

HOW TO ENTER PROGRAMS

The programs which appear in this magazine have been run, tested and checked for bugs and errors. After a program is tested, it is printed on a letter quality printer with some formatting changes. This listing is then photographed directly and printed in the magazine. Using this method ensures the most error-free program listings possible.

Whenever you see a word inside brackets, such as [DOWN], the word represents a keystroke or series of keystrokes on the keyboard. The word [DOWN] would be entered by pressing the cursor-down key. If multiple keystrokes are required, the number will directly follow the word. For example, [DOWN4] would mean to press the cursor-down key four times. If there are multiple words within one set of brackets, enter the keystrokes directly after one another. For example, [DOWN, RIGHT 2] would mean to press the cursor-down key once and then the cursor-right key twice.

In addition to these graphic symbols, the keyboard graphics are all represented by a word and a letter. The word is either SHFT or CMD and represents the SHIFT key or the Commodore key. The letter is one of the letters on the keyboard. The combination [SHFT E] would be entered by holding down the SHIFT key and pressing the E. A number following the letter tells you how many times to type the letter. For example, [SHFT A4,CMD B3] would mean to hold the SHIFT key and press the A four times, then hold down the Commodore key and press the B three times.

The chart on this page tells you the keys to press for any word or words inside brackets. Refer to this chart whenever you aren't sure what keys to press. The little graphic next to each keystroke shows you what you will see on the screen.

SYNTAX ERROR

This is by far the most common error encountered while entering a program. Usually (sorry folks) this means that you have typed something incorrectly on the line the syntax error refers to. If you get the message "?Syntax Error Break In Line 270", type LIST 270 and press RETURN. This will list line 270 to the screen. Look for any non-obvious mistakes like a zero in place of an O or vice-versa. Check for semicolons and colons reversed and extra or missing parenthesis. All of these things will cause a syntax error.

There is only one time a syntax error will tell you the 'wrong' line to look at. If the line the syntax error refers to has a function call (i.e., FN A(3)), the syntax error may be in the line that defines the function, rather than the line named in the error message. Look for a line near the beginning of the program (usually) that has DEF FN A(X) in it with an equation following it. Look for a typo in the equation part of this definition.

ILLEGAL QUANTITY ERROR

This is another common error message. This can also be caused by a typing error, but it is a little harder to find. Once again, list the line number that the error message refers to. There is probably a poke statement on this line. If there is, then the error is referring to what is trying to be poked. A number must be in the range of zero to 255 to be poke-able. For example, the statement POKE 1024,260 would produce an illegal quantity error because 260 is greater than 255.

Most often, the value being poked is a variable (A,X...). This error is telling you that this variable is out of range. If the variable is being read from data statements, then the problem is somewhere in the data statements. Check the data statements for missing commas or other typos.

If the variable is not coming from data statements, then the problem will be a little harder to find. Check each line that contains the variable for typing mistakes.

OUT OF DATA ERROR

This error message is always related to the data statements in a program. If this error occurs, it means that the program has run out of data items before it was supposed to. It is usually caused by a problem or typo in the data statements. Check first to see if you have left out a whole line of data. Next, check for missing commas between numbers. Reading data from a page of a magazine can be a strain on the brain, so use a ruler or a piece of paper or anything else to help you keep track of where you are as you enter the data.

OTHER PROBLEMS

It is important to remember that the 64 and the PET/CBM computers will only accept a line up to 80 characters long. The VIC 20 will accept a line up to 88 characters long. Sometimes you will find a line in a program that runs over this number of characters. This is not a mistake in the listing. Sometimes programmers get so carried away crunching programs that they use abbreviated commands to get more than 80 (or 88)

CHART OF SPECIAL CHARACTER COMMANDS

"[HOME]" = UNSHIFTED CLR/ HOME
"[CLEAR]" = SHIFTED CLR/HOME
"[DOWN]" = CURSOR DOWN
"[UP]" = CURSOR UP
"[RIGHT]" = CURSOR RIGHT
"[LEFT]" = CURSOR LEFT
"[RVS]" = CONTROL 9
"[RVOFF]" = CONTROL 0
"[BLACK]" = CONTROL 1
"[WHITE]" = CONTROL 2
"[RED]" = CONTROL 3
"[CYAN]" = CONTROL 4
"[PURPLE]" = CONTROL 5
"[GREEN]" = CONTROL 6
"[BLUE]" = CONTROL 7
"[YELLOW]" = CONTROL 8
"[ORANGE]" = COMMODORE 1
"[BROWN]" = COMMODORE 2
"[L. RED]" = COMMODORE 3
"[GRAY1]" = COMMODORE 4
"[GRAY2]" = COMMODORE 5
"[L. GREEN]" = COMMODORE 6
"[L. BLUE]" = COMMODORE 7
"[GRAY3]" = COMMODORE 8
"[F1]" = F1
"[F2]" = F2
"[F3]" = F3
"[F4]" = F4
"[F5]" = F5
"[F6]" = F6
"[F7]" = F7
"[F8]" = F8
"[POUND]" = ENGLISH POUND
"[SHFT ↑]" = PI SYMBOL
"[↑]" = UP ARROW

GRAPHIC SYMBOLS WILL BE REPRESENTED AS EITHER THE LETTERS SHFT (SHIFT) AND A KEY ("[SHFT Q,SHFT J,SHFT D,SHFT S]") OR THE LETTERS CMDR (COMMODORE) AND A KEY ("[CMDR Q,CMDR G,COMDR Y,CMDR H]"). IF A SYMBOL IS REPEATED, THE NUMBER OF REPITITIONS WILL BE DIRECTLY AFTER THE KEY AND BEFORE THE COMMA ("[SPACE3,SHFT S4,CMDR M2]").

characters on one line. You can enter these lines by abbreviating the commands when you enter the line. The abbreviations for BASIC commands are on pages 133-134 of the VIC 20 user guide and 130-131 of the Commodore 64 user's guide.

If you type a line that is longer than 80 (or 88) characters, the computer will act as if everything is ok, until you press RETURN. Then, a syntax error will be displayed.

THE PROGRAM WON'T RUN!!

This is the hardest of problems to resolve; no error message is displayed, but the program just doesn't run. This can be caused by many small mistakes typing a program in. First check that the program was written for the computer you are using. Check to see if you have left out any lines of the program. Check each line of the program for typos or missing parts. Finally, press the RUN/STOP key while the program is 'running'. Write down the line the program broke at and try to follow the program backwards from this point, looking for problems.

IF ALL ELSE FAILS

You've come to the end of your rope. You can't get the program to run and you can't find any errors in your typing. What do you do? As always, we suggest that you try a local user group for help. In a group of even just a dozen members, someone is bound to have typed in the same program.

If you do get a working copy, be sure to compare it to your own version so that you can learn from your errors and increase you understanding of programming.

If you live in the country, don't have a local user group, or you simply can't get any help, write to us. If you do write to us, include the following information about the program you are having problems with:

- The name of the program
- The issue of the magazine it was in
- The computer you are using
- Any error messages and the line numbers
- Anything displayed on the screen
- A printout of your listing (if possible)

Send your questions to:

Commodore Microcomputers
1200 Wilson Drive
West Chester, PA 19380
ATTN: Program Problem **C**

How to Use the Magazine Entry Program

The Magazine Entry Program on page 125 is a machine language program that will assist you in entering the programs in this magazine correctly. It is for use with the Commodore 64 only and was written by Mark Robin using the IEA Editor/Assembler. Once the program is in place, it works its magic without you having to do anything else. The program will not let you enter a line if there is a typing mistake on it, and better yet, it identifies the kind of error for you.

Getting Started

Type in the Magazine Entry Program carefully and save it as you go along (just in case). Once the whole program is typed in, save it again on tape or disk. Now RUN the program. The word POKING will appear on the top of the screen with a number. The number will increment from 49152 up to 50052, and just lets you know that the program is running. If everything is ok, the program will finish running **and end. Then** type NEW. If there is a problem with the data statements, the program will tell you where to look to find the problem.

Once the program has run, it is in memory ready to go. To activate the program, type SYS49152 and press RETURN. When the READY prompt is displayed, type TEST and press RETURN. You are now ready to enter the programs from the magazine.

Typing the Programs

All the program listings in this magazine that are for the 64 have an apostrophe followed by four letters at the end of the line (i.e., 'ACDF). The apostrophe and letters *should* be entered along with the rest of the line. This is a checksum that the Magazine Entry Program uses.

Enter the line and the letters at the end and then press RETURN, just as you normally would.

If the line is entered correctly, a bell is sounded and the line is entered into the computer's memory (without the characters at the end).

If a mistake was made while entering the line, a noise is sounded and an error message is displayed. Read the error message, then press any key to erase the message and correct the line.

IMPORTANT

If the Magazine Entry Program sees a mistake on a line, it *does not* enter that line into memory. This makes it impossible to enter a line incorrectly.

Error Messages and What They Mean

There are six error messages that the Magazine Entry Program uses. Here they are, along with what they mean and how to fix them.

NO CHECKSUM: This means that you forgot to enter the apostrophe and the four letters at the end of the line. Move the cursor to the end of the line you just typed and enter the checksum.

QUOTE: This means that you forgot (or added) a quote mark somewhere in the line. Check the line in the magazine and correct the quote.

PARENTHESIS: This means that you forgot (or added) a parenthesis somewhere in the line. Check the line in the magazine again and correct the parenthesis.

KEYWORD: This means that you have either forgotten a command or spelled one of the BASIC keywords (GOTO, PRINT..) incorrectly. Check the line in the magazine again and check your spelling.

 # OF CHARACTERS: This means that you have either entered extra characters or missed some characters. Check the line in the magazine again. This error message will also occur if you misspell a BASIC command, but create another keyword in doing so. For example, if you misspell PRINT as PRONT, the 64 sees the letter P and R, the BASIC keyword ON and then the letter T. Because it sees the keyword ON, it thinks you've got too many characters, instead of a simple misspelling. Check spelling of BASIC commands if you can't find anything else wrong.

UNIDENTIFIED: This means that you have either made a simple spelling error, you typed the wrong line number, or you typed the checksum incorrectly. Spelling errors could be the wrong number of spaces inside quotes, a variable spelled wrong, or a word mispelled. Check the line in the magazine again and correct the mistake. **C**

Magazine Entry Program

The Magazine Entry Program is available on disk, along with the other programs in this magazine, for $9.95. To order, contact Loadstar at 1-800-831-2694.

```
1 PRINT "[CLEAR]POKING-";
5 P=49152 :REM $C000
10 READ A$:IF A$="END"THEN 80
20 L=ASC(MID$(A$,2,1))
30 H=ASC(MID$(A$,1,1))
40 L=L-48:IF L>9 THEN L=L-7
50 H=H-48:IF H>9 THEN H=H-7
60 PRINT"[HOME,RIGHT12]"P;
70 B=H*16+L:POKE P,B:T=T+B:P=P+1
   :GOTO 10
80 IF T<>103233 THEN PRINT"MISTAKE IN
   DATA --> CHECK DATA STATEMENTS":END
90 PRINT"DONE":END
1000 DATA 4C,23,C0,00,00,00,00,00
1001 DATA 00,00,00,00,00,00,00,0D
1002 DATA 00,58,C1,5E,C1,66,C1,76
1003 DATA C1,83,C1,8F,C1,EA,EA,EA
1004 DATA 4C,83,C0,A2,05,BD,1D,C0
1005 DATA 95,73,CA,10,F8,60,A0,02
1006 DATA B9,00,02,D9,3C,C1,D0,0B
1007 DATA 88,10,F5,A9,01,8D,10,C0
1008 DATA 4C,1F,C1,60,A0,03,B9,00
1009 DATA 02,D9,38,C1,D0,E0,88,10
1010 DATA F5,A9,00,8D,10,C0,4C,1F
1011 DATA C1,60,A0,03,B9,00,02,D9
1012 DATA 34,C1,D0,E0,88,10,F5,A0
1013 DATA 05,B9,A2,E3,99,73,00,88
1014 DATA 10,F7,A9,00,8D,18,D4,4C
1015 DATA 1F,C1,E6,7A,D0,02,E6,7B
1016 DATA 4C,79,00,A5,9D,F0,F3,A5
1017 DATA 7A,C9,FF,D0,ED,A5,7B,C9
1018 DATA 01,D0,E7,20,5A,C0,AD,00
1019 DATA 02,20,A3,C0,90,DC,A0,00
1020 DATA 4C,EA,C1,C9,30,30,06,C9
1021 DATA 3A,10,02,38,60,18,60,C8
1022 DATA B1,7A,C9,20,D0,03,C8,D0
1023 DATA F7,B1,7A,60,18,C8,B1,7A
1024 DATA F0,35,C9,22,F0,F5,6D,05
1025 DATA C0,8D,05,C0,AD,06,C0,69
1026 DATA 00,8D,06,C0,4C,BD,C0,18
1027 DATA 6D,07,C0,8D,07,C0,90,03
1028 DATA EE,08,C0,EE,0B,C0,60,18
1029 DATA 6D,0A,C0,8D,0A,C0,90,03
1030 DATA EE,09,C0,EE,0C,C0,60,0A
1031 DATA A8,B9,11,C0,85,FB,B9,12
1032 DATA C0,85,FC,A0,00,A9,12,20
1033 DATA D2,FF,B1,FB,F0,06,20,D2
1034 DATA FF,C8,D0,F6,20,54,C3,20
1035 DATA 7E,C3,20,E4,FF,F0,FB,A0
1036 DATA 1B,B9,3F,C1,20,D2,FF,88
1037 DATA 10,F7,68,68,A9,00,8D,00
1038 DATA 02,4C,74,A4,4B,49,4C,4C
1039 DATA 54,45,53,54,41,44,44,91
1040 DATA 91,0D,20,20,20,20,20,20
1041 DATA 20,20,20,20,20,20,20,20
1042 DATA 20,20,20,20,20,20,91,0D
1043 DATA 51,55,4F,54,45,00,4B,45
1044 DATA 59,57,4F,52,44,00,23,20
1045 DATA 4F,46,20,43,48,41,52,41
1046 DATA 43,54,45,52,53,00,55,4E
1047 DATA 49,44,45,4E,54,49,46,49
1048 DATA 45,44,00,4E,4F,20,43,48
1049 DATA 45,43,4B,53,55,4D,00,50
1050 DATA 41,52,45,4E,54,48,45,53
1051 DATA 49,53,00,C8,B1,7A,D0,FB
1052 DATA 84,FD,C0,09,10,03,4C,C7
1053 DATA C1,88,88,88,88,88,B1,7A
1054 DATA C9,27,D0,13,A9,00,91,7A
1055 DATA C8,A2,00,B1,7A,9D,3C,03
1056 DATA C8,E8,E0,04,D0,F5,60,4C
1057 DATA F2,C2,A0,00,B9,00,02,99
1058 DATA 40,03,F0,F2,C8,D0,F5,A0
1059 DATA 00,B9,40,03,F0,E8,99,00
1060 DATA 02,C8,D0,F5,20,D7,C1,4C
1061 DATA 56,C2,A0,0B,A9,00,99,03
1062 DATA C0,8D,3C,03,88,10,F7,A9
1063 DATA 80,85,02,20,1B,C3,A0,00
1064 DATA 20,9B,C1,20,CA,C1,20,31
1065 DATA C2,E6,7A,E6,7B,20,7C,A5
1066 DATA A0,00,20,AF,C0,F0,CD,24
1067 DATA 02,F0,06,20,D7,C0,4C,12
1068 DATA C2,C9,22,D0,06,20,BC,C0
1069 DATA 4C,12,C2,20,E7,C0,4C,12
1070 DATA C2,A0,00,B9,00,02,20,A3
1071 DATA C0,C8,90,0A,18,6D,09,C0
1072 DATA 8D,09,C0,4C,33,C2,88,A2
1073 DATA 00,B9,00,02,9D,00,02,F0
1074 DATA 04,E8,C8,D0,F4,60,18,AD
1075 DATA 0B,C0,69,41,8D,0B,C0,38
1076 DATA AD,0C,C0,E9,19,90,06,8D
1077 DATA 0C,C0,4C,60,C2,AD,0C,C0
1078 DATA 69,41,8D,0C,C0,AD,05,C0
1079 DATA 6D,07,C0,48,AD,06,C0,6D
1080 DATA 08,C0,8D,0E,C0,68,6D,0A
1081 DATA C0,8D,0D,C0,AD,0E,C0,6D
1082 DATA 09,C0,8D,0E,C0,38,E9,19
1083 DATA 90,06,8D,0E,C0,4C,96,C2
1084 DATA AD,0E,C0,69,41,8D,0E,C0
1085 DATA AD,0D,C0,E9,19,90,06,8D
1086 DATA 0D,C0,4C,AB,C2,AD,0D,C0
1087 DATA 69,41,8D,0D,C0,A0,01,AD
1088 DATA 0B,C0,CD,3C,03,D0,20,C8
1089 DATA AD,0C,C0,CD,3D,03,D0,17
1090 DATA C8,AD,0D,C0,CD,3E,03,D0
1091 DATA 0E,AD,0E,C0,CD,3F,03,D0
1092 DATA 06,20,64,C3,4C,7A,C0,AD
1093 DATA 10,C0,D0,11,98,48,68,4C
1094 DATA F7,C0,AD,10,C0,F0,01,60
1095 DATA A9,04,4C,F7,C0,A4,FD,A9
1096 DATA 27,91,7A,A2,00,C8,BD,0B
1097 DATA C0,91,7A,C8,E8,E0,04,D0
1098 DATA F5,A9,00,91,7A,20,64,C3
1099 DATA 4C,7A,C0,A0,00,B9,00,02
1100 DATA F0,11,C9,28,D0,03,EE,03
1101 DATA C0,C9,29,D0,03,EE,04,C0
1102 DATA C8,D0,EA,AD,03,C0,CD,04
1103 DATA C0,D0,01,60,A9,05,4C,F7
1104 DATA C0,A9,20,8D,00,D4,8D,01
1105 DATA D4,A9,09,8D,05,D4,A9,0F
1106 DATA 8D,18,D4,60,20,41,C3,A9
1107 DATA 81,20,77,C3,A9,80,20,77
1108 DATA C3,4C,71,C3,20,41,C3,A9
1109 DATA 11,20,77,C3,A9,10,20,77
1110 DATA C3,A9,00,8D,04,D4,60,8D
1111 DATA 04,D4,A2,70,A0,00,88,D0
1112 DATA FD,CA,D0,FA,60,END
```

SPACE STATION

Continued from pg. 18

sure you've received approval for your overall plan, and the computer will go over each item line-by-line, telling you what's approved and where you went wrong—if anywhere. It also lets you know if you've forgotten any essential equipment. I especially liked this aspect—because there was just enough "hand-holding" so I wouldn't commit myself to a totally disastrous plan.

After you've received approval for your plan, you can schedule the actual launch. Launch time is very exciting. You receive a launch date and time as well as weather conditions and wind speed. And, as in reality, you may find yourself on "hold" only five seconds before launch because of a breakdown.

Watch your lower screen—when it's flashing, you have a message and you must press the F7 key to read it. Maybe the problem will be resolved, but then, maybe it won't. You may end up being delayed.

When you actually do launch, it's exhilarating. You see the shuttle turning and the booster rockets separating. The graphics on this simulation are great, as well as the sound effects! I live near Kennedy Space Center and have seen several launches. I was impressed with this simulation.

Your work isn't over after launching—there's plenty more to do. Steer your shuttle on the appropriate trajectory marked for you, and move out into space to begin research.

Eventually it's time to come home, so enter "land" mode and steer your craft home to safety. Once home, you receive a "damage control" report.

Project: Space Station is an exceptionally good program for children ages 12 and up, as well as for adult space freaks. The program offers a tremendous variety of options for many hours of fun and learning. The program is also superior for business and economics training.

You can save a mission and come back to play it later this afternoon or next week, so time is never a problem. I see this program as especially well suited for home users, and highly recommend *Project: Space Station* to parents. Who knows, your son or daughter—or even you—could be inspired to be an astronaut of the future! C

Continued from pg. 83

neck, shoulders, arms, and legs. A purple pendulum starts swinging back and forth, and then flashes like a strobe light. If all goes well, you'll be in a trance and ready for suggestion.

Fortunately, you programmed in the suggestions beforehand. Let's say you want to lose weight. The program will strobe words like "food," "eat," and "lunch" across the screen, paired with negative words like "puke," "scab," "urine" and other equally tasteless suggestions. The idea is that if your subconscious mind learns to associate eating with something negative, eating won't seem so attractive anymore and you'll lose weight. That's just a theory, mind you.

I know what you're thinking. Do you stay in a trance until smoke starts pouring out of your disk drive? No. After a while Kurian gives off a screaming siren and tells you to "WAKE UP!" If you wake up with an intense craving for fudge brownies, something must have gone wrong somewhere. Like professional hypnosis, *The Hypnotist* doesn't work on everybody. The same can be said for all kinds of therapy.

The End

Dan: I have another problem.
Eliza: Do you say you have another problem for a special reason?
Dan: I don't have an ending for the article.
Eliza: Why don't you have an ending for the article?
Dan: I can't think of one.
Eliza: How do you know you can't think of one?
Dan: I tried.
Eliza: That's quite interesting.
Dan: Don't patronize me.
Eliza: What does that suggest to you?
Dan: That you can't finish the article, either.
Eliza: We were discussing you—not me.
Dan: Why don't you want to talk about yourself?
Eliza: Do you believe I don't want to talk about myself?
Dan: You're avoiding my questions.
Eliza: That's interesting.
Dan: Maybe you have a few problems of your own you'd like to discuss?
Eliza: Oh, I have a problem?
Dan: Are you asking me or telling me?
Eliza: Why are you interested in whether I am asking you or telling you?
Dan: You need a shrink yourself!
Eliza: You're not really talking about me, are you?
Dan: I don't see anyone else here. How long have you been avoiding people's questions?
Eliza: Why do you ask?
Dan: Forget it. I'm finished with the article anyway.
Eliza: Do you believe it's normal to be finished?
Dan: Shut up. You've been no help at all.
Eliza: You seem to have avoided speaking of your parents.

Dan Gutman received a degree in psychology from Rutgers University. Help him cut down on his psychiatric bills by asking your local newspaper to run his weekly column, "Computers & Technology," distributed by The McNaught Syndicate.

Directory of Self-Enrichment Manufacturers

Artificial Intelligence Research Group
921 North La Jolla Avenue
Los Angeles, CA 90046
Makes *Eliza,* the computer psychiatrist

Avant-Garde
37B Commercial Boulevard
Novato, CA 94947
Makes *The Einstein Memory Trainer,* a program that attempts to improve your memory.

Bantam Software
666 Fifth Avenue
New York, NY 10103
Makes *Know Your Own I.Q./Know Your Own Personality,* psychological evaluation tests.

DesignWare
185 Berry Street
San Francisco, CA 94107
Makes *Remember!,* a program that helps you memorize written material.

Franklin Software
P.O. Box 337
Blue Bell, PA 19422
Makes *The Handwriting Analyst*

Human Edge Software
2445 Faber Place
Palo Alto, CA 94303
Makes *Mind Prober,* which helps make evaluations of other people's personalities.

International Self-Help Institute
P.O. Box 520950
Salt Lake City, UT 84152
Makes *The Self Analysis Program,* which evaluates your maturity level.

Mindscape
3444 Dundee Road
Northbrook, IL 60062
Makes *The Luscher Profile,* which creates personality evaluations on the basis of how you respond to various colors. Not currently available for 64.

Personal Growth Technologies
Box 1884
Boston, MA 02105
Makes *The Software Listener,* which is similar to *Eliza,* but also attempts to lead you toward more positive thinking.

Psycom Software International
2118 Forest Lake Drive
Cincinnati, OH 45244
Makes *The Hypnotist,* a hardware/software program that attempts to use hypnosis theraputically. Also makes *Personality Analyzer,* an evaluation program.

Stimutech
3711 Plaza Drive #4
Ann Arbor, MI 48104
Makes *Expando-Vision,* which inserts subliminal messages into your TV viewing.

Sunsoft
Box 99
Alturas, FL 33820
Makes *IQ 64,* an intelligence test.

Synapse
17 Paul Drive
San Rafael, CA 94903
Makes *Relax,* a hardware/software program that uses biofeedback to reduce stress.

Thought Technology
2180 Belgrave Avenue
Montreal, Quebec, H4A 2L8 Canada
Makes *Calmpute,* a hardware/software program that uses biofeedback to reduce stress.

NEXT MONTH

BUT WAIT, THERE'S MORE

in upcoming issues of
Commodore Microcomputers and
Commodore Power/Play

Get more out of your Commodore computer in the months ahead with:

- Free programs to type in
- Up-to-the-minute software and hardware reviews
- Hands-on projects that put your computer through its paces
- Buyer's guides to help you make informed choices
- Late-breaking information on Commodore's Amiga computer

If you own a Commodore computer, you can't afford to miss a single issue!

Commodore Microcomputers and
Commodore Power/Play
the authorized source for Commodore users

ADVERTISERS' INDEX

*NO READER RESPONSE NUMBER

Technical Breakthrough #29

What you see is what you get!

You're playing HitchHiker's Guide to the Galaxy™, and you're presented with an option never before revealed to the human life form. You know, when you hit the key, it's going to be gone for all eternity. What do you do?

Freeze Frame!

It takes an instant "snapshot" of your screen. Dumps it to your printer, so you can file it for future reference . . . and returns you to the game, **exactly where you left off!**

Works with...	Freeze Frame	Other Screen Dumps
Disk programs	Yes	Sometimes
Cartridge programs	Yes	Seldom
All programming languages	Yes	No
Absolutely everything	Yes	No way

Freeze Frame is Cardco's greatest technical breakthrough yet. It is the first **totally transparent** screen dump utility for Commodore computers. We mean absolutely, positively, 100% TRANSPARENT . . . to **any** program in **any** language. Period. As if that weren't enough, **Freeze Frame** also gives you . . .

- **Cartridge-based program for instant-on (with female connector for chaining cartridges)**
- **Standard, Reverse, and Mirror imaging**
- **Multiple printer adaptability. Works on any printer or printer interface combination that emulates Commodore 1525 operation! High-speed options are included for Epson-compatible and Okidata-compatible printers.**

The Wizards from the Land of Oz Have Done It Again!

CARDCO, Inc. / 300 S. Topeka / Wichita, KS 67202

A Printer For All Reasons

Search For The Best High Quality Graphic Printer

If you have been looking very long, you have probably discovered that there are just too many claims and counter claims in the printer market today. There are printers that have some of the features you want but do not have others. Some features you probably don't care about, others are vitally important to you. We understand. In fact, not long ago, we were in the same position. Deluged by claims and counter claims. Overburdened by rows and rows of specifications, we decided to separate all the facts — prove or disprove all the claims to our own satisfaction. So we bought printers. We bought samples of all the major brands and tested them.

Our Objective Was Simple

We wanted to find that printer which had all the features you could want and yet be sold directly to you at the lowest price. We didn't want a "close-out special" of an obsolete product that some manufacturer was dumping, so we limited our search to only those new printers that had the latest proven technology. We wanted to give our customers the **best** printer on the market today at a bargain price.

The Results Are In

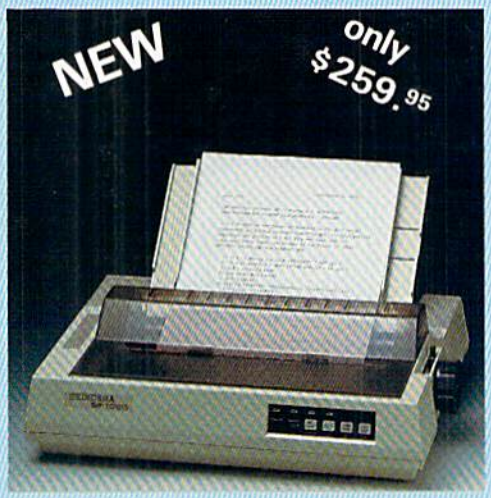
The search is over. We have reduced the field to a single printer that meets all our goals (and more). The printer is the SP-1000 from Seikosha, a division of Seiko (one of the foremost manufacturers in the world). We ran this printer through our battery of tests and it came out shining. This printer *can* do it all. Standard draft printing at a respectable 100 characters per second, and with a very readable 12 (horizontal) by 9 (vertical) character matrix. This is a full bi-directional, logic seeking, true descender printer.

"NLQ" Mode

One of our highest concerns was about print quality and readability. The SP-1000 has a print mode termed Near Letter Quality printing (NLQ mode). This is where the SP-1000 outshines all the competition. Hands down! The character matrix in NLQ mode is a very dense 24 (horizontal) by 18 (vertical). This equates to 41,472 addressable dots per square inch. Now we're talking ***quality*** printing. It looks like it was done on a typewriter. You can even print graphics using the standard graphics symbols built into your computer. The results are the best we've ever seen. The only other printers currently available having resolution this high go for $500 and more *without* the interface or cable needed to hook up to your computer.

Features That Won't Quit

With the SP-1000 your computer can now print 40, 48, 68, 80, 96, or 136 characters per line. You can print in ANY of 35 character styles including 13 double width and 3 reversed (white on black) styles. You not only have the standard Pica, Elite, Condensed and Italics, but also true Superscripts and Subscripts. Never again will you have to worry about how to print H_2O or X^2. This fantastic machine will do it automatically, through easy commands right from your keyboard. Do you sometimes want to emphasize a word? It's easy, just use **bold** (double strike) or use *italics* to make the words stand out. Or, if you wish to be even more emphatic, underline the words. You can combine many of these modes and styles to make the variation almost endless. Do you want to express something that you can't do with words? Use graphics with your text — even on the same line. You have variable line spacing of 1 line per inch to infinity (no space at all) and 143 other software selectable settings in between. You can control line spacing on a dot-by-dot basis. If you've ever had a letter or other document that was just a few lines too long to fit a page, you can see how handy this feature is. Simply reduce the line spacing slightly and ... VOILA! The letter now fits on one page.



Forms? Yes! Your Letterhead? Of Course!

Do you print forms? No problem. This unit will do them all. Any form up to 10 inches wide. The tractors are adjustable from 4 to 10 inches. Yes, you can also use single sheets. Plain typing paper, your letterhead, short memo forms, labels, anything you choose. Any size to 10" in width. In fact this unit is so advanced, it will load your paper automatically. Multiple copies? Absolutely! Use forms (up to 3 thick). Do you want to use spread sheets with many columns? Of course! Just go to condensed mode printing and print a full 136 columns wide. Forget expensive wide-carriage printers and changing to wide carriage paper. You can now do it all on a standard 8½" wide page, and you can do it quietly. The SP-1000 is rated at only 55 dB. This is quieter than any other impact dot matrix printer that we know of and is quieter than the average office background noise level.

Consistent Print Quality

Most printers have a ribbon cartridge or a single spool ribbon which gives nice dark printing when new, but quickly starts to fade. To keep the printers output looking consistently dark, the ribbons must be changed quite often. The SP-1000 solves this problem by using a wide (½") ribbon cartridge that will print thousands of pages before needing replacement. (When you finally do wear out your ribbon, replacement cost is only $11.00. Order *#2001.)*

The Best Part

When shopping for a printer with this quality and these features, you could expect to pay around $500 or more. *Not now!* We sell this fantastic printer for **only $259.95! You need absolutely nothing else to start printing — just add paper.**

No Risk Offer

We give you a 2-week satisfaction guarantee. If you are not completely satisfied for any reason we will promptly refund your purchase. A 1-year warranty is included with your printer. The warranty repair policy is to repair or replace and reship to the buyer within 72 hours of receipt.

The Bottom Line

Be sure to specify the order # for the correct version printer designed for your computer.

Commodore C-64 & C-128, Order #2200, cable included

IBM-PC and compatibles, Order #2100, plus 8' cable #1103, $26.00

Standard Parallel with 36 pin Centronics connector, Order #2400, no cable

We also have interfaces and cables for other computers not listed. Call Customer Service at 805/987-2454 for details.

Shipping and insurance is $10.00 — UPS within the continental USA. If you are in a hurry, UPS Blue (second day air), APO or FPO is $22.00. Canada, Alaska, Mexico are $30.00 (air). Other foreign is $70.00 (air). California residents add 6% tax. The above are cash prices — VISA and MC add 3% to total. We ship the next business day on money orders, cashiers' checks, and charge cards. A 14-day clearing period is required for checks.

For information call 805/987-2454

TO ORDER CALL TOLL FREE
1-(800) 962-5800 USA
1-(800) 962-3800 CALIF. (8-6 PST)

or send order to:

APROTEK

1071-A Avenida Acaso
Camarillo, CA 93010