The Digital Window: Computer Graphics Today And Tomorrow

COMPUTE!

$2.95
May
1986
Issue 72
Vol. 8, No. 5
$3.75 Canada
02193
ISSN 0194-357X

The Leading Magazine Of Home, Educational, And Recreational Computing

Two New Monthly Columns
Insight:ST And AmigaView

Managing Files
From Atari ST BASIC: A Major Book Excerpt

Better Branching In Applesoft
Computed GOTO And GOSUB

64 Autobooter
Make Disk Programs Run Automatically

Atari DEBUT:
Add Commands To BASIC For Easy Debugging

Hickory, Dickory, Dock
An Educational Game For Commodore 64 And 128, Atari, Apple, Atari ST, Amiga, And IBM PC/PCjr

MODified Shapes For IBM

Amiga Puzzle
Learn New Amiga BASIC Programming Techniques

0 71486 02193 05

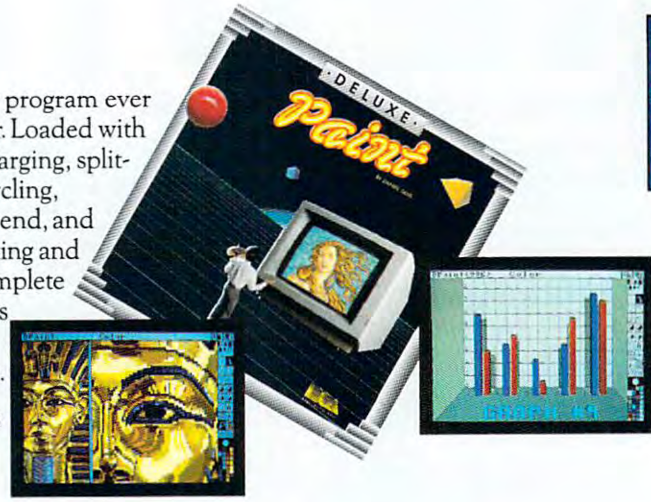
www.commodore.ca

Electronic Arts Presents 8 Good

1. DeluxePaint™

The most sophisticated paint program ever made for a personal computer. Loaded with special features like zoom enlarging, split-screen magnification, color cycling, blend, smear, shade, stretch, bend, and rotate. The custom palette mixing and unlimited brushes give you complete control over all of the Amiga's 4,096 colors. And it even works in 640 x 400 hi-res mode.

"The kinds of things painters love and miss in computer programs are all here in DeluxePaint."
—Amigaworld



4. Arcticfox™

The tank battle simulator. Feel the heat of tank combat as you command your two on-screen hands and their arsenal of heavy cannon, "seeing" missiles, and mines. Face a highly intelligent enemy in a completely three-dimensional Arctic battlefield. An accurate simulation of tank movement and strategies, with the bone-jarring sound effects and super hot graphics of the best arcade games.

5. Financial Cookbook™

Take control of your finances. Financial Cookbook answers all your personal financial questions and saves you money. From checking and savings accounts to IRAs and taxes. From variable interest rates to amortization schedules. Performs like a spreadsheet, a calculator, and an investment advisor, all rolled into one. The simple fill-in-the-blanks format makes complex financial questions—and saving money—a breeze.

7. Archon™

The action chess game with dragons, magicians, and trolls for pieces. When one piece lands on another, they have to fight a white-knuckle arcade battle for control of the square. The perfect blend of action and strategy, and a classic award winner.

"Game of the Year"
—Creative Computing

"Most Innovative Game"
—Electronic Games

Amiga is a registered trademark of Commodore Business Machines, Inc. DeluxePaint, Skyfox, Dr. J and Larry Bird Go One-on-One, Arcticfox, Financial Cookbook, Archon, Seven Cities of Gold, and Electronic Arts are registered trademarks of Electronic Arts.

www.commodore.ca

Reasons to Own an Amiga.™

2. Skyfox™

Jump into the cockpit of a fighter pilot's dream. Enemy tanks and jets splash into brilliant flames when you score a hit. You'll hear the shriek of the doppler effect as enemy jets strafe past. Skyfox is the fastest-selling game in E.A. history, and a multi-award winner.

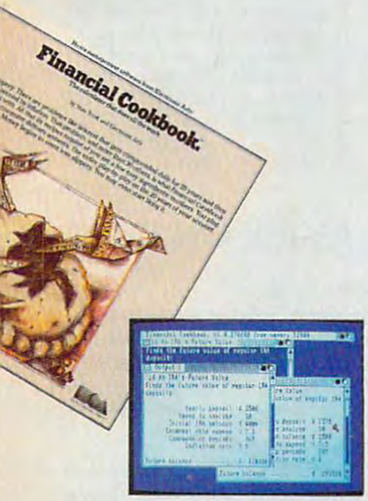
"Best Shoot-em-Up Arcade Game"
—*Family Computing*

"Best Action/Arcade Game"
—*Computer Entertainer*

3. Dr. J and Larry Bird Go One-on-One™

The number-one computer sports simulation of all time. The players look real, and the sounds are so detailed you can even hear the squeaking sneakers on the hardwood floor. Feels so real you'll think you're down on the court with these basketball superstars.

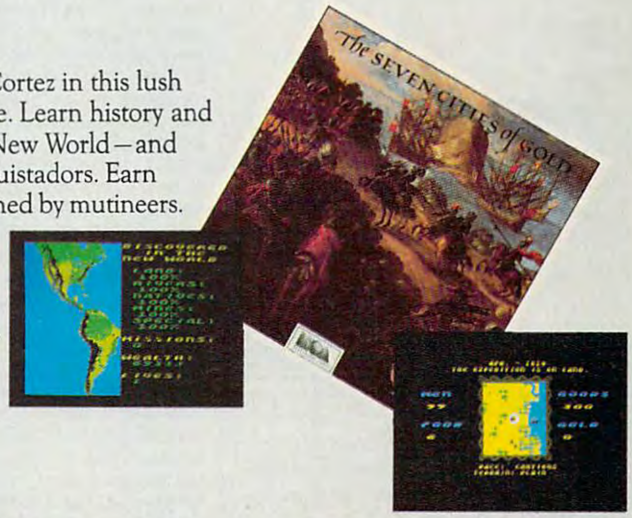
"Game of the Year"
—*Electronic Games*



6. Seven Cities of Gold™

Play the role of Columbus or Cortez in this lush simulation and adventure game. Learn history and geography as you explore the New World—and face the problems of the Conquistadors. Earn glory and gold, or wind up beached by mutineers. If you survive, the computer will generate unlimited new continents for you to explore.

"Best Role-Playing Adventure"
—*Family Computing*



8. The Eighth Reason?

These Electronic Arts products are available NOW, so you can stop waiting for the high-quality software that will let you get the most from your Amiga.

ELECTRONIC ARTS™

How to Order: Visit your retailer or call 800-245-4525 (in CA call 800-562-1112) for VISA or MasterCard orders. To purchase by mail, send check or money order to Electronic Arts, P.O. Box 7530, San Mateo, CA 94403. Add $5 for shipping and handling. Allow 2-4 weeks for delivery. All of the above Amiga products are $39.95, except Financial Cookbook at $49.95 and DeluxePaint at $99.95. To get a complete product catalog and order form, send 50¢ and a stamped, self-addressed envelope to Electronic Arts Catalog, 1820 Gateway Drive, San Mateo, CA 94404.

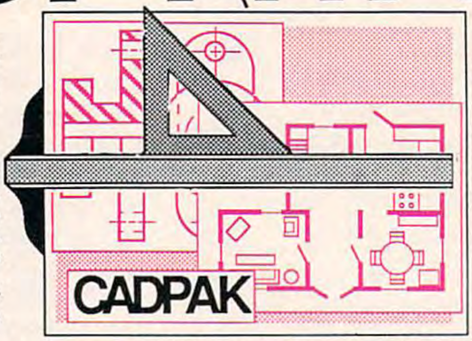
www.commodore.ca

'128™ and C-64™ SPECTACULAR SOFTWARE

Our BASIC Compilers are the *complete* compiler and development packages. Speed up your programs from 5x to 35x.

Our BASIC Compilers give you many options: flexible memory management; choice of compiling to machine code, compact p-code or a mixture of both. Also on the '128, 40 or 80 column monitor output and FAST-mode operation.

The '128 Compiler's extensive 80-page programmer's guide covers compiler directives and options, two levels of optimization, memory usage, input/output handling, 80 column hi-res graphics, faster, higher precision math functions, speed and space saving tips, more. A great package that no software library should be without.
BASIC 128 Compiler $59.95
BASIC 64 Compiler $39.95



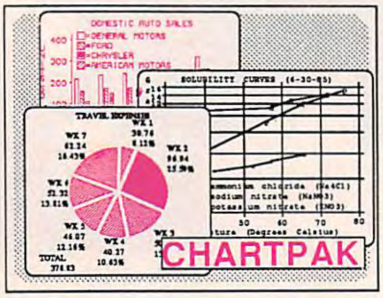
CADPAK is a remarkably easy to use drawing package for accurate graphic designs.

Using CADPAKs new dimensioning features you can create exact scaled output to all major dot-matrix printers.

This enhanced version of CADPAK allows you to input via the keyboard or a high quality lightpen. Two graphic screens permit you to COPY from one screen to another. DRAW, LINE, BOX, CIRCLE, ARC, ELLIPSE are but a few of the many selections to choose from. FILL objects with preselected PATTERNS; add TEXT; SAVE and RECALL designs to/from disk. You can define your own library of intricate symbols/objects with the easy-to-use OBJECT MANAGEMENT SYSTEM–it will store up to 104 separate objects.
C-128 $59.95
C-64 $39.95

For school or software development, select **SUPER C**. Learn to use C on your Commodore with our in-depth tutorial. Compile your C programs into *fast* machine language. C-128 version has added features: Unix™-like operating system; 60K RAM disk for fast editing and compiling and is usable in your C programs; Use up to 7 disk drives; Linker combines up to 10 modules; Combine M/L and C using CALL; Up to 51K available for object code; Fast loading (8 sec. 1571, 18 sec. 1541); Two standard I/O librarys plus two additional libraries—math functions (sin, cos, sqrt, etc.) & 20+ graphic commands (line, fill, dot, etc.).
C-128 $79.95
C-64 $79.95

Not just a compiler, but a complete system for developing applications in Pascal. Extensive editor with search, replace, auto, renumber, etc. Standard J & W compiler that generates fast machine code. If you want to learn Pascal or to develop software using the best tools available–**SUPER Pascal** is your first choice.
C-128 $59.95
C-64 $59.95



For the professional who wants to easily create high quality charts and graphs without programming. You can immediately change the scaling, labeling, axis, bar-filling, etc. to suit your needs. Accepts data from CalcResult and MultiPlan. C-128 version has 3X the resolution of the '64 version. Outputs to most printers.
C-128 $39.95
C-64 $39.95

OTHER TITLES AVAILABLE:

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. Enter data from the keyboard or from online financial services. **C-64 $59.95**

Personal Portfolio Manager
Complete portfolio management system for the individual or professional investor. Easily manage your portfolios, obtain up-to-the minute quotes and news, and perform selected analysis. Enter quotes manually or automatically through Warner Computer Systems. **C-64 $39.95**

Xper
XPER is the first "*expert system*" for the C-128 and C-64. While ordinary data base systems are good for reproducing facts, XPER can derive knowledge from a mountain of facts and help you make expert decisions. Large capacity. Complete with editing and reporting. **C-64 $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. Power-Graph, the graphics package, is included to create integrated graphs & charts. **C-64 $39.95**

C-128 and C-64 are trademarks of Commodore Business Machines Inc.
Unix is a trademark of Bell Laboratories

Abacus Software

P.O. Box 7219 Grand Rapids, MI 49510 - Telex 709-101 - Phone (616) 241-5510

Call **now** for the name of your nearest dealer. Or to order directly by credit card, MC, AMEX of VISA call **(616) 241-5510**. Other software and books are available–Call and ask for your **free** catalog. Add $4.00 for shipping per order. Foreign orders add $12.00 per item. Dealer inquires welcome–1400+ nationwide.

www.commodore.ca

C-128™ and C-64™ REQUIRED READING

Detailed guide presents the 128's operating system, explains graphic chips, Memory Management Unit, 80 column graphics and commented ROM listings. 500pp $19.95

Get all the inside information on BASIC 7.0. This exhaustive handbook is complete with commented BASIC 7.0 ROM listings. Coming Summer'86. $19.95

Filled with info for everyone. Covers 80 column hi-res graphics, windowing, memory layout, Kernal routines, sprites, software protection, autostarting. 300pp $19.95

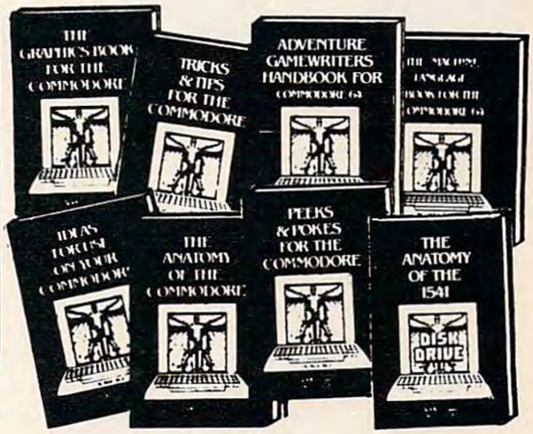
Insiders' guide for novice & advanced users. Covers sequential & relative files, & direct access commands. Describes DOS routines. Commented listings. 450pp $19.95

Learn fundamentals of CAD while developing your own system. Design objects on your screen to dump to a printer. Includes listings for '64 with Simon's Basic. 300pp $19.95

Introduction to programing; problem analysis; thorough description of all BASIC commands with hundreds of examples; monitor commands; utilities; much more. 200pp $19.95

Presents dozens of programming quick-hitters. Easy and useful techniques on the operating system, stacks, zero-page, pointers, the BASIC interpreter and more. $19.95

Essential guide for everyone interested in CP/M on the 128. Simple explanation of the operating system, memory usage, CP/M utility programs, submit files & more. $19.95



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

ANATOMY OF 1541 DRIVE Best handbook on floppy explains all. Many examples and utilities. Fully commented 1541 ROM listings. 500pp $19.95 (REVISED & EXPANDED)

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

GRAPHICS BOOK C-64 - best reference covers basic and advanced graphics. Sprites, animation, Hires, Multicolor, lightpen, 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, more. 275pp $19.95

1541 REPAIR & MAINTENANCE Handbook describes the disk drive hardware. Includes schematics and techniques to keep 1541 running. 200pp $19.95

ADVANCED MACHINE LANGUAGE Not covered elsewhere: - video controller, interrupts, timers, clocks, I/O, real time, extended BASIC, more. 210pp $14.95

PRINTER BOOK C-64/VIC-20 Understand Commodore, Epson-compatible printers and 1520 plotter. Packed: utilities; graphics dump; 3D-plot; commented MPS801 ROM listings, more. 330pp $19.95

SCIENCE/ENGINEERING ON C-64 In depth intro to computers in science. Topics: chemistry, physics, biology, astronomy, electronics, others. 350pp $19.95

CASSETTE BOOK C-64/VIC-20 Comprehensive guide; many sample programs. High speed operating system fast file loading and saving. 225pp $14.95

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

COMPILER BOOK C-64/C-128 All you need to know about compilers: how they work; designing and writing your own; generating machine code. With working example compiler. 300pp $19.95

Adventure Gamewriter's Handbook Step-by-step guide to designing and writing your own adventure games. With automated adventure game generator. 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 to get the most out of your '64. 200pp $14.95

Optional Diskettes for books For your convenience, the programs contained in each of our books are available on diskette to save you time entering them from your keyboard. Specify name of book when ordering. $14.95 each

C-128 and C-64 are trademarks of Commodore Business Machines Inc.

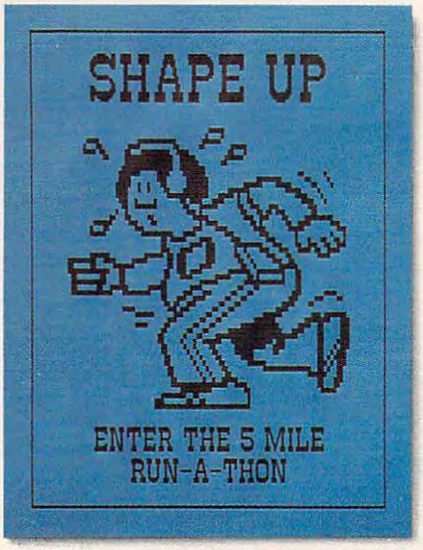
Abacus Software

P.O. Box 7219 Grand Rapids, MI 49510 - Telex 709-101 - Phone (616) 241-5510

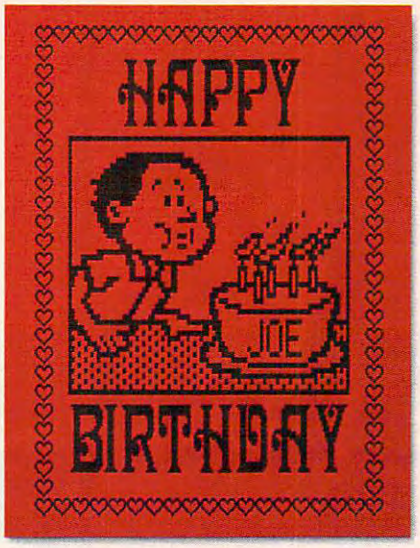
Call **now** for the name of your nearest dealer. Or to order directly by credit card, MC, AMEX of VISA call **(616) 241-5510**. Other software and books are available–Call and ask for your **free** catalog. Add $4.00 for shipping per order. Foreign orders add $10.00 per book. Dealer inquires welcome–1400+ nationwide.

www.commodore.ca

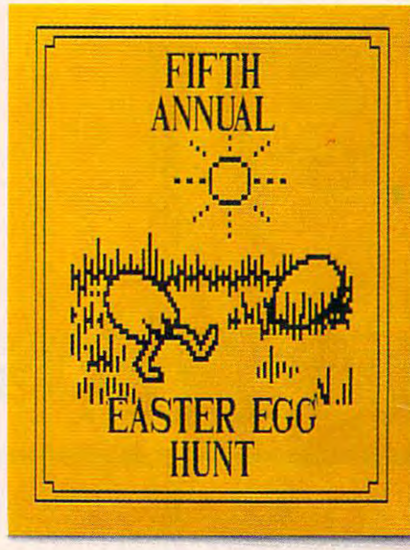
BET YOU NEVER PICTURED THE PRINT SHOP™ DOING THIS.



The Graphics Expander™ has over 300 outstanding new graphics that add dazzle and flair to your Print Shop™ creations.



The Graphics Expander™ has powerful drawing and editing tools to easily modify and customize any graphic.



Now you can actually combine graphics for cards, banners and signs that no one can match. That's being creative!

If you use The Print Shop,™ you need the Graphics Expander™ Volume 1.

You get over 300 new, useful graphics which can be combined and arranged as you wish and then brought into The Print Shop. Make your work come alive with a variety of impressive, realistic art or select from delightful, cartoon-like characters. There's so much to choose from with the Graphics Expander.

Use these graphics exactly as they appear to give your Print Shop creations a refreshing, unique look. Or, have fun customizing the graphics to make each one perfect for the occasion. With the Graphics Expander you can easily add a specific number of candles to a birthday cake or put the coach's name right on a football.

It's easy because the Graphics Expander provides powerful drawing and editing tools

that work just as well with graphics from The Print Shop or The Print Shop Graphics Library.™ You can select from a variety of drawing pens, fill patterns, automatic lines, circles or boxes. Flip graphics left to right or magnify them for detail work. The tools are fun to use and make it easy to personalize your Print Shop creations.

Futhermore, the Graphics Expander allows you to make Print Shop graphics out of your favorite computer pictures from any source such as a Koala Pad™ or digitizer. Any portion of a standard high resolution picture can now be part of your banners, signs and greeting cards.

With The Graphics Expander Volume 1, not only do you have access to over 300 wonderful new graphics, but with the drawing and editing tools, you have access to your imagination. Picture that!

SPRINGBOARD

Springboard Software • 7808 Creekridge Circle • Minneapolis, • 55435 • (612) 944-3915.

Available for the Apple II +, IIe, IIc. IBM and Commodore versions are coming soon.
The Print Shop and The Print Shop Graphics Library are trademarks of Broderbund Software, Inc.
Koala Pad is a trademark of Koala Technologies Corp.

www.commodore.ca

COMPUTE!

MAY 1986
VOLUME 8
NUMBER 5
ISSUE 72

GUIDE TO ARTICLES AND PROGRAMS

FEATURES

REVIEWS

COLUMNS AND DEPARTMENTS

THE JOURNAL

NOTE: See page 108 before typing in programs.

AP Apple, **Mac** Macintosh, **AT** Atari, **ST,** Atari ST, **V** VIC-20, **64** Commodore 64, **+4** Commodore Plus/4, **16** Commodore 16, **128** Commodore 128, **P** PET/CBM, **TI** Texas Instruments, **PC** IBM PC, **PCjr** IBM PCjr, **AM** Amiga. *General interest.

TOLL FREE Subscription Order Line
800-247-5470 (In IA 800-532-1272)

COMPUTE! Publications, Inc. abc
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies
ABC Publishing, President, Robert G. Burton
1330 Avenue of the Americas, New York, New York 10019

COMPUTE! The Journal for Progressive Computing (USPS: 537250) is published monthly by COMPUTE! Publications, Inc., 825 7th Ave., New York, NY 10019 USA. Phone: (212) 265-8360. Editorial Offices are located at 324 West Wendover Avenue, Greensboro, NC 27408. Domestic Subscriptions: 12 issues, $24. POSTMASTER: Send address changes to: **COMPUTE!** Magazine, P.O. Box 10955, Des Moines, IA 50950. Second class postage paid at Greensboro, NC 27403 and additional mailing offices. Entire contents copyright ©1986 by COMPUTE! Publications, Inc. All rights reserved, ISSN 0194-357X.

www.commodore.ca

Editor's Notes

A recent book, *Alan Turing: The Enigma,* is a fascinating study of the life of a brilliant scientist and of the development of early "computing engines" which helped decode secret German messages and significantly contributed to the Allied victory in World War II.

Turing worked with primitive decoders. Called *Bombes* because they ticked loudly, they were something like old-style adding machines, computing with gears and wheels—all mechanical in the days before the electronic revolution. In fact, much of the computation was done by hundreds of women on an assembly line:

> ...the Bombes ticked away, getting on with the work...while the Wrens [Women's Royal Naval Service] did their appointed tasks, without knowing what any of it was for. He [Turing] was fascinated by the fact that people could be taking part in something clever, in a quite mindless way.
>
> Machines, and people acting like machines, had replaced a good deal of human thought, judgment, and recognition. Few knew how the system worked, and for anyone else, it was a mystic oracle, producing an unpredictable judgment. Mechanical, determinate processes were producing clever, astonishing decisions.

Indeed, this large room of workers surrounding the Bombe suggested nothing to Turing so much as a giant machine. Here a group was mechanically adding results; over there was another crew responsible for feeding information back into the Bombe. Some people had to file information, some had to compare a template against each new pattern as it was passed down the line. We can now easily recognize that these activities are the elements of computers and software: RAM, ROM, masking, CPU, feedback loops, branching, and so forth. In those days, however, it took genius to see that the Bombe could be expanded to take over and speed up the functions of the hundreds of clerks working around it.

U-boats were sinking ships all over the Atlantic. Turing and his associates were always working against time, trying to decode messages faster. Eventually, they began to experiment with ways to store information electronically. It's intriguing to read of their efforts to hold onto a few bits of information for a brief time. One of the best solutions they came up with was to store the bits in a cathode ray tube, an early TV screen. This had the advantage that you could amuse yourself by watching the bits flickering while they briefly rested until needed again by the central processor.

But Turing's most famous contribution to computing is the related concepts we now call the *Turing machine* and the *Turing test.* His idea of the machine revealed that he achieved the first comprehensive understanding of the possibility of artificial intelligence. He imagines a universal machine, one that could perform the job of all the other, more specialized, machines. Adding machines operated according to fixed rules which were reflected in their metal cogs and gears. The Bombe, too, performed its job because its mechanism was physically shaped in certain ways.

Turing thought of "tapes" which could contain instructions describing the "state of mind" of the adding machine, the bombe, or any other calculating engine including human "computers." A tape could be fed into a supermachine, and it would then adapt to the state of mind, the description of some other machine, contained on the tape. In this way, the supermachine could "perform the equivalent of human mental activity. A single machine to replace the human computer! An electric brain!"

By the 1950s, Turing had completely formulated another startling concept: How can you tell if a machine is truly *thinking*? The Turing test is deceptively simple: If a questioner cannot tell the difference between written answers from two intelligences, then, for any practical purpose *there is no difference between the intelligences.*

> He imagined a game in which an interrogator would have to decide, on the basis of written replies alone, which of two people in another room was a man and which a woman.... They would alike be making claims such as "I am the woman, don't listen to him!".... A successful imitation of a woman's responses by a man would *not* prove anything. Gender depended on facts which were *not* reducible to sequences of symbols. In contrast, he wished to argue that such an imitation principle did apply to "thinking" or "intelligence." If a computer, on the basis of its written replies to questions, could not be distinguished from a human respondent, then "fair play" would oblige one to say that it must be "thinking."
>
> ...he produced an argument in favor of adopting the imitation principle as a criterion. This was that there was no way of telling that other *people* were "thinking" or "conscious" except by a process of comparison with oneself, and he saw no reason to treat computers any differently.

Turing expected a machine to pass his test around the end of this century:

> I believe that in about fifty years' time it will be possible to programme computers, with a storage capacity of about 10^9, to make them play the imitation game so well that an average interrogator will not have more than 70 per cent chance of making the right identification after five minutes of questioning. The original question, "can machines think?" I believe to be too meaningless to deserve discussion. Nevertheless I believe that at the end of the century the use of words and general educated opinion will have altered so much that one will be able to speak of machines thinking without expecting to be contradicted.

Ten to the ninth power is 122,070K or 119 megabytes. (Turing's 10^9 represents bits). One hundred and nineteen megabytes is not an uncommon storage capacity nowadays—we've got more than that here at COMPUTE! in the hard disks servicing our minicomputer editing system. Yet our system would never pass the Turing test.

Nonetheless, Turing's machine, his test, and his other ideas continue to have enormous impact, and *Alan Turing: The Enigma* is a lively, understandable portrait of a major thinker's life and ideas. If you're curious about where computers came from and where they're likely to go from here, you'll enjoy this book very much indeed.

Senior Editor

www.commodore.ca

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.

Apple II is a trademark of Apple Computer, Inc.
Atari XL and XE are trademarks of Atari Corp.
Commodore 64 and 128 are trademarks of Commodore Electronics Ltd.
IBM PC is a registered trademark of International Business Machines Corp.

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

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

www.commodore.ca

Publisher	James A. Casella
Founder/Editor in Chief	Robert C. Lock
Senior Editor	Richard Mansfield
Managing Editor	Kathleen Martinek
Executive Editor	Selby Bateman

Editor	Tom R. Halfhill
Assistant Editor	Philip Nelson
Production Director	Tony Roberts
Production Editor	Gail Cowper
Editor, COMPUTE!'s GAZETTE	Lance Elko
Technical Editor	Ottis R. Cowper
Assistant Technical Editor	George Miller
Program Editor	Charles Brannon
Assistant Editor, COMPUTE!'s GAZETTE	Todd Heimarck
Assistant Features Editor	Kathy Yakal
Programming Supervisor	Patrick Parrish
Editorial Programmers	Tim Victor, Kevin Mykytyn
Research/Copy Editor	Joan Rouleau
Copy Editor	Ann Davies
Submissions Reviewer	Mark Tuttle
Programming Assistants	David Florance, David Hensley
Executive Assistant	Debi Nash
Administrative Assistants	Julia Fleming, Iris Brooks, Mary Hunt, Sybil Agee
Associate Editors	Jim Butterfield, Toronto, Canada; Harvey Herman, Greensboro, NC; Fred D'Ignazio, Roanoke, VA; David Thornburg, Los Altos, CA
Contributing Editor	Bill Wilkinson

COMPUTE!'s Book Division	
Editor	Stephen Levy
Assistant Editor	Gregg Keizer
Director, Book Sales & Marketing	Steve Voyatzis

Production Manager	Irma Swain
Art & Design Director	Janice R. Fary
Assistant Editor, Art & Design	Lee Noel
Mechanical Art Supervisor	De Potter
Artists	Debbie Bray, Dabney Ketrow
Typesetting	Terry Cash, Carole Dunton
Illustrator	Harry Blair

Director of Advertising Sales	Peter Johnsmeyer
Associate Advertising Director	Bernard J. Theobald, Jr.
Production Coordinator	Kathleen Hanlon

Promotion Assistant	Caroline Dark

Customer Service Manager	Diane Longo
Dealer Sales Supervisor	Orchid Tamayo
Individual Order Supervisor	Cassandra Green
Receptionist	Anita Armfield
Warehouse Manager	John Williams

Data Processing Manager	Leon Stokes

James A. Casella, President
Richard J. Marino, Vice President, Advertising Sales

COMPUTE! Publications, Inc. publishes:

COMPUTE! The Journal For Progressive Computing

COMPUTE!'s GAZETTE For VIC-20 And Commodore 64 Personal Computers

COMPUTE! Books

COMPUTE!'s GAZETTE DISK

COMPUTE!'s Apple Applications Special

Editorial offices:	324 West Wendover Avenue Suite 200 Greensboro, NC 27408 USA
Corporate offices:	825 7th Avenue New York, NY 10019 212-265-8360
Customer Service:	800-346-6767 (In NY 212-887-8525)
Hours:	9:30 A.M.–4:30 P.M. Monday–Friday

Coming In Future Issues

Buyer's Guide To The Printers Of 1986

Looking Glass: Powerful Screen Windows For The Commodore 64

Atari Password Protect Your Programs From Prying Eyes

The Screen Machine II Graphics-Design Canvas For IBM PC, PCjr, and Compatibles

Alphabetized Directories For Commodore

Miami Ice: Slick Action Game For Commodore, Atari, Amiga and Others

Member Audit Bureau of Circulations

MPA Magazine Publishers Association

Subscription Orders

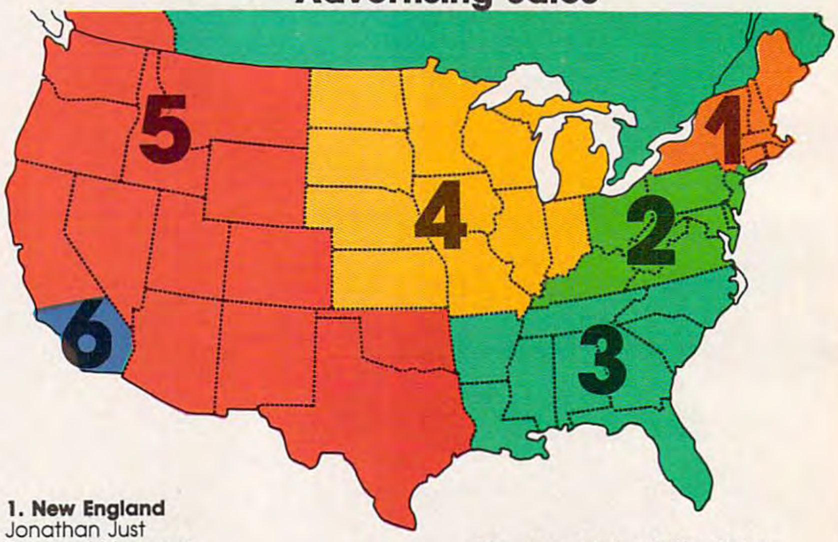
COMPUTE!
P.O. Box 10954
Des Moines, IA 50340

TOLL FREE Subscription Order Line
800-247-5470
In IA 800-532-1272

COMPUTE! Subscription Rates (12 Issue Year):

US	(one yr.)	$24
	(two yrs.)	$45
	(three yrs.)	$65
Canada and Foreign Surface Mail		$30
Foreign Air Delivery		$65

Advertising Sales



1. New England
Jonathan Just
Regional Manager
212-315-1665

2. Mid Atlantic
Jonathan Just
Regional Manager
212-315-1665

3. Southeast & Foreign
Harry Blair
919-275-9809

4. Midwest
Gordon Benson
312-362-1821

5. Northwest/ Mountain/Texas
Phoebe Thompson
Dani Nunes
408-354-5553

6. Southwest
Ed Winchell
213-378-8361

Director of Advertising Sales:
Peter Johnsmeyer

Associate Advertising Director:
Bernard J. Theobald, Jr.

COMPUTE! Home Office 212-887-8460.

Address all advertising materials to:
Kathleen Hanlon
Advertising Production Coordinator
COMPUTE! Magazine
324 West Wendover Avenue
Suite 200
Greensboro, NC 27408

The COMPUTE! subscriber list is made available to carefully screened organizations with a product or service which may be of interest to our readers. If you prefer not to receive such mailings, please send an exact copy of your subscription label to: COMPUTE! P.O. Box 10955, Des Moines, IA 50950. Include a note indicating your preference to receive only your subscription.

Authors of manuscripts warrant that all materials submitted to COMPUTE! are original materials with full ownership rights resident in said authors. By submitting articles to COMPUTE!, authors acknowledge that such materials, upon acceptance for publication, become the exclusive property of COMPUTE! Publications, Inc. No portion of this magazine may be reproduced in any form without written permission from the publisher. Entire contents copyright © 1986, COMPUTE! Publications, Inc. Rights to programs developed and submitted by authors are explained in our author contract. Unsolicited materials not accepted for publication in COMPUTE! will be returned if author provides a self-addressed, stamped envelope. Programs (on tape or disk) must accompany each submission. Printed listings are optional, but helpful. Articles should be furnished as typed copy (upper- and lowercase, please) with double spacing. Each page of your article should bear the title of the article, date and name of the author. COMPUTE! assumes no liability for errors in articles or advertisements. Opinions expressed by authors are not necessarily those of COMPUTE!.

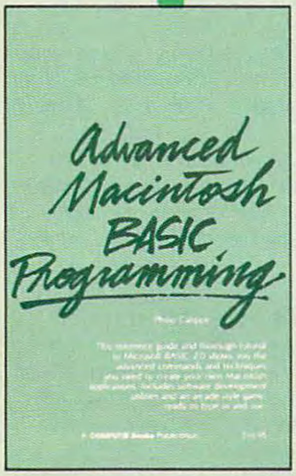
PET, CBM, VIC-20 and Commodore 64 are trademarks of Commodore Business Machines, Inc. and/or Commodore Electronics Limited
Apple is a trademark of Apple Computer Company
IBM PC and PCjr are trademarks of International Business Machines, Inc.
ATARI is a trademark of Atari, Inc.
TI-99/4A is a trademark of Texas Instruments, Inc.
Radio Shack Color Computer is a trademark of Tandy, Inc.

www.commodore.ca

All The Best

APPLE INFORMATION

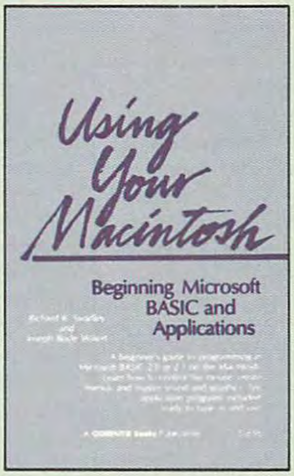
Take advantage of the spectacular special features of your Apple II-series and Macintosh computers with these bestsellers from COMPUTE! Books.



Advanced Macintosh BASIC Programming

Philip Calippe, 309 pages
A reference guide and tutorial to Microsoft BASIC which shows you how to use the Macintosh's advanced features to create impressive programs. *A disk is also available which includes programs in the book, $15.95 (0300BDSK).*

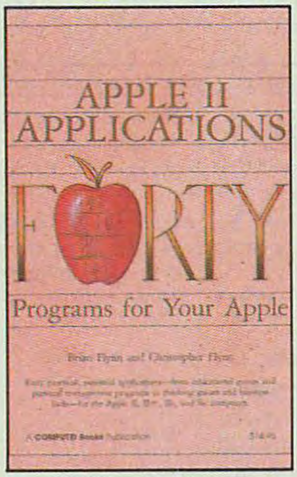
$16.95 ISBN 0-87455-030-0



Using Your Macintosh: Beginning Microsoft BASIC and Applications

Richard K. Swadley and Joseph Boyle Wikert, 274 pages
Necessary and easy-to-understand information about the revolutionary Macintosh along with clear, easy-to-follow explanations of BASIC. Everything from writing your first statement to creating a finished program. *A disk is also available which includes programs in the book, $15.95 (0211BDSK).*

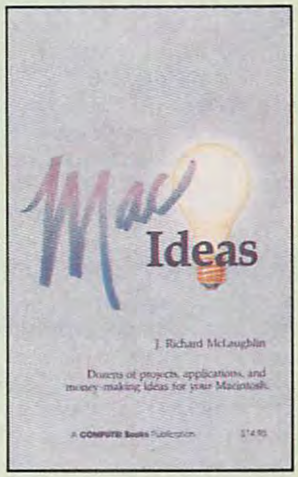
$16.95 ISBN 0-87455-021-1



Apple II Applications: Forty Programs for Your Apple

Brian Flynn and Christopher Flynn, 374 pages
Forty educational and strategy games, business and science applications, and home and personal organizational tools to use on any Apple II-series computer. *A disk package is also available which includes programs in the book. Specify DOS 3.3 or ProDOS, $12.95*

$14.95 ISBN 0-87455-016-5



MacIdeas

J. Richard McLaughlin, 240 pages
More than 100 ways to utilize the Macintosh's powerful graphics capabilities. Beautify everything from personal gifts to correspondence, and learn how to use digitizers to create dazzling graphics.

$14.95 ISBN 0-87455-015-7

Mail this coupon with your payment to COMPUTE! Books, P.O. Box 5038, F.D.R. Station, New York, NY 10150.
Or call toll-free 800-346-6767 (in NY 212-887-8525).

Please send me the following books and disks:

- ____ **Advanced Macintosh BASIC Programming (030-0), $16.95 each** ____
- ____ **Advanced Macintosh BASIC Programming Disk (0300BDSK), $15.95 each** ____
- ____ **Apple II Applications (016-5), $14.95 each** ____
- ____ **Apple II Applications Disk, DOS 3.3 (0165DSK1), $12.95 each** ____
- ____ **Apple II Applications Disk, ProDOS (0165DSK2), 12.95 each** ____
- ____ **Apple II Applications Book and DOS 3.3 Disk Combination package (050-5), $29.95 each** ____
- ____ **MacIdeas (015-7), $14.95 each** ____
- ____ **Using Your Macintosh (021-1), $16.95 each** ____
- ____ **Using Your Macintosh Disk (0211BDSK), $15.95 each** ____

Subtotal ____
NC residents add 4.5% sales tax ____
Shipping and handling (In U.S. $2.00 per disk or book; airmail $5.00 per item) ____
Total enclosed ____

All orders must be prepaid in U.S. funds.
Payment enclosed (check or money order)
☐ Charge ☐ MasterCard ☐ Visa ☐ American Express

Account No. ____ Exp. Date ____ (required)
Name ____
Address ____
City ____ State ____ Zip ____

Please allow 4–6 weeks for delivery.
Prices subject to change without notice.

36412214

COMPUTE! Publications, Inc. abc
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor, New York, NY 10019
Publishers of COMPUTE!, COMPUTE!'s Gazette, COMPUTE!'s Gazette Disk, COMPUTE! Books, and COMPUTE!'s Apple Applications

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, England, and in Canada from McGraw-Hill, Ryerson Ltd., 330 Progress Ave., Scarborough, Ontario, Canada M1P 2Z5.

www.commodore.ca

Readers Feedback

The Editors and Readers of COMPUTE!

If you have any questions, comments, or suggestions you would like to see addressed in this column, write to "Readers' Feedback," COMPUTE!, *P.O. Box 5406, Greensboro, NC 27403. Due to the volume of mail we receive, we regret that we cannot provide personal answers to technical questions.*

Unique Random Numbers

I am writing a program that requires random numbers to control the game, but the problem is I want to filter out the random numbers that have already been picked. I'm using an Atari computer. What's the solution?

Daren O'Brien

There are a couple of solutions to this problem, and they apply to nearly any version of BASIC. Which method is fastest and most efficient depends on the particular requirements of your program. You may be writing an arcade-style game which needs to position characters or objects randomly about the screen, each in a different location; or you may be writing a card game simulation that requires a "shuffled" pile of 52 random numbers.

The most common approach is to generate a random number, check it against a list of previously generated random numbers stored in an array, and then generate a new number if there's a match. This can be done by a subroutine that your program calls whenever it needs a unique number. Here's an example in Atari BASIC that works with little or no modification in most other BASICs:

```
10 SIZE=100:DIM ARRAY(SIZ
   E):A=1
20 FOR J=1 TO SIZE:ARRAY(
   J)=0:NEXT J
30 GOSUB 1000:REM GET UNI
   QUE RANDOM NUMBER
40 PRINT "UNIQUE VALUE #"
   ;A-1;" = ";RNUM
50 IF A<SIZE+1 THEN 30
60 END
1000 RNUM=INT(RND(1)*1000
   +1)
1010 PRINT RNUM
1020 J=1
1030 IF RNUM=ARRAY(J) THE
   N 1000
1040 J=J+1:IF J<SIZE+1 TH
   EN 1030
1050 ARRAY(A)=RNUM:A=A+1:
   RETURN
```

Line 10 defines the size of the array which stores the random numbers and sets A=1 as an index into the array. You can change the variable SIZE, of course, to create any size array you need. Line 20 clears out the array with zeroes. The statements in these two lines should be placed in the initialization section of your program, followed by your own code.

Line 30 calls the random number subroutine, which begins at line 1000. The statement in line 1000 generates a random number (RNUM) between 1 and 1000 in Atari BASIC. Make whatever changes are appropriate for your particular program or version of BASIC. Line 1010 can be omitted in your own program—it simply prints the newly generated random number on the screen for this demonstration. Lines 1020–1040 set up a loop which compares RNUM to each element in the array. If a match is found—meaning the random number was previously generated and stored in the array—the program goes back to line 1000, which makes a new random number and repeats the process. If no match is found anywhere in the array—meaning the random number is unique—the program continues to line 1050. This line adds the number to the array, increments the array index (A=A+1), and finally ends the subroutine with a RETURN.

(Incidentally, some of you may be wondering why we didn't use a FOR-NEXT statement for the loop in lines 1020–1040. We didn't because the IF-THEN statement in line 1030 frequently jumps out of the loop back to line 1000 whenever it finds a match, and this could eventually cause a stack overflow and out-of-memory error.)

Lines 40 and 50 can be omitted from your own program. They merely print the random numbers on the screen for this demonstration.

This method of generating unique random numbers works, but suffers from a few problems. For one thing, if you need a great many random numbers, or if you can't predict how many you'll need as your program runs, your computer may not have enough memory for the large array that's required. Also, if the number *of random numbers you need coincides with the allowable* range *of the random numbers, this routine slows down almost to a crawl as it struggles to generate the last few unique numbers.*

An example of the latter problem is when you're programming a card game simulation and need a randomized list of 52 unique numbers from 1 to 52, each number representing a card. To see a demonstration, set SIZE=52 in line 10 and modify line 1000 so it generates random numbers restricted to the range of 1–52. When you run the program, at first it has no trouble coming up with unique numbers. But soon it begins exhausting the possibilities. By the time it reaches the forty-fifth unique number, it is having real trouble generating numbers which haven't been made before. The fifty-second number is the hardest of all—it might take a minute or longer. (There are better ways to shuffle a list of numbers for card games, but we haven't room to cover them here.)

If repeated calls to this subroutine slow down your program too drastically, rewrite it slightly to fill the array with all the unique numbers your program will need. Then call the routine just once, as your program initializes. Later, whenever your program needs a unique random number, it can simply pull a number out of the array, incrementing a counter each time so the same number isn't retrieved more than once.

Wide *SpeedCalc* Printouts

Does *SpeedCalc* work with the *Sideways* program? I would like the ability to print out a spreadsheet with lots of columns.

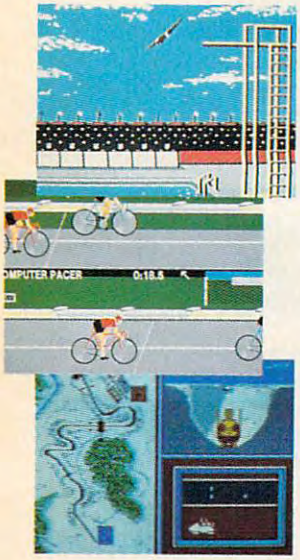
Stephen Forstein

Because SpeedCalc *allows as many as 50 cells per horizontal row, many worksheets can't be printed on a standard 80-column printer. For example, a 12-month worksheet with an extra left cell for row titles and an extra right column cell for totals will be 126 columns wide (if you use the default column width of nine characters per cell). If you print this in the normal manner, you'll find that each worksheet row wraps to one and a half lines on the printout, making the sheet difficult to read.*

A crude solution is to divide the completed sheet into several sections—each no more than 80 columns wide—then print the sections separately and tape them together. Condensed printing mode offers a more elegant solution, if

www.commodore.ca

22 FAST-PACED, EXCITING EVENTS. GO FOR THE GOLD!



SUMMER GAMES.®
Want some great play action? This award winning best seller brings you eight great events, including Swimming, Diving, Track, Gymnastics, and more. Compete against world records. Or get together with a group of up to eight for some good competitive *fun*. Realistic, graphics and *action* will challenge you again and again to *go for the gold*.

SUMMER GAMES II.™
You asked us for more great events. Here they are. Rowing, Triple Jump, Javelin, High Jump, Fencing, Cycling, Kayaking, even Equestrian. Like SUMMER GAMES, you get spine tingling *action and challenge* for one to eight players. These aren't just games. They're *the games*. And you're the star.

WINTER GAMES.™
You've got to see the graphics, and *feel* the knot in your stomach as your bobsled careens along the walls of the run to know why people are wild over WINTER GAMES. Seven events give you a variety of action—from the endurance of the Biathlon to the timing of the Ski Jump, and more.

What are you waiting for?
Play your favorite events over and over. Play all 22. Set up teams. Challenge your friends. These great action-strategy games are sure to be the stars of *your* collection.

At your local Epyx dealer.

	APPLE	MAC	ATARI	ATARI ST	C64/128	AMIGA
Summer Games	✓		✓		✓	
Summer Games II	✓				✓	
Winter Games	✓	✓		✓	✓	✓

EPYX® COMPUTER SOFTWARE
1043 Kiel Ct., Sunnyvale, CA 94089

www.commodore.ca

your printer supports it (Commodore printers do not). On most printers this allows a 132-character line, enough for more than 14 default-width cells across. You'll need to set the printer for condensed printing before you run SpeedCalc, *either by setting the appropriate DIP switch on the printer or by sending the proper command codes. For example, CHR$(15) works for Epson and Epson-compatible printers. You may have to use "transparent mode" to send this command through your interface. For instance, Cardco interfaces require OPEN 4,4,4: PRINT#4,CHR$(15): CLOSE 4.*

With the Commodore (January 1986) and Apple (February 1986) versions of SpeedCalc, *the commercial program* Sideways *does let you print your sheets vertically down the page instead of horizontally across. This allows you to print spreadsheets of virtually unlimited width. The only restriction is that* Sideways *requires ASCII text files. For the Apple, just use Open Apple-CTRL-P to print a copy of the sheet to disk. For the 64/128, the procedure is slightly more complicated, since* Sideways *requires that the data be in a SEQ (sequential) file and* SpeedCalc *prints a PRG (program) file to disk. (The* SpeedCalc *article in the January issue is in error when it states that printing to disk stores the data in a sequential file.) To use* Sideways *with Commodore* SpeedCalc, *you must print (not save) the sheet to disk, then convert the printed data file from PRG format to SEQ format. The short program below performs this conversion:*

```
10 PRINT"{CLR}{DOWN}{RVS} PRG
     {SPACE}-> SEQ FILE CONVE
     RTER ":Z$=CHR$(0):OPEN 3
     ,8,15,"I0:":GOSUB 100
20 FP$="":INPUT"NAME OF PRG-TY
     PE FILE";FP$:IF FP$="" T
     HEN 20
30 OPEN 1,8,8,"0:"+FP$+",P,R":
     GOSUB 100:IF S THEN CLOS
     E 1:GOTO 20
40 FS$="":INPUT"NAME FOR SEQ-T
     YPE FILE";FS$:IF FS$=""
     {SPACE}THEN 40
50 OPEN 2,8,9,"0:"+FS$+",S,W":
     GOSUB 100:IF S THEN CLOS
     E 2:GOTO 40
60 PRINT"WORKING...";:GET#1,A$
     ,A$
70 GET#1,A$:A=ASC(A$+Z$):S=ST:
     PRINT#2,CHR$(A);:IF (S O
     R ST)=0 THEN 70
80 IF (S<>64) OR ST THEN A$=""
     :GOTO 120
90 CLOSE 2:CLOSE 1:CLOSE 3:PRI
     NT"DONE":END
100 INPUT#3,S,A$:IF S=0 THEN P
     RINT:RETURN
110 IF (S=62) OR (S=63) THEN P
     RINT"{RVS}";A$:RETURN
120 CLOSE 2:CLOSE 1:CLOSE 3:PR
     INT"{DOWN}{RVS}DISK ERRO
     R: ";A$
```

Sideways *is a product of Funk Software (222 Third Street, Cambridge, MA 02142); the Commodore 64/128 version is distributed by Timeworks, Inc. (444 Lake Cook Road, Deerfield, IL 60015).*

Atari ST Languages

What languages are available for the Atari ST?

Randy Johnson

Atari includes ST BASIC and Logo with the purchase of an ST system. Below is a list of additional languages available or under development for the ST at the time of this writing (mid-February). This list is by no means complete—new ST applications, including languages, are announced on an almost daily basis. Your Atari dealer is a good source of information about new products. You should also keep an eye on the product evaluations in COMPUTE!'s *"Reviews" and "News & Products" sections. Advertisements are timely information sources as well, since it's common for a developer to announce a product in ads before actually releasing it on the market.*

Language	*Company*
Compiled BASIC, Henry's Fundamental BASIC, C, Pascal, FORTRAN	*Philon 641 Avenue of the Americas New York, NY 10011*
Haba Hippo-C	*Haba Systems 15154 Stagg Street Van Nuys, CA 91405*
Lattice C, Meta Pascal, Macro Assembler Editor	*Antic Software 524 Second Street San Francisco, CA 94107*
Personal Pascal, Personal Prolog	*Optimized Systems Software (OSS) 1221-B Kentwood Avenue San Jose, CA 95129*
4 × FORTH	*The Dragon Group 148 Poca Fork Road Elk View, W. VA 25071*
Modula-2	*TDI Software Inc. 10410 Markison Road Dallas, TX 75238*

Reading Apple Keys

I have just read Apple *SpeedCalc* in COMPUTE!'s February 1986 issue and noticed that the program's commands use the Open Apple and Closed Apple keys. How do you read those keys? They don't show up in INPUT statements or when the keyboard is read. Can I access them from BASIC or only from machine language?

David Reed

The Open Apple and Closed Apple keys, which appear only on Apple IIe and IIc computers, are easy to read from BASIC or machine language. Both can be read in BASIC by PEEKing certain memory locations. The value in location 49249 ($C061) is 128 or greater if the Open Apple key is pressed, and less than 128 if it is not. Location 49250 ($C062) gives the same information for the Closed Apple key. Here's a program fragment that gets a keypress and also checks these two special keys:

```
100 GET K$ : REM WAIT FOR A
     KEYPRESS
110 OA = PEEK(49249) : CA =
     PEEK(49250)
120 IF K$="a" AND OA >= 128 AND
     CA < 128 THEN GOSUB 1000
```

These statements call a subroutine at line 1000 if Open Apple-A is pressed, but not if Open Apple-Closed Apple-A is pressed.

Amiga Graphics Update

We enjoyed your article on the graphics for Commodore's new Amiga computer ["Amiga's Amazing Graphics," COMPUTE!, November 1985]. Most of the information was very accurate, and the picture describing bit plane graphics was particularly effective. There was some inaccuracy, however. The actual color map at 640 × 200 resolution is the same as it is at 320 × 200, and there are no restrictions on adjacent pixels. Somehow, you must have gotten some old documentation. Also, although Intuition supports dual playfield mode, it does not do multiple screens using dual playfield mode—it uses the video coprocessor to rewrite the display parameters. This is much more flexible, as *any* display parameter can be changed instead of just the color map. You can change resolution, color map, or number of bit planes between screens running simultaneously.

Glenn Keller
Commodore-Amiga, Inc.

We appreciate your comments. At the time we wrote the article, we were working with the prerelease Hardware Manual, *which described an earlier version of the graphics chips.*

As you say, the final graphics hardware does not impose any restrictions on color resolutions. Early graphics chips could not fully change the color signal between adjacent pixels in 640 × 200 mode, but the current hardware permits any of 16 colors (from the palette of 4,096 colors) at any pixel position, or 32 simultaneous colors for the 320 × 200 mode. The 400-line interlaced modes have the same color capability as 320 × 200 and 640 × 200.

Since the Workbench uses only four colors (two bit planes), we assumed that dual playfields (where two independent screens can be overlapped and merged) were used to support pull-down custom screens. Since every multitasking application can call for its own custom bitmap, Intuition allows these custom screens to be overlapped and repositioned vertically with the mouse. This effect can be seen

www.commodore.ca

PREMIUM QUALITY! LIFETIME WARRANTY!

DISK SALE!

Made by top makers, not low-end or seconds. We buy millions of factory overproduction. We can't print the maker, but you'll recognize them as among the HIGHEST QUALITY PREMIUM DISKS MADE! Certified, guaranteed 100% error free. UNCONDITIONAL 30 DAY MONEY BACK SATISFACTION GUARANTEE!

from **35¢**

Prices are per disk.	BULK 100	500	1K	5K	10K	BOXED 10	100	500	1K	5K
SS/DD-35trk for Apple II series #UDA	47¢	45¢	41¢	38¢	35¢	60¢	55¢	52¢	49¢	46¢
SS/DD-for Single side computers #UD1	59¢	57¢	55¢	50¢	46¢	75¢	67¢	65¢	63¢	57¢
DS/DD for IBM-PC & compatibles #UD2	67¢	60¢	58¢	56¢	54¢	85¢	75¢	68¢	66¢	64¢
DS/QUAD DENSITY (96tpi) #UD4	1.09	1.04	99¢	92¢	89¢	1.19	1.11	1.06	1.02	99¢
DS/HD "HI-DENSITY" for IBM-AT #UAT	2.15	2.12	2.10	2.08	1.99	2.40	2.25	2.22	2.20	2.18
3½" single side #UD3	1.86	1.83	1.80	1.77	1.74	1.89	1.86	1.83	1.80	1.77
8" SS/SD soft sect. #UD8						1.20	1.00	1.00	1.00	1.00
5¼" SLEEVES. #24 white wove #SLV	4¢	3.5¢	3¢	2.8¢	2.6¢	Specify BULK or BOXED when ordering disks.				
5¼" SLEEVES. white tyvek #TYV	6¢	5¢	4.5¢	4.3¢	4.1¢					

INCREDIBLE PRICES ON COMPUTER ACCESSORIES!

UNIFILE-100 holds 100 disks, with lock and key, removeable top $13.88
UNIFILE-70, flip-top, holds 70 disks, like "Flip-n-File" $8.88
UNIFILE-10 LIBRARY CASES, holds 10 disks, clear plastic. 99¢
UNIPAK 5¼" DISK MAILERS, cardboard, holds 2-3 disks, 10 for $5
RIGID-PAK 5¼" DISK MAILERS, hard plastic, holds 5 disks $1 ea
LETTER QUAL COMPUTER PAPER 2500 sheets 20# microperf 8½x11" #LQP $29
DRAFT QUAL COMPUTER PAPER 3300 sheets 16# std-perf 8½x11" #DQP $29

HOW TO ORDER: Order by phone or mail, pay by MC-Visa-Amex-COD, or send check with order, minimum order $20, add $3 for shipping (for paper "DQP/LQP": zips 0-2 $3/ctn, zips 3-6 $5/ctn, zips 7-9 $8/ctn). Add $2 if COD. Canada $5 shipping. We can ship open account to schools with good credit, minimum purchase order $100, FOB Unitech. All orders must include daytime phone and STREET address. Send for FREE CATALOG listing 100's of discount computer supplies!

TOLL FREE **(800)343-0472**

UNITECH
IN MASS:(617)"UNITECH"
20-U HURLEY STREET
CAMBRIDGE, MA 02141

ATTENTION T.I. 99/4A OWNERS

- Diskettes - 59¢ each! Your choice SS or DD
- 512K Now Available for the 99/4A!
- 99/8 Level 4 Computer Upgrade Now Available
- Over 1500 Hardware and Software Accessories at Similar Savings

THE WORLD'S LARGEST COMPUTER ASSISTANCE GROUP

Now serving over 35,000 members worldwide with the best in technical assistance, service, and products for the Texas Instrument 99/4A Computer.

To become a member and receive newsletters, catalog, technical assistance and membership package, send $10.00 for a ONE Year Membership to:

99/4A National Assistance Group
National Headquarters
P.O. Box 290812
Ft. Lauderdale, Florida 33329
Attention Membership Division
For Further Information Call 24 Hours
(305) 583-0467

DISKS 69¢

Foolish to pay more.
Dangerous to pay less.

- QUALITY MEDIA
- LIFETIME REPLACEMENT GUARANTEE
- HUB RINGS
- TYVEC EPS.
- WRITE PROTECTS

	1–50	51+
5.25" SSDD	.79	.69
5.25" DSDD	.89	.79
PC FORMATTED	1.09	.99
AT 1.2MB/3.5 1D (Mac)	1.99	CALL

BLACKSHIP COMPUTER SUPPLY
P.O. Box 883362
San Francisco, CA 94188
In California 415-550-0512
USA orders **800-431-6249**
In Canada 403-428-6229

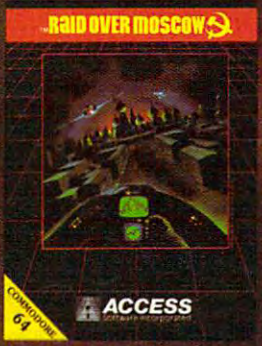
Add $3.00 shipping and handling per 100 Diskettes
COD add $1.95 (CA residents add 6.5% sales tax)
VISA MC COD

NO GUTS.....NO GLORY

ACCESS SOFTWARE'S best selling and award winning war game series is now available for **APPLE IIc IIe II+, ATARI 800** series and **COMMODORE 128/64.** Each game features strategy, arcade action and incredible three dimensional graphics.

BEACH-HEAD - A peaceful island is being held by a ruthless dictator and its up to you to liberate the natives! Be careful, because you'll encounter enemy aircraft, battleships, tanks, and an almost impenetrable enemy fortress!
$34.95 - Commodore/Apple/Atari.

BEACH-HEAD II - The dictator's back and more ruthless than ever! If your troops can survive the land assault and rescue the prisoners, you'll face the dictator in a one on one battle of superiority. Beach-Head is a true "head to head" two player game that allows you to play with a friend or the computer.
$39.95 - Commodore, $34.95 Apple/Atari.



RAID OVER MOSCOW - The alert sounds at SAC (Strategic Air Command). The Soviets are about to launch a nuclear strike! With our own nuclear arsenal dismantled, our only hope is our space station armed with stealth aircraft. Can your commandos turn back the attack, strike the Soviet Defense Center and avert nuclear disaster? Top multiscreen action!
$39.95 - Commodore, $34.95 Apple/Atari.

ACCESS Software Incorporated

For Apple Atari and Commodore

VISIT YOUR FAVORITE RETAILER...

For mail orders, enclose check or money order, plus $1.00 shipping and handling, and specify machine version desired. Order by phone on VISA, MasterCard or C.O.D. by calling 801 / 298-9077.

ACCESS SOFTWARE, INC. 2561 South 1560 West Woods Cross, UT 84087

www.commodore.ca

when you click and drag the menu bar with the left *mouse button. When the left button is held down, pulling the mouse downward reveals the background AmigaDOS screen. It's also possible to switch between a custom screen and the Workbench screen by pressing either Left Amiga-M or Left Amiga-N.*

The special copper *(as in coprocessor) circuitry tracks the video beam on the fly. The copper's instruction list, similar in concept to Atari 800 display lists, can perform any video change at any line, as with display list interrupts (Atari) or raster interrupts (Commodore). The operating system permits applications to modify the copper list, giving full video control to the application, while using the copper list itself for the graphic effect of overlapping screens. A* copper wait *instruction tells the copper to wait until the video beam reaches a certain line, and then the video registers are reset to display another screen. The normal display is automatically reset at the top, so you get two overlapping screens, even with different colors and resolutions.*

TI Tips Book

In an effort to provide easily accessed documentation to TI users, I have put together a TI tips booklet that consists of 99 tips for the TI-99/4A. These are a compilation of suggestions given in our user group newsletter. They include PEEKs, POKEs, listings, hints, and so on. Also included is a complete disk drive memory map, summary of Extended BASIC commands, and a sorting program written in BASIC and machine language. One such tip that may interest your readers allows them to disable FCTN = (QUIT) in Extended BASIC. To do this, enter this statement:

CALL INIT :: CALL LOAD(−31806,16)

To enable it again, type CALL LOAD(−31806,0). Another POKE allows you to prevent Extended BASIC programs from being listed. Type CALL LOAD(−31931,128) to do this. To unprotect Extended BASIC programs, enter CALL LOAD(−31931,0). The TI tips booklet is available through the Central Iowa 99/4A Users Group for $4 (the cost of materials, printing, and postage) at the following address:

Central Iowa 99/4A Users Group
Box 3043
Des Moines, IA 50316

John Hamilton

Thank you for providing this information.

Setting Atari Tabs

I read with interest your examples of programming Atari tab stops using memory location 201. This location is in the BASIC zero page RAM because it is used only by BASIC, and can only be used with PRINT statements containing commas. The true tab function is executed by the operating system (the text editor). It is associated with memory locations 675–689. These memory locations form the tab set map. The highest bit of location 675 corresponds to column zero, and the lowest bit of location 689 corresponds to column 120 of the logical line. Normally every eighth bit is set (as can be seen by experimenting with the TAB key). This can be changed either with the TAB CLEAR and TAB SET keys or by POKEing values to the map locations. For example the following program clears the map with POKEs, then prints the heading while using the TAB SET character, CHR$(159). Then the names and addresses are printed while using the TAB character, CHR$(127).

```
10 FOR I=675 TO 689:POKE
   I,0:NEXT I
20 DIM NAME$(10),ADDRESS$
   (25),TAB$(1),TABSET$(1
   )
30 TAB$=CHR$(127):TABSET$
   =CHR$(159)
40 PRINT "NAME{13 SPACES}"
   ;TABSET$;"ADDRESS"
50 PRINT
60 FOR A=1 TO 4
70 READ NAME$;ADDRESS$
80 PRINT NAME$;TAB$;ADDRE
   SS$
90 NEXT A
100 END
110 DATA ADAMS,12 MAIN ST
    REET
120 DATA ARTHUR,1515 SUNN
    Y STREET
130 DATA SMITHSON,100 CIR
    CLE DRIVE
140 DATA WEEKS,2 DONNA LA
    NE
```

Sherwood Stolt

Thank you for the additional information.

Custom Characters For Plus/4 And 16

I was wondering how the Commodore Plus/4 can generate user-defined characters and where I would POKE them into memory.

Sean Donovan

The following program redefines the @ character as the familiar Commodore logo symbol.

```
FX 10 FORA=828TO860:READB:POKE
      A,B:NEXT:SYS828
AM 20 POKE65298,PEEK(65298)AND
      251
HP 30 POKE65299,PEEK(65299)AND
      3OR60
SR 40 FORA=15360TO15367:READB:
      POKEA,B:NEXT
JA 50 DATA 169,60,133,3,169,0,
      133,2,133,4,169,208,133
AC 60 DATA 5,162,3,160,0,177,4
      ,145,2,136,208,249,230
SK 70 DATA 3,230,5,202,16,242,
      96
HD 80 DATA 98,146,130,130,144,
      98,0,0:REM CUSTOM CHARAC
      TER DATA
```

Two memory locations are important for custom characters on the Commodore Plus/4 and 16. Bit 2 of location 65298 controls whether character data is fetched from ROM or RAM. The POKE in line 20 switches from the normal ROM character definitions to a custom character set in RAM. The upper six bits of location 65299 tell the computer where the character set is located in memory. When POKEing to location 65299 it is important not to disturb its lower two bits. To determine the number to POKE into this location, divide the starting address of the custom character set by 256. Since the example program puts the character set at location 15360, we use the POKE in line 30.

The program uses a machine language routine to copy the character set from ROM to RAM beginning at location 15360. To copy the character set to some location other than 15360, replace the second number in line 50 (currently 60) with the number you POKEd into line 30. Line 40 reads the custom character data from line 80 and POKEs it into the area reserved for the definition of the @ symbol.

IBM Mazes And Movement

I am trying to write a maze game for the IBM. How could I write the program to randomly generate a simple maze, allowing me to choose the dimensions, and make sure that it is a solvable one?

Aaron Greenberg

Few COMPUTE! *programs have spawned so many offspring as Charles Bond's maze generation algorithm, originally published in the December 1981 issue. The short and simple procedure generates a maze that's different every time the program is run, yet is always solvable. Here's a PC/PCjr adaptation for SCREEN 1, the medium-resolution graphics screen (for the PC, this requires a color/graphics adapter). In place of the PEEKs and POKEs of the earlier versions, this one uses BASIC's SCREEN, LOCATE, and PRINT statements:*

```
OK 100 KEY OFF:SCREEN 1,0:COLOR
       1,0:CLS:RANDOMIZE TIMER
DJ 110 MAXROW=23:MAXCOL=40:DIM P
       (3,1):FOR J=0 TO 3:READ P
       (J,0),P(J,1):NEXT
IE 120 DATA 0,2,-2,0,0,-2,2,0
DM 130 HL=8:YPOS=2:XPOS=2:LOCATE
       YPOS,XPOS:PRINT CHR$(5)
OO 140 J=INT(RND(1)*4):X=J
MP 150 NY=YPOS+P(J,0):NX=XPOS+P(
       J,1):IF NY<1 OR NY>MAXROW
       OR NX<1 OR NX>MAXCOL THE
       N 170
JP 160 IF SCREEN(NY,NX)=0 THEN L
       OCATE NY,NX:PRINT CHR$(J+
       1):LOCATE (YPOS+P(J,0)/2)
       ,(XPOS+P(J,1)/2):PRINT CH
       R$(HL):YPOS=NY:XPOS=NX:GO
```


www.commodore.ca

IT ALSO RUNS ON 64K.

Serious runners know it takes more than great running shoes to improve performance. It takes knowledge. Now Puma gives you both. With the RS Computer Shoe. The first training shoe to combine advanced footwear technology with computer technology.

The RS Computer Shoe has a custom-designed gate array built into its heel. This computer chip records your run, then communicates the results to any Apple IIE, Commodore 64 or 128, or IBM PC computer.

A software program included with the shoe automatically calculates your time, distance and calories expended. Then graphically compares them to past performances and future goals.

The RS Computer Shoe from Puma. We're so out front in technology, we put computers in the backs of our shoes.

Apple is a registered trademark of Apple Computer, Inc.; Commodore 64 and 128 are trademarks of Commodore Computer Systems; IBM and IBM PC are registered trademarks of IBM.

PUMA®
OUR WORD FOR QUALITY

---

kyan

ALL NEW SOFTWARE LINEUP!

***kyan pascal* (Version 2.0) $69.95**

kyan pascal is the ideal system for learning Pascal and developing Pascal programs. It's a full implementation of ISO Pascal and conforms to the standards set by the Federal Software Testing Center. ***kyan pascal*** features a menu-driven environment with multiple HELP screens; a full-screen text editor; and, optimized 6502 machine code compiler/assembler. It produces code that runs at the maximum speed possible on the 6502 microprocessor. ***kyan pascal*** supports many extensions including string handling, linking, chaining, random files, and included or inline assembly source code. It also supports a line of powerful toolkits which make it possible for even novice programmers to develop sophisticated software. ***kyan pascal*** (Version 2.0) requires only one disk drive. It is available for the Apple II (runs in ProDOS and requires 64K); Atari (runs DOS 2.5 and requires 48K); and Commodore 64/128. ***kyan pascal*** is not copy protected and comes with a Pascal tutorial and reference guide.

***Programming Utility Toolkit* $49.95**
Programming is faster and easier with this extensive library of utility programs and file management procedures. The Toolkit includes source code for more than 20 utility programs.

***Advanced Graphics Toolkit* $49.95**
Add stunning graphics to your Pascal or assembly language programs. With the Toolkit's graphics primitives, you can build a custom graphics library. Or, you can use the Toolkit's library for 2 and 3 dimensional transformations, windows and clipping, shading, and more.

MouseText Toolkit *(available for Apple II only)* **.. $49.95**
Add Macintosh-like graphics to your Pascal programs. The Toolkit includes routines for pull-down menus, windows, and mouse-controlled cursor events (Toolkit requires Apple IIc or enhanced IIe).

***Macro Assembler/Linker* $69.95**
kyan's latest programming tool adds a new dimension to assembly language programming. The Assembler/Linker includes a text editor, 65C02 macro assembler, object module linker, debugger, and librarian.

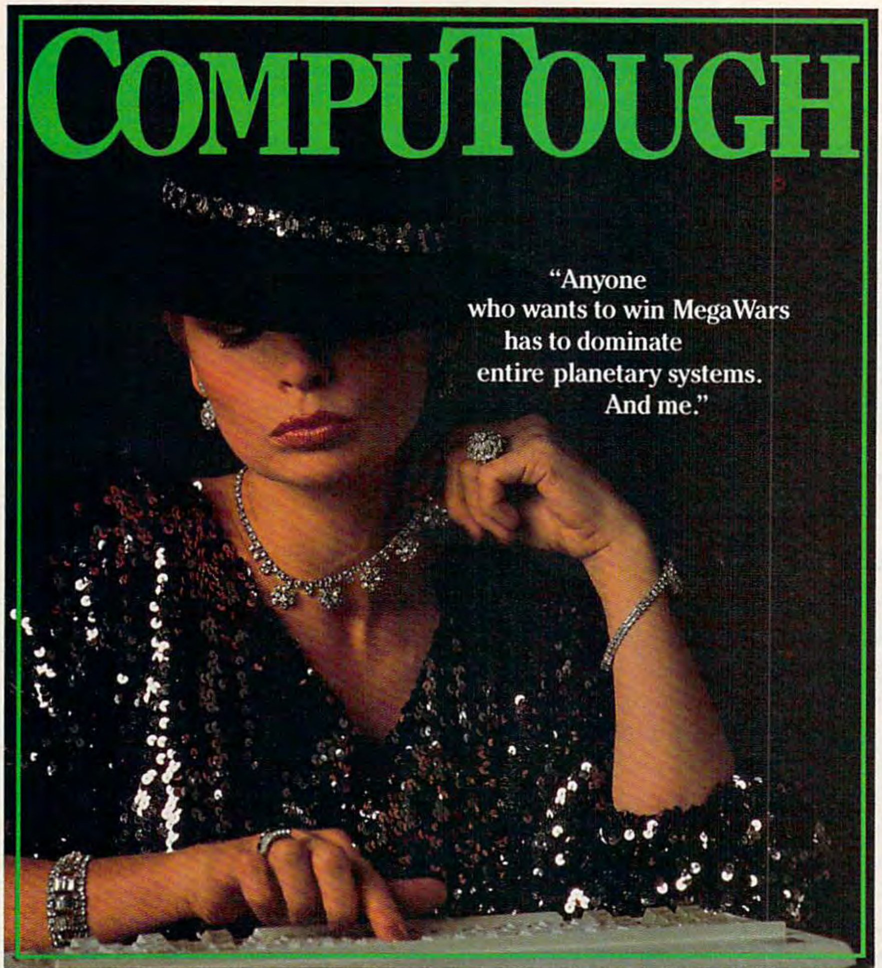
kyan Software offers you a 15 day money back guarantee. See for yourself . . . kyan is the best programming software.

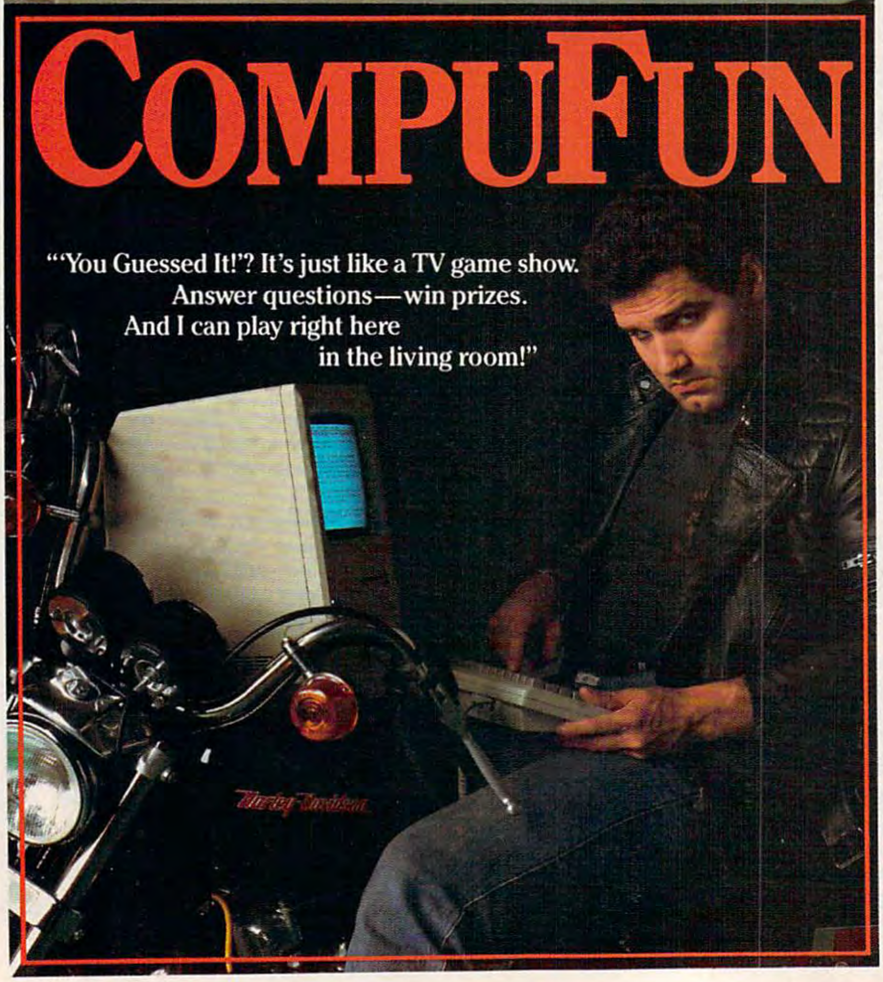
Send Check/Money Order: kyan software, Dept. P • 1850 Union Street, #183 • San Francisco, CA 94123

Or Call: (415) 626-2080 • Visa/MC Accepted

Please include $4.50/order for shipping/ handling; $12 outside North America. CA residents add 6.5% sales tax.

www.commodore.ca






```
        TO 14Ø
DC 17Ø J=(J+1)*-(J<3):IF J<>X TH
        EN 15Ø
JK 18Ø J=SCREEN(YPOS,XPOS)-1:LOC
        ATE YPOS,XPOS:PRINT CHR$(
        HL);:IF J<4 THEN YPOS=YPO
        S-P(J,Ø):XPOS=XPOS-P(J,1)
        :GOTO 15Ø
HB 19Ø GOTO 19Ø
```

To customize the routine for your own use, change MAXROW and MAXCOL (line 110) for the maximum number of rows and columns in the maze. (Don't make MAXROW greater than 23, since printing on the bottom two lines of the screen causes scrolling.) As it stands now, the routine always starts constructing the maze from the upper-left corner. You can change this by changing the values of XPOS and YPOS (line 130). The values should always be at least 2, but less than MAXROW and MAXCOL. The variable HL (line 130) defines the character used for the paths of the maze. You can change this to any character you desire, but its value must be greater than 5 (lower values are used to draw the maze) and less than 128 (higher values are not available on the graphics screen). Unfortunately, this set of characters does not include a reverse space that would draw solid paths for the maze. It's up to you to define which end point is the finish of the maze.

Now that the maze is in place, it's an ideal time to answer a letter from R.C. Loveland, who wants to know how to use the IBM joystick. The joystick is an ideal tool for maneuvering a player through the maze, and BASIC's STICK and STRIG functions make it easy to read. IBM joysticks are "positional"; they return values that reflect the horizontal and vertical deflection of the stick relative to a simple coordinate system. In this system, coordinate 0,0 means the stick is pushed to the upper-left corner, and 255,255 means the stick is pushed to the lower-right corner. STICK(0) returns the horizontal (x) coordinate of the first joystick, while STICK(1) returns the vertical (y) coordinate. STICK(2) and STICK(3) perform the corresponding functions for the second joystick. The only special rule is that STICK(0) must be read first, before any other directions. (Even if you only want positions from the second joystick, you must read STICK(0) first.)

STRIG reads the status of the joystick buttons—most IBM joysticks have two, but only one per joystick can be read unless you're using BASICA. You must use the statement STRIG ON before you can read button status. After enabling button reading, STRIG(0) returns −1 if the primary button on the first joystick has been pressed since the last time STRIG(0) was called, or 0 if it has not been pressed. STRIG(1) is slightly different—it returns −1 if the primary button on the first joystick is currently pressed (regardless of its previous state), or 0 if it is not

www.commodore.ca

pressed. STRIG(2) and STRIG(3) perform the corresponding functions for the primary button on the second joystick.

This system makes it easy to determine the position of the joystick. But in a situation like navigating the maze drawn by the routine above, what you really need to know is the direction in which the stick is pressed. Add the lines below to the maze-drawing routine above:

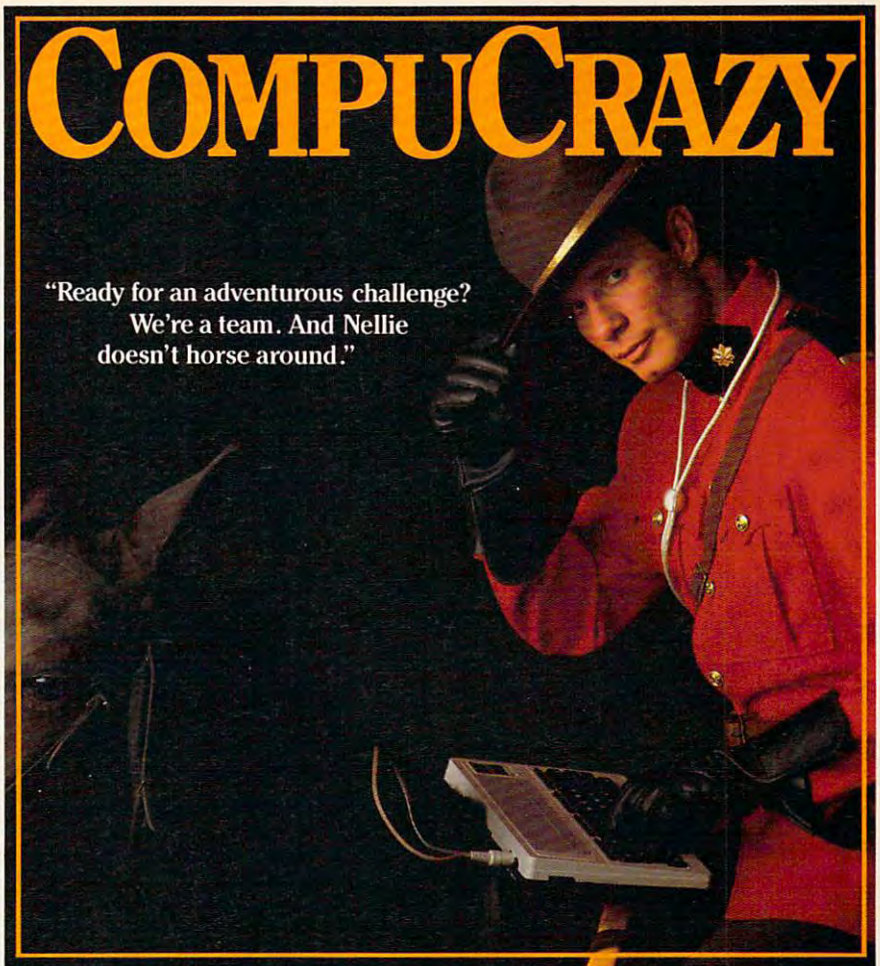
```
LO 190 CH=1:XPOS=2:XPOS=2:LOCATE
        YPOS,XPOS:PRINT CHR$(CH)
NK 200 XMOV=STICK(0)-XCTR:XJOY=S
       GN(XMOV):IF ABS(XMOV)<10
       THEN XJOY=0
JH 210 YMOV=STICK(1)-YCTR:YJOY=S
       GN(YMOV):IF ABS(YMOV)<10
       THEN YJOY=0
OB 220 NY=YPOS+YJOY:NX=XPOS+XJOY
       :IF NY<1 OR NY>23 OR NX<1
        OR NX>40 THEN 200
CO 230 IF SCREEN(NY,NX)=0 THEN 2
       00
JG 240 LOCATE YPOS,XPOS:PRINT CH
       R$(8):LOCATE NY,NX:PRINT
       CHR$(1):YPOS=NY:XPOS=NX:G
       OTO 200
```

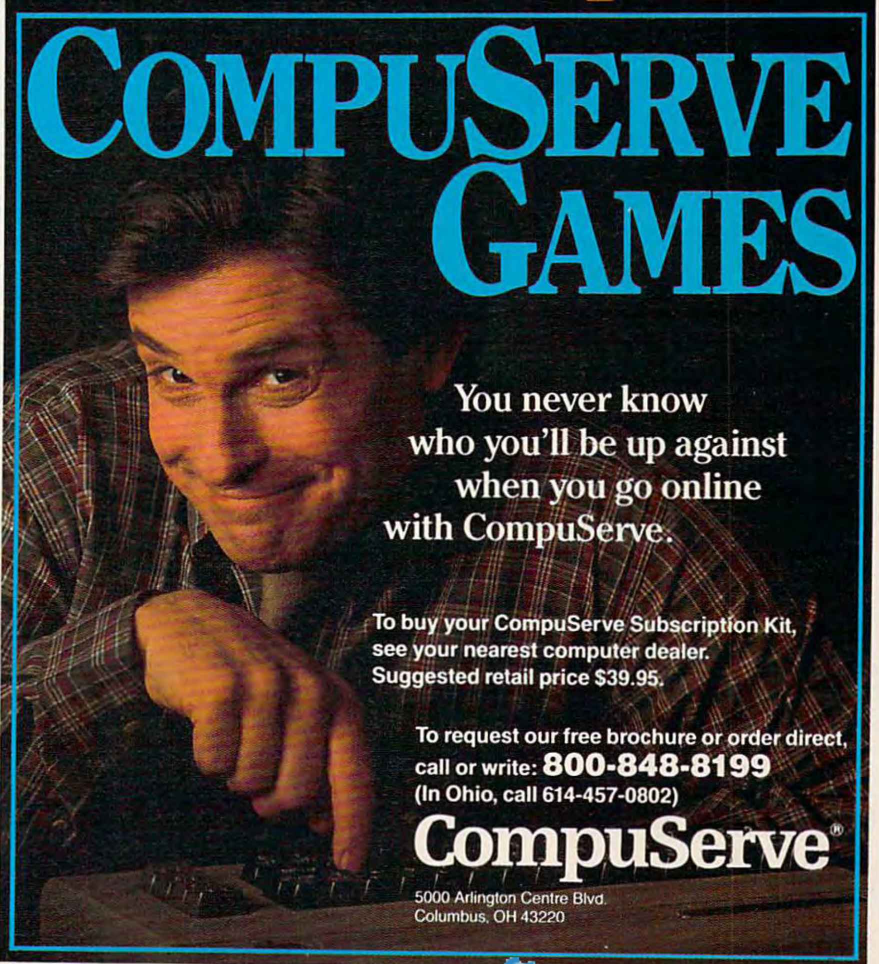
Line 190 defines character 1 (the reverse-smiling face) as the player, then positions it at the start of the maze. Lines 200–210 calculate two directional values, XJOY and YJOY, based on how far the stick is moved from the center positions (XCTR and YCTR). XJOY is −1 if the stick is moved to the left, 1 if the stick is moved to the right, and 0 if the stick is not moved horizontally. YJOY is −1 if the stick is moved up, 1 if the stick is moved down, and 0 if the stick is not moved vertically.

The advantage of this system is that the screen player can be moved very simply in relationship to the joystick by adding the XJOY and YJOY values to the current position and using the LOCATE statement (lines 220 and 240). The sensitivity of the joystick can be adjusted by changing the value in the ABS test (lines 200–210). As shown, the joystick must be moved at least 10 increments in the desired direction for the change to register. This prevents small jiggles of the stick from causing unwanted movement. The test in line 230 prevents the player from leaving the maze. The SCREEN function returns 0 if no character has been printed in a position, while a maze path position will hold the value defined by HL in the maze-drawing routine.

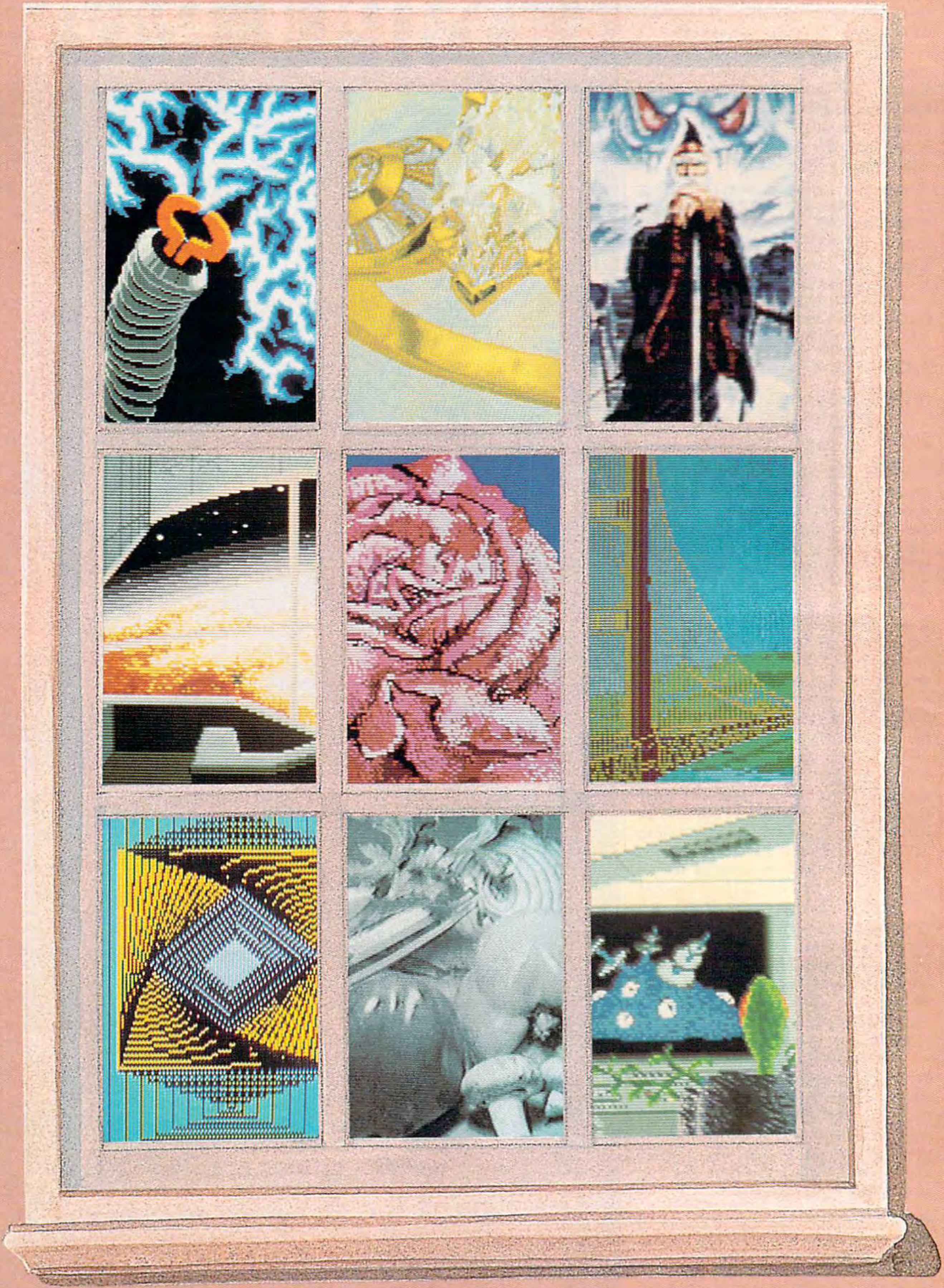
One additional step is required to use this joystick routine. Each joystick returns slightly different readings, so it's difficult to predict what the values for the center coordinates will be. Thus, it's necessary to calibrate the joystick at the start of every program that reads it. The following lines show how this can be done:

```
PC 10 CLS:WIDTH 40:STRIG ON:PRIN
      T "Press fire button to se
      t center position."
PL 20 IF STRIG(0)=0 THEN 20
CO 30 XCTR=STICK(0):YCTR=STICK(1
      )
```
©





www.commodore.ca



www.commodore.ca

THE DIGITAL WINDOW

COMPUTER GRAPHICS TODAY AND TOMORROW

Kathy Yakal
Assistant Features Editor

Maybe you've never thought of yourself as an artist, but maybe you've never had the right tools to experiment easily with shapes and colors. There's a lot of software available—ranging from simple drawing programs to digitizers to sophisticated animation creators—that lets you explore your talent. Some is even used by professional artists and animators.

Before there were words, there were pictures. Human beings have always communicated with art. Although the media have changed over the centuries, the need to express ourselves through pictures has always existed, even if it's just doodling on a legal pad while talking on the telephone.

Computers got into the act about 25 years ago when Ivan Sutherland, a graduate student at the Massachusetts Institute of Technology, programmed a computer to draw a straight line. Since then, computer-generated graphics have evolved rapidly. And they've been used for many business applications as well as for aesthetic purposes.

We're at an interesting stage in the evolution of microcomputer graphics in 1986. Many programs have been developed over the last few years for casual use on home computers. But the new 68000-based machines, such as the Atari ST and Commodore Amiga, offer graphics capabilities previously seen only on mini- and mainframe computers. The little computers are starting to catch up with the big ones.

You probably see more computer-generated graphics every day than you realize, especially if you watch much television. Makers of rock videos use computers extensively to produce bizarre images and special effects. Station promos and title screens for movies, TV shows, and newscasts are commonly created with computers. Weather forecasters use computer equipment not only to track and predict weather conditions, but also to prepare graphic presentations for viewers. Usually, the graphics you see on TV

www.commodore.ca

are created with dedicated work stations—special computers and programs designed specifically for advanced graphics applications.

As the early microcomputer programmers explored applications for their machines, they came up with simple drawing programs. Though many of these programs were crude and difficult to use compared to what's available today, they allowed nonprogrammers to do something they couldn't do before: create images on a computer screen, using the keyboard or a joystick to select options and draw pictures.

In 1984, something happened that pushed the evolution of microcomputer graphics a bit further: Apple introduced the Macintosh. Two elements of this computer's design had a great impact on the next step in computer graphics. First, it used Motorola's 68000 microprocessor, a more sophisticated chip that allows higher-resolution bitmapped screens. Second, its user interface, incorporating pull-down menus and mouse control, made the Mac very easy for the average person to use.

In 1985, the Atari ST and Commodore Amiga hit the market, offering similar ease of use and superior graphics capabilities, including color. And since they cost less, both machines are opening up the computer graphics field to even more software developers and consumers.

Amiga LIVE, *a peripheral that accommodates digitization of real images on the computer screen, was used to create this picture. (Courtesy of Commodore-Amiga.)*

The Amiga's impressive graphics capabilities are well illustrated in this picture created using Aegis Images, *from Aegis Development.*

When you draw and paint a picture using traditional artist's tools, there are several processes required after visualizing what you're going to draw—though the order may vary from person to person. You draw shapes and fill some of them in. You mix different colors of paints to come up with just the right shade. You create backgrounds. And sometimes, you scrap the whole thing and start over.

Many draw-and-paint programs for computers let you do all of those things, but take advantage of the computer's processing powers to help with some of the detail work. Many of them use similar terms to describe these features, and here are some of the most common:

DRAW You can choose the width of stroke you want to use and draw your own shapes freehand.

FILL You can select a color or predefined pattern (or create your own pattern) and paint the inside of a hollow shape or the background. This doesn't require painting the area with the input device as you would with a paintbrush; you just indicate the area you want filled and press a button or key. It fills almost instantly.

ZOOM or **MAGNIFY** This lets you zoom in on one tiny area to draw in detail, pixel by pixel. With some programs, you can see the magnified area and the overall picture simultaneously, or at least by flipping screens.

CIRCLE or **BOX** By defining two points (centerpoint and radius for a circle or two opposite corners of a box), you can instantly create a shape between those points. Many programs have a variety of shapes to choose from. Before you actually set the second point, most programs let you preview the shape and adjust its size. This is called a "rubber band" effect because of the way the shape stretches on the screen.

www.commodore.ca

Introducing high tech magic!

From the minds of MASTERVOICE. Introducing Butler - In - A - Box. The world's first environmental control system that *responds to voice commands.* The first with A.I.R. (Artificial Intelligent Recognizer); a futuristic software program which makes it a reality.

Replace pushbutton control with voice activated magic!

Butler-In-A-Box replaces old-fashioned pushbutton control systems, making them obsolete. At the sound of your voice, he carries out your tasks. All you have to do is ASK! From the comfort of your favorite easy chair, up to twenty feet away.

Control all of your electronic devices at the sound of your voice.

Butler-In-A-Box puts all of your electronic devices, high tech or not, under your voice control. He will dial your phone and answer incoming calls without ever touching him. He turns your TV, stereo, heating systems and other electronic devices on and off, even dims lights. All of this instantly or at the predetermined times you desire.

Computerized protection against intruders.

Butler-In-A-Box has a unique, built-in infrared sensor and *intrusion detection* system that guards your home and alerts you to uninvited guests. When he detects intruders, he will *speak,* and *ask* them to identify themselves. Only you can verbally command him to turn off his intrusion detection system, because he is trained to recognize only your voice. He is also capable of interfacing with your existing home security system, so it can be activated by your voice.

Speaks and understands any language.

Your Butler-In-A-Box is smart enough to call you by name and answer "intelligently" with a variety of random responses — in any language you wish! Even with an *accent.*

Easy to install and use.

Your Butler has been designed with you in mind. He's so simple to install and use, you won't believe it. Complete with instructional audio cassette and easy-to-follow written instructions. And, *no special wiring* of your home is required.

Experience the technology of tomorrow, today! Put Butler-In-A-Box to work for you. Only $1,195.

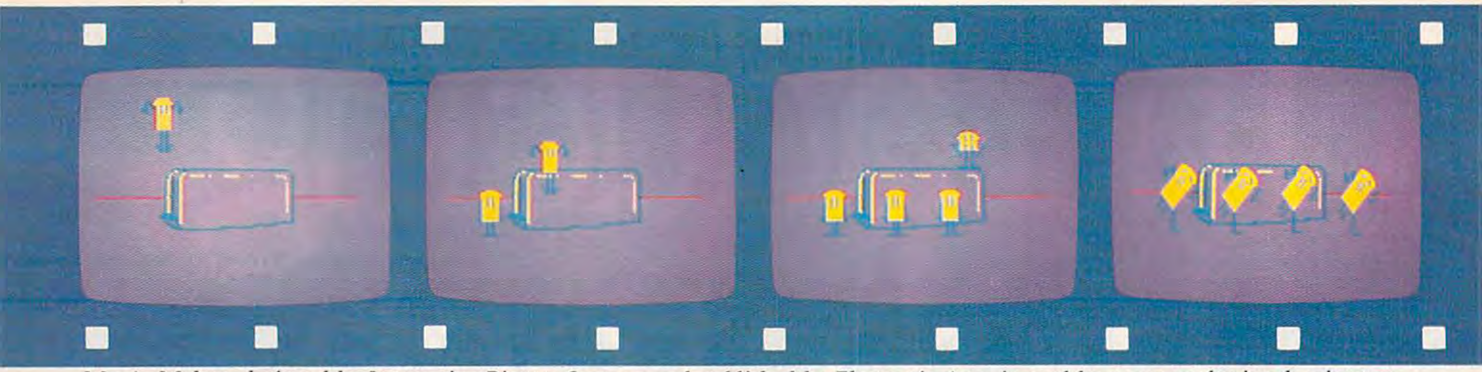
Order direct or send for our free brochure by just lifting a finger (possibly for the last time), and dialing our toll-free hot-line: 1-800-O-BUTLER. (In California) 1-(714) 952-7056. Or write:

Future Systems Marketing
5067 Cumberland Drive
Cypress, CA 90630

MASTERVOICE
BUTLER IN A BOX

Future Systems Marketing — Tomorrow's dreams, today's reality.

© 1986 Mastervoice, Ltd.



Movie Maker, *designed by Interactive Picture Systems and published by Electronic Arts, is used by many professional animators. With some practice, even the casual user can create sequences like the one shown above. These four frames were excerpted from a minute-long cartoon of four pieces of toast popping out of a toaster, performing a short dance, taking their bows, and popping back in.*

FLIP Using this option, you can invert or otherwise reorient a shape so it sits at a different angle.

SPRAY PAINT or **AIR BRUSH** This simulates what an airbrush does, something like fine-tuned spray-painting. Most programs let you select from a variety of widths, densities, and colors.

MIRROR IMAGE This feature lets you reverse an image you've already created, producing a mirror-image effect. With some programs, you can draw while this mode is active and see a reversed reflection of what you're drawing on the other half of the screen.

ERASE or **UNDO** There are usually two separate commands for deleting mistakes. One erases only the last thing you've done, and the other erases the entire picture. Also, you can use the input device as an eraser, moving around and deleting individual areas by redrawing them in the background color.

TEXT Some programs let you design your own character sets and integrate text with illustrations.

These are just some of the most common features found in drawing programs; each has its own additional options. The basic look of these programs is fairly similar, though. Lately, many of them have imitated *MacPaint* by displaying option icons around the perimeter of the screen. You select the icon representing the function you want, move onto the drawing surface, and begin using the function. At any time you can move off the drawing surface and choose another icon. Pull-down or drop-down menus are becoming popular, too. Input devices range from the keyboard to joysticks, light pens, graphics tablets, or mouse controllers.

When you're done with a picture, you can save it on disk or tape, print it out in black and white or color (most programs are compatible with popular printers), exchange it with other enthusiasts, hang it on a wall, or use it as an illustration in a newsletter or other publication. (Many of the figures appearing in COMPUTE! and COMPUTE! books have been created or conceptualized on a Macintosh.) Programmers can use the finished drawings as backgrounds or title screens. For instance, the drawing program that comes with an Atari ST system, *Neochrome*, has

Aegis Images *was also used in designing this picture. Aegis Development recently began shipping* Aegis Animator, *an animation package that includes the Images program.*

been used by at least one commercial software company to produce impressive effects in a graphics adventure game.

Though many drawing programs developed for the newer 16-bit computers are aimed at the commercial market (see "Creating with CAD: Computer-Aided Design" in this issue), there are several programs that are inexpensive and easy to use, ideal for people who like to experiment without investing a lot of money. That's who Tom Hudson had in mind when he designed *Degas*.

Hudson had seen some demo programs for the Atari ST last June and was impressed with the machine's graphics capabilities. More of a programmer than an artist, Hudson sought out the advice of several artists and started writing his own drawing program.

The result is *Degas*, published by Batteries Included. *Degas* offers a palette of 500 colors and works in any of the Atari ST's three screen modes. The number of colors that can be displayed simultaneously depends on the mode—from 16 colors in low resolution (320 × 200) to black-and-white only in high resolution (640 × 400). In addition to these colors, the program also offers a selection of 60 patterns. Unlike many drawing programs, the actual working area is separate from the menu screen; you click the ST's mouse to move back and forth. *Degas* retails for $39.95.

One of the best drawing programs for Apple computers is Brøderbund's *Dazzle Draw*. This package for the Apple IIc or 128K Apple IIe uses a Macintosh-like user interface, multiple windows, cut-and-paste editing, and help screens. It allows ten shapes in 16 colors and 30 patterns in any one picture, and is compatible with a mouse, graphics tablet, KoalaPad, or joystick. *Dazzle Draw* also supports Apple's new UniDisk 3.5, which expands the program's slide show capacity to more than 40 images. It retails for $59.95.

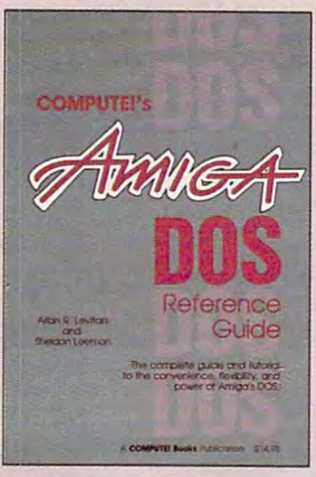
Graphics programs were among the first software packages introduced for the Commodore Amiga, and with good reason. It's a powerful graphics machine, capable of producing almost TV-quality

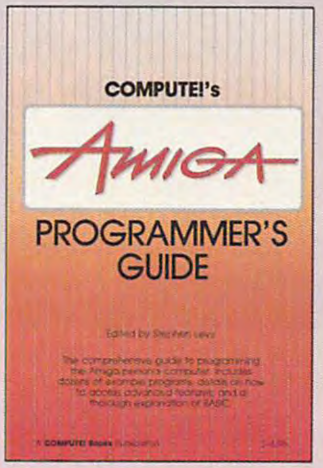
www.commodore.ca

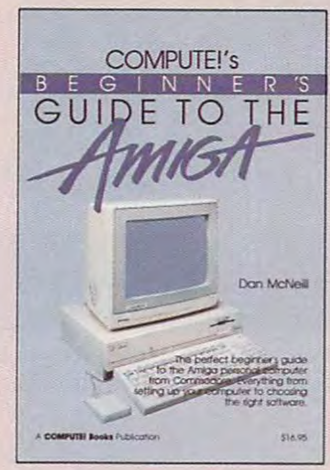
AMIGA

SUPPORT FROM COMPUTE! BOOKS

Everything for the Amiga. From BASIC beginner's guides to advanced programming handbooks, COMPUTE! offers you information-packed tutorials, reference guides, programming examples, ready-to-enter applications, and games to help you develop your computing skills on Commodore's Amiga.







COMPUTE!'s AmigaDOS
Reference Guide

Arlan R. Levitan and Sheldon Leemon

A comprehensive tutorial and reference guide to the powerful AmigaDOS—the operating system underlying the Workbench and Intuition—this book offers information useful to every Amiga owner. It defines and illustrates all DOS commands, and shows you how to create file directories, access peripherals, run batch file programs, and avoid "disk shuffle." The screen- and line-oriented text editors are explained in detail. Numerous examples and techniques explain how to use AmigaDOS to make operating your Amiga both convenient and efficient.

$14.95 ISBN 0-87455-047-5

Elementary Amiga BASIC

C. Regena

Here's your introduction to the new and powerful BASIC on the Amiga personal computer. The Amiga's impressive graphics, animation, and sound can be unlocked with the right commands, and BASIC is the place to start. Complete descriptions of Amiga BASIC's commands, syntax, and organization take you from the beginner level to a full-fledged programmer. Plus, the book offers you ready-to-type-in programs and subroutines while showing you how to write your own programs. *There is a disk available which includes the programs in the book,* $12.95. *This title is also available as a book/disk combination for $29.95* (057-2).

$14.95 ISBN 0-87455-041-6

COMPUTE!'s Amiga
Programmer's Guide

Edited

Your tutorial and reference manual to AmigaDOS, BASIC, Intuition, and other important software tools which accompany the new Amiga, *COMPUTE!'s Amiga Programmer's Guide* is a clear and thorough guide to the inner workings of this fascinating new-generation computer. The great speed of its 68000 microprocessor, coupled with the versatility of the Amiga-specific graphics and sound, makes the Amiga one of the most powerful computers available today.
This book is the key to accessing the Amiga's speed and power.

$16.95 ISBN 0-87455-028-9

Advanced Amiga BASIC

Tom R. Halfhill and Charles Brannon

This guide to applications programming on Commodore's new Amiga contains everything an intermediate programmer requires to begin creating sophisticated software on this powerful machine, including several ready-to-type-in programs. Clear, yet comprehensive documentation and examples cover advanced BASIC commands, designing graphic applications, generating sound and music, using the Amiga's built-in speech synthesizer, creating a user interface, and programming the computer's peripherals. *There is a disk available which includes the programs in the book,* $15.95. (June release)

$16.95 ISBN 0-87455-045-9

COMPUTE!'s Beginners Guide to the Amiga

Dan McNeill

Written in a lively and entertaining style, this book teaches you everything a beginner needs to know to get started quickly with the Amiga from Commodore. You will learn about setting up the system, all the most popular types of software, and details about the hardware.

$16.95 ISBN 0-87455-025-4

Inside Amiga Graphics

Sheldon Leemon

The Amiga, Commodore's powerful new computer, is an extraordinarily impressive graphics machine. Easy to use, the Amiga can produce color graphics and excellent animation. You'll find thorough descriptions of the computer's abilities and the hardware required to create a complete graphics system. Software, too, is central to the Amiga's power, and complete tutorials show you how to get the most from the machine. (June release)

$16.95 ISBN 0-87455-040-8

COMPUTE!'s Kids and the Amiga

Edward H. Carlson

The latest in this bestselling series written by Edward Carlson, *COMPUTE!'s Kids and the Amiga,* will acquaint you with BASIC. Over 30 sections—all with instructor notes, lessons, assignments, and lively illustrations—entertain and amuse you as you learn to program your new computer. Clear writing and concise examples make it easy for anyone—children and adults alike—to painlessly learn BASIC. (May release)

$14.95 ISBN 0-87455-048-3

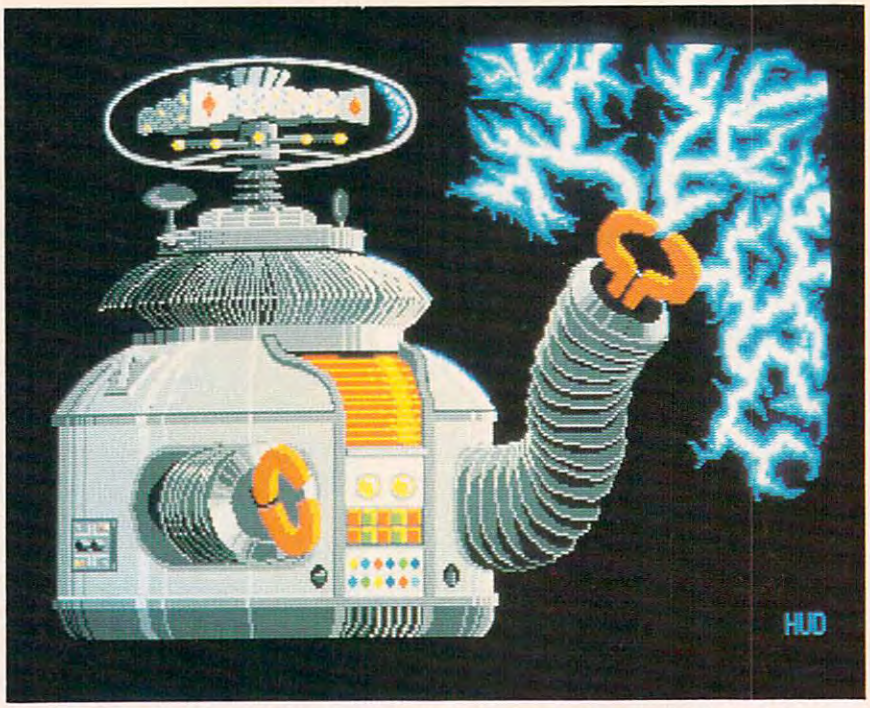
Look for these books at your local book or computer store.
Or order directly from COMPUTE!.
Call toll-free 1-800-346-6767 (in NY 212-887-8525).

Please allow 4–6 weeks for delivery after your order is received.

COMPUTE! Publications, Inc. abc
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor, New York, NY 10019
Publishers of COMPUTE!, COMPUTE!'s Gazette, COMPUTE!'s Gazette Disk, COMPUTE! Books, and COMPUTE!'s Apple Applications

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, England, and in Canada from McGraw-Hill, Ryerson Ltd., 330 Progress Ave., Scarborough, Ontario, Canada M1P 2Z5.

www.commodore.ca



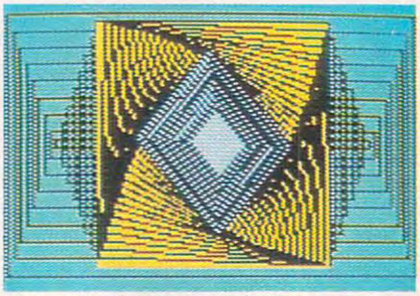
Tom Hudson, designer of Batteries Included's Degas *for the Atari ST, used the product to create this picture.*

pictures. Electronic Arts has released *Deluxe Paint* for the Amiga, a very sophisticated paint program. Beyond the usual features, *Deluxe Paint* has a special palette window that lets you easily mix any combination of the Amiga's 4,096 colors. With *color cycling*, you can make parts of the picture appear to move, like a waterfall flowing or Christmas lights twinkling on a tree. Brush size can vary from a single pixel to a full screen, and any piece of a picture can be picked up and used as a brush. *Deluxe Paint* retails for $99.95.

The next step, logically, is software which can take a static screen picture and add animation. This is an extremely complex procedure that taxes the power of even the fastest of today's personal computers. But even though the kind of animation seen in Saturday morning cartoons is still a few years away, you might be surprised at what can be done already.

Fantavision, from Brøderbund, is so easy to use that you can turn out an animated sequence in a few minutes. It incorporates the same kind of user interface adopted by many drawing programs—you pick up icons on the border of the screen, move to the drawing surface and work, then move back to the border to change functions. When you've finished each frame, you save it, then move on to the next. When you're done, you execute a command that runs the sequence.

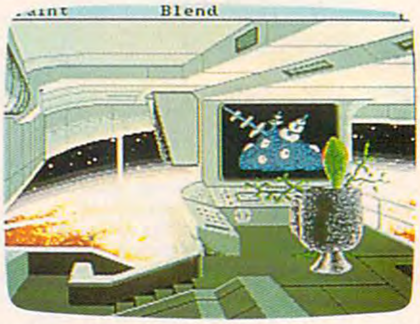
Fantavision employs an unusual animation technique. In the past, conventional animation has required artists to draw each frame of a sequence when there is the slightest change. It takes thousands of drawings for even a short piece of animation. But *Fantavision* uses *tweening*, a process that automatically fills in the transitional frames, making the sequence run smoothly. In effect, you tell the program where to start and end, and it fills in the middle. *Fantavision*, available for Apple II-series computers with at least 64K, retails for $49.95.



This still shot from Brøderbund's Fantavision *gives you an idea of what the program's palette allows you to do. Its real power, though, lies in its incredibly easy-to-use animation features.*

Movie Maker, designed by Interactive Picture Systems and published by Electronic Arts, is a bit more complicated to use, but the payoff is more sophisticated animation. So sophisticated, in fact, that some professional animators use it.

Using *Movie Maker*, you can produce an animated sequence composed of up to 300 frames, with up to six video tracks and three sound tracks. The program consists of four main areas. In *Compose*, you create files of characters and backgrounds. *Record* asks you to recall those files and order them in the sequence you want, adding sound where appropriate. *Smooth* automatically rounds off the rough



Deluxe Paint, *from Electronic Arts, was one of the first products shipped for the Commodore Amiga. This scene, created with the product, illustrates the sophisticated shapes and shading possible.*

Commodore 64 ($34.95), Atari ($32.95), and Apple ($39.95).

The Graphics Magician, designed by Polarware and published by Penguin Software, is actually two programs. *The Graphics Magician Painter* lets you draw pictures and save them in a very compact format on disk; *The Graphics Magician Animator* animates them. It's $59.95 for the Apple II series and $79.95 for Macintosh.

At this point, the way to create the most lifelike graphics on personal computers is to make digital images of real objects. *Computereyes*, from Digital Vision, Inc., lets you feed an image from a

www.commodore.ca

videocassette recorder, video camera, TV, or any other video source into a computer and digitize it on the screen. The image can then be incorporated in a game or dumped to a printer. (An upgraded version of *Computereyes* that makes it compatible with both *The Newsroom* and *The Print Shop* can be purchased for $15.) The *Computereyes* software is sold both separately and as part of a package along with a black and white video camera. *Computereyes* is available for the Commodore 64, Apple, and Atari ($129.95; $399.95 with video camera) and IBM PC family ($249.95; $519.95 with camera).

Two peripherals which have been announced for the Amiga allow video mixing and digitizing. The Genlock mixes external video signals with the computer's own video, and the Amiga Live digitizer captures external video images in color and in realtime. Hippopotamus Software has announced a black and white digitizer for the Atari ST, with plans for an Amiga version later.

Powerful personal computer software is closing the gap between what an artist can do with state-of-the-art drawing tools and what a photographer can do with a camera. This ring was created using Amiga's GraphiCraft *program. (Courtesy of Commodore-Amiga and artist Jack Haeger.)*

edges, and *Play* runs the movie. *Movie Maker* is available for the

As technology advances, a debate rages over two issues related to computer-generated art. First, will these sophisticated tools mean that anyone can be an artist? Some people feel they could compete with traditionally trained artists if they had the right tools. They feel that they lack only the mechanical—not the *artistic*—abilities.

Second, can a computer ever be creative in the same way human beings are? Will it ever be the artist, and not just an artists's tool? No one knows yet, but if the evolution of computer graphics continues at the pace it's been going, it may not be too many years before we have some answers.

Pawn, *a new graphics and text adventure from Firebird Licensees, Inc., contains numerous sharp images like this. The program's graphics were developed using* Neochrome, *a graphics package included in the Atari 520 ST's development system.*

For More Information

To learn more about any of the products mentioned here, contact:

Batteries Included
30 Mural Street
Richmond Hill, Ontario
Canada L4B 1B5

Brøderbund Software
17 Paul Drive
San Rafael, CA 94903

Digital Vision
14 Oak Street
Suite 2
Needham, MA 02192

Electronic Arts
2755 Campus Drive
San Mateo, CA 94403

Firebird
P.O. Box 49
Ramsey, NJ 07446

Penguin Software
830 Fourth Avenue
P.O. Box 311
Geneva, IL 60134

©

www.commodore.ca

Creating With CAD:

COMPUTER-AIDED DESIGN

Selby Bateman, Features Editor

From a simple floor plan to the most complex electronic circuitry, computers are changing the worlds of design and engineering in virtually every field. Formerly the sole province of mainframe computers and expensive graphics terminals, sophisticated computer-aided design (CAD) software is now available for a wide variety of personal computers. Even for casual users, CAD programs are practical design tools that can also be entertaining and educational.

In an automobile factory, a design engineer puts a new axle through a series of stress and endurance tests. In an aircraft plant, another engineer studies how a breakthrough in fuselage design improves the agility of a jet fighter. And in the civil engineering department of a major city, a highway planner examines how a new thoroughfare changes the urban landscape.

Yet, all of these projects are merely sketched in phosphor on computer screens. The axle, the fuselage, and the highway have never been constructed. But the computer-aided designs are so accurate that they won't have to be built until they've been thoroughly analyzed and tested.

Each year, engineers and designers save millions of dollars in time and effort by modeling new projects with a computer before actually manufacturing them. In high-risk endeavors, such as aircraft design, lives can be saved by discovering design bugs on a computer monitor rather than watching them appear at 30,000 feet. Whether today's designers are creating new shoes or rockets, they're finding that computers dramatically change the way they work.

Up to recently, however, these complex CAD projects could be accomplished only with expensive, powerful mainframe computers and dedicated graphics work stations. Microcomputers simply lacked the memory, screen resolution, and sophisticated software to let them be-

www.commodore.ca

BATTERIES INCLUDED

"The Energized Software Company!"

We started with Commodore, designing programs that quickly became industry success stories. Now we're moving on, applying our expertise to other systems.
*Look for **Apple**, **Atari**, **IBM**, and **Commodore** software with the Batteries Included label!*

Look forward to excellence, in every respect. Power and performance, ease of use, incredibly low prices, for programs that help you in so many ways.

NEW! 130 XE VERSION

"The best Atari word-processor ever."
ANTIC MAGAZINE

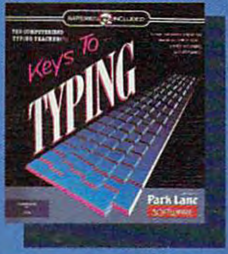
NEW! C128 VERSION

"...capable of very large and complicated searches ...a very good system."
TPUG MAGAZINE

NEW! C128 VERSION

"Performance: excellent... Error-Handling: excellent... Value: excellent"
FAMILY COMPUTING
(a "Billboard" magazine #1 best seller)

NEW! C128 VERSION



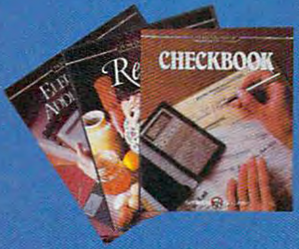
"quite simply the best ...the highest rating possible."
ANALOG COMPUTING

B.I.-80 Column Display: add-on module doubles your screen capacity and improves visibility.

Also from BATTERIES INCLUDED:

Calkit: problem-solving spreadsheet program with built-in templates for the most-needed home and business applications — including income tax, budgets and many more.

B/Graph: professional-quality graphics/charting and statistical analysis package turns your data into superb visuals.



HOME ORGANIZER SERIES home database managers. (8 programs including Home Inventory, Recipes, CheckBook, Address Book, Audio/Video Catalogue and more)

NEW! FOR 1986
PAPERCLIP ELITE   I*S TALK   ISGUR PORTOLIO SYSTEM.
FOR THE AMIGA, ATARI ST and IBM/MS DOS SYSTEMS.

BATTERIES INCLUDED

"The Energized Software Company!"

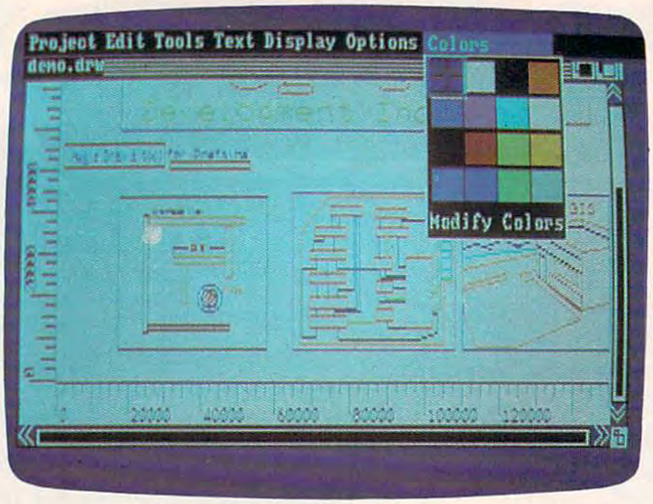
30 Mural Street
Richmond Hill, Ontario
L4B 1B5 CANADA
(416) 881-9941
Telex: 06-21-8290

17875 Sky Park North, Suite P
Irvine, California
USA 92714
(416) 881-9816
Telex: 509-139

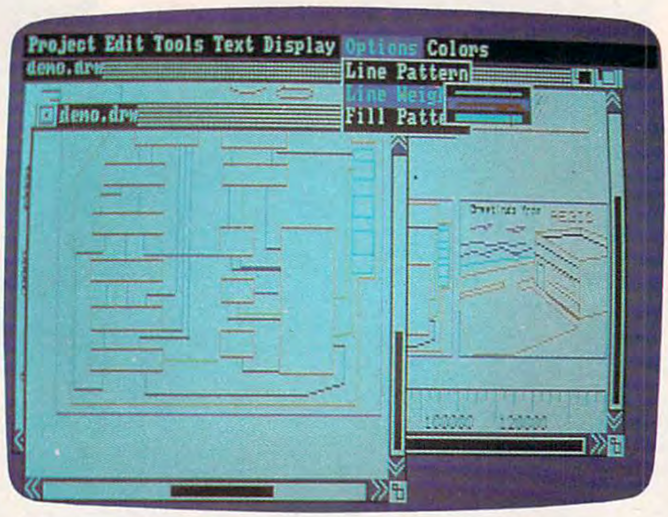
WRITE TO US FOR FULL COLOUR CATALOGUE of our products for COMMODORE, ATARI, APPLE and IBM SYSTEMS.
FOR TECHNICAL SUPPORT OR PRODUCT INFORMATION PLEASE PHONE (416) 881-9816.
SOME PROGRAMS ARE NOT AVAILABLE FOR ALL SYSTEMS.

Commodore, Apple, Atari and IBM PC are registered trademarks of Apple Computers, Inc., Atari, Inc., Commodore Business Machines, Inc., and International Business Machines, respectively

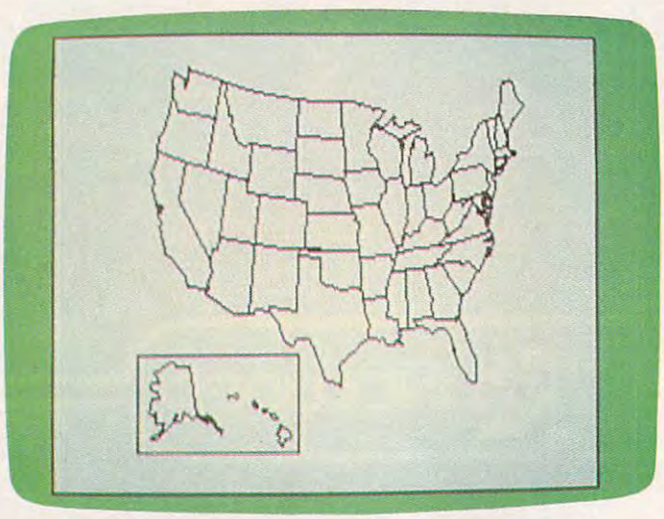
www.commodore.ca



A computer-aided design created with Aegis Draw *on the Amiga.*



Aegis Draw *uses the Amiga's multitasking capability to permit more than one onscreen window to be used actively in a CAD project.*



A design created on the Commodore 64 with CADPAK *(Enhanced Version) from Abacus Software.*

gin to approach such tasks.

But that's changing, thanks to computers like the Apple Macintosh, Atari ST, and Commodore Amiga. Already there are numerous professional CAD programs for the IBM PC family, though the PCs may have a difficult time keeping up with the CAD capabilities of the newer computers in the future. The Amiga in particular, with its versatile graphics and built-in multitasking, offers considerable power to CAD users. Simpler CAD work can even be done on some eight-bit computers with good graphics systems, such as the Commodore 64 and Atari 400/800/XL/XE series.

The variety of CAD projects is virtually limitless—flow charts, family trees, building designs, neighborhood maps, floor plans, architectural drawings, circuit designs, topographical maps, landscaping plans. All of these tasks, and many more, are being accomplished every day with microcomputers.

Advanced CAD programs for personal computers share many basic concepts, though the execution and ease of use varies from program to program. Whether these programs use the keyboard, a light pen, mouse, or some other input device, they reduce the tedious and repetitive work that has traditionally impeded the design process.

A few common features of CAD programs include:

• **Libraries of predrawn images:** When you need to use images specific to a particular type of design—whether an office building, a landscaped yard, a plumbing layout, or a circuit board—why waste time creating them from scratch? Templates and libraries of predrawn images can offer everything from a door or window to a steel I-beam, circuit gate, or tree.

• **Zoom and scroll commands:** Create a design larger than your computer screen, then scroll anywhere over that image. Zoom in on the smallest aspect of any design for detailed work, and then see instantly how it affects the work as a whole.

• **Drawing commands:** Unlike many computer paint programs,

www.commodore.ca

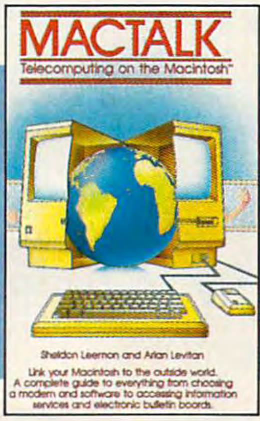
TELECOMPUTING

It's only a phone call away.

MacTalk: Telecomputing on the Macintosh
Sheldon Leemon
Arlan Levitan

A complete guide to telecomputing on the Macintosh from choosing a modem and software to accessing information services and electronic bulletin boards.

$14.95 ISBN 0-942386-85-X



COMPUTE!'s Telecomputing on the IBM
Arlan R. Levitan
Sheldon Leemon

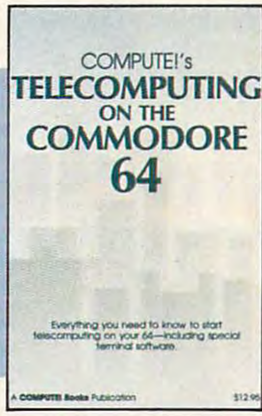
The ins and outs of telecomputing on the IBM PC or PCjr, selecting a modem and evaluating terminal software, how to go online with the major information services.

$14.95 ISBN 0-942386-96-5

COMPUTE!'s Telecomputing on the Commodore 64
Edited

Introduces readers to telecommunications, with sections on buying and using modems, accessing information services and bulletin boards, and uploading and downloading files. There is also a disk available which includes the programs in the book.

$12.95 ISBN 0-87455-009-2



Telecomputing lets you call up computers around the world through a network of telephone lines.

To get you started in telecomputing, COMPUTE! Books offers you five top-selling books. Written for the Apple II-series, Commodore 64, IBM PC and PCjr, and Macintosh, the books give you all the information you need, from selecting software to dialing large databases.

To order your complete guide to telecomputing, give us a call. In the U.S., call toll free 1-800-346-6767 (in NY call 212-887-8525).

COMPUTE!'s Personal Telecomputing
Don Stoner

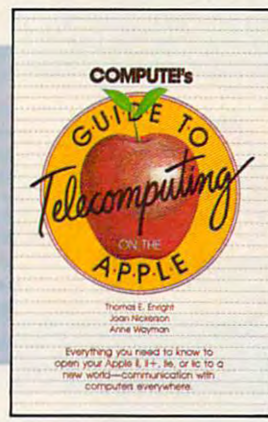
This comprehensive general guide to the world of telecomputing shows how to access databases, receive software, and communicate with others using a personal computer.

$12.95 ISBN 0-942386-47-7

COMPUTE!'s Guide to Telecomputing on the Apple
Thomas E. Enright
Joan Nickerson
Anne Wayman

An informative, easy-to-understand guide to telecomputing on the Apple: covers everything from selecting hardware and software to accessing large databases.

$9.95 ISBN 0-942386-98-1



COMPUTE! Publications, Inc. abc
One of the ABC Publishing Companies

Publishers of COMPUTE!, COMPUTE!'s Gazette, COMPUTE!'s Gazette Disk, COMPUTE! Books, and COMPUTE!'s Apple Applications

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, England.

www.commodore.ca

Lyco Computer Marketing & Consultants

ATARI **apple** **COMMODORE**

ATARI
130XE ... CALL
65XE ... CALL
800XL ... CALL
520ST ... CALL
1050 Drive ... 145
1027 Printer ... 145
850 Interface ... 109
SF314 Drive ... 229
Sf354 Drive ... 179
Sm804 Printer ... 199
Atari Writer ROM ... 33
Atari Writer PLUS ... 36

ATARI
VIP Professional ... 74.50
ST Companion ... 15.95

ATARI SOFTWARE (NEW)
Codewriter ... 35.75
Filewriter ... 20.75
Reportwriter ... 20.75
Menuwriter ... 20.75
Home Integrator ... 19.75
Small Bus. Inventory ... 11.75
Salesman Expenses ... 11.75
Accs Rec/Pay ... 11.75
Retail Invoice ... 11.75
Final Legacy ... 15.75
Adventure Writer ... 18.75

INFOCOM (520ST)
Wishbringer ... 28.95
Zork I ... 28.95
Zork II ... 31.95
Zork III ... 31.95
Deadline ... 33.95
Starcross ... 33.95
Suspect ... 31.95
Suspended ... 33.95
Cutthroats ... 28.95
Sea Stalker ... 28.95
Infidel ... 31.95
Sorcerer ... 31.95
Enchanter ... 28.95
Witness ... 28.95
Planetfall ... 28.95
SpellBreaker ... 33.95

SPINNAKER
Delta Drawing Room ... 19.95
Cosmic Life Room ... 19.95
Up for Grabs Room ... 19.95

PERSONAL PERIPHERALS
Super Sketch C-64 ... 29.95
Printer Utility C-64 ... 14.99
Super Sketch Atari ... 25.95
Super Sketch Apple/Ibm ... 55.00

SUB LOGIC (Atari)
Flight Simulator II ... 32.75
Night Mission Pinball ... 20.75

BRODERBUND (Atari)
The Print Shop ... 28.95
Graphics Library ... 18.95
Graphics Library II ... 19.50
Graphics Library III ... 19.50
Bank St. Writer ... 42.75
Whistler's Brother ... 18.95
Spelunker ... 18.95
Stealth ... 18.95
Serpent's Star ... 24.95
Mask of the Sun ... 24.95

ACTIVISION (ATARI)
Hacker ... 15.75
MindShadow ... 15.75
Ghostbusters ... 15.75
Great Am. Race ... 15.75

MICROPROSE (Atari)
Kennedy Approach ... 21.75
Crusade in Europe ... 24.75
Decision in Desert ... 24.75
Solo Flight ... 20.75
Nato Commander ... 20.75
Spitfire Ace ... 18.75
F-15 Strike Eagle ... 20.75

ACTIVISION (520ST)
Borrowed Time ... 29.75
Hacker ... 26.75

VIP TECH
VIP Professional 520ST ... 115
VIP LITE 520St ... 65
VIP Professional Amiga ... 129

HABA
10 Megabyte ... CALL
Hippo C Compiler ... 49.95
Check Minder ... 49.95
Business Letters ... 36.95
Wills ... 36.95

ACTIVISION (APPLE)
Alter Ego ... 28.75
Ghostbusters ... 24.75
Little People ... 24.75
Space Shuttle ... 24.75
MindShadow ... 24.75

SCARBOROUGH (Apple)
NET WORTH ... 48.75
Improved Mastertype ... 22.75
Mastertype Filer ... 22.75

BRODERBUND (Apple)
The Print Shop ... 31.50
Graphics Library ... 18.50
Graphics Library II ... 18.95
Paper Refill ... 11.95
Bank St Writer 64K ... 41.50
Bank St Writer 128K ... 42.75
Bank St Filer ... 42.75
Bank St Mailer ... 42.75
Bank St Speller ... 42.75
Carmen Sandiego ... 22.75
Karateka ... 22.75
Captain Goodnight ... 22.75
Muppey Cruise ... 25.75

SUB LOGIC (Apple)
Flight Simulator II ... 32.75
Night Mission Pinball ... 20.75

MICROPROSE (Apple)
Crusade in Europe ... 25.75
Decision in Desert ... 25.75
F-15 Strike Eagle ... 21.75
Nato Commander ... 21.75
AcroJet ... 21.75
Silent Service ... 21.75
Gunship ... 21.75

INNOVATIVE CONCEPTS
Flip-n-File 10 ... 3.50
Flip-n-File 15 ... 8.25

COMMODORE
MPS1000 Printer ... 259
C1350 Mouse ... 42
C1700 128K RAM ... 145
C1750 512K RAM ... 269
JANE ... 35
Perfect Writer ... 49
Perfect Calc ... 49
Perfect Filer ... 49

ACTIVISION (AMIGA)
Hacker ... 26.75
MindShadow ... 26.75
Borrowed Time ... 26.75

ACTIVISION (C-64/128)
Alter Ego ... 28.75
Hacker ... 18.75
Little People ... 20.75
Gamemaker ... 24.75
Borrowed Time ... 18.75
Space Shuttle ... 18.75
MindShadow ... 18.75

MICROPROSE (C-64)
Kennedy Approach ... 21.75
Crusade in Europe ... 24.75
Decision in Desert ... 24.75
Solo Flight ... 20.75
Nato Commander ... 20.75
Spitfire Ace ... 18.75
F-15 Strike Eagle ... 20.75
Hellcat Ace ... 18.75

SUB LOGIC (C-64)
Flight Simulator II ... 32.75
Night Mission Pinball ... 20.75

COMTEL
Enhancer 2000 (c-64) ... 179

XETEC
Super Graphix 64 ... 64
Super Graphix JR 64 ... 45
Super Graphix Atari ... 48
Font Master II 64 ... 29

CARDCO
Digitizer Camera ... 189.95
32K Printer Buffer ... 59.95
Numeric Keypad ... 34.95
CB/5 5-slot Board(64) ... 54.00
CB/2 2-slot Board(64) ... 25.00
S'More Basic Rom ... 49.95
Write Now-64 ... 35.00
Mail Now-64 ... 29.00
Spell Now-64 ... 29.00
File Now-64 ... 29.00
Paint Now-64 ... 29.00
Calc Now-64 ... 29.00
Tax Survival ... 29.00
Super Printer Utility ... 27.95
Write Now-Vic-20 ... 29.95

BATTERIES INCLUDED
Paper Clip ... 59.95
Spell Pak ... 34.95
Consultant ... 59.95
Paper Clip w/Spell Pak ... 75.95
Home Pak ... 34.95
Bus Card ... 129.95
80 Column Board ... 109.95

BRODERBUND
The Print Shop ... 28.75
Graphics Library ... 18.75
Graphics Library II ... 19.75
Graphics Library III ... 19.95
Karateka ... 19.75
Castles Dr. Creep ... 19.75
Bank St. Writer ... 32.75
Loderunner ... 20.75
Mask of the Sun ... 24.75
Spelunker ... 19.75
Serpent's Star ... 24.75
Whistler's Brother ... 18.75

MODEMS

RACAL-VADIC
2400PC ... 549
2400PA ... 799
2400V ... 559
1200PC ... 329
300V ... 205
300PC ... 199

TELE LEARNING
Total Telecommunications (C-64) ... 29.95
AP-250 (300 Baud Apple) ... 69.95
IB-250 (300 Baud IBM) ... 69.95

ANCHOR
Volksmodem ... 55
Volksmodem 12 ... 186

US ROBOTICS
Password 1200M ... 229
Password 1200F ... 229
Password 300M ... 139
Password 300F ... 139
Autodial 212A ... 359
PCM5 ... 319
PCM64 ... 519
PCM256 ... 769
S-100 ... 379
Courier ... 469
Microlink ... 469
Telpac Ms-DOS ... 79

NOVATION
IBM 300/1200 MS-DOS ext ... 319
IBM 300/1200 CPM-86 ext ... 319
IBM 300/1200/2400 ext ... 529
IBM 300/1200/2400 MS-DOS ... 579
IBM MS-DOS int ... 325
IBM CPM-86 int ... 325
Macmodem 300/1200 ... 315

HAYES
Smartmodem 300 ... 133
Smartmodem 1200 ... 377
Smartmodem 1200B ... 347
Smartmodem 2400 ... 598
Micromodem IIE ... 135

3.5" DISKETTES

DENNISON
3.5" SSDD ... 5 pak ... 14.95
3.5" SSDD ... 10 pak ... 26.95

MAXELL
3.5" SSDD ... 10 pak ... 29.95
3.5" DSDD ... 10 pak ... 36.95

3 M
3.5" SSDD ... 10 pak ... 26.95
3.5" DSDD ... 10 pak ... 32.95

MONITORS

SAKATA
SG 1000 12" Green ... 99
SA 1000 12" Amber ... 109
SG 1500 12" Green TTL ... 119
SA 1500 12" Amber TTL ... 129
SC 100 13" Color Comp. ... 209
SC 200 13" RGB ... 389
STS1 Tilt Stand ... 29

PRINCETON GRAPHICS
MAX-12 Amber ... 185
HX-12 RGB ... 465

TAXAN
115 12" Greeen Composite ... CALL
116 12" Amber Composite ... CALL
121 12" Green TTL ... 135
122 12" Amber TTL ... 145
220 14" Color Composite ... 259
410 12" RGB Hi Res IBM ... 329
420 12" RGB Super Hi IBM ... 409
440 12" RGB Ultra Hi Res ... 555
Tilt Stand ... 35

TEKNIKA
MJ-10 Composite ... 179

PANASONIC
DT1300D 13" RGB/Composite ... 247
DTM140 14" RGB/Composite ... 329
DTH103 10" RGB Hi Res ... 395
DTS101 10" Composite ... 175
DT1000G 10" RGB ... 166
TX12H3P 12" Color ... 419
TR120M1PA 12" Green ... 109
TR120MBPA 12" Amber ... 109
TR122M9P 12" Green IBM ... 148
TR122MYP 12" Amber IBM ... 148

ZENITH
ZVM 122A Amber ... 75
ZVM 123G Green ... 75
ZVM 124 Amber IBM ... 129
ZVM 131 Color ... 275
ZVM 133 RGB ... 389
ZVM 135 Composite ... 449
ZVM 136 Hi Res Color ... 589
ZVM 1220 ... 95
ZVM 1230 ... 95
ZVM 1240 ... 149

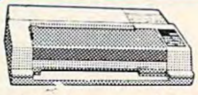
AMDEK
300 Green ... 118
300 Amber ... 128
310 Amber IBM ... 155
Color 300 Audio ... 234
Color 500 Composite ... 369
Color 600 ... 397
Color 700 ... 495
Color 710 ... 569

NEC
JB-1260 Green ... 95
JB-1201 Green ... 135
JC 1215 Color ... 235

LYCO COMPUTER
AMERICA'S MAIL ORDER HEADQUARTERS!

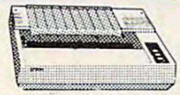
www.commodore.ca

Lyco Computer Marketing & Consultants



1091 . . . . . $228

SAVE ON THESE IN STOCK PRINTERS



SG-10 . . . . $205

SEIKOSHA
SP-1000UC (C-64) . . . 169
SP-1000A Centronics . 185

C. ITOH
Prowriter 8510 sp+ . . Call
15505 sp+ . . . . Call
Printmaster . . . . Call

TOSHIBA
P1340 469
P351+ 1149
P341P 969
P341S 999
351 Sheet Feeder 529

DIGITAL DEVICES
16K BUFFER 75
32K BUFFER 89
64K BUFFER 125

CORONA
LP300 Laser Printer 2686
200361 Toner Cartridge 89

EPSON
LX80 209
FX85 333
JX80 . . . . . Call
Homewriter 10 193
DX10 207
DX20 297
DX35 597
AP-80 244
HI-80 355
HS-80 298
FX-286 (NEW) 489
LQ-800 (NEW) 529
LQ-1000 (NEW) 659

CITIZEN
MSP-10 255
MSP-15 355
MSP-20 337
MSP-25 495

OKIDATA
Okimate 10 179
Okimate 20 . . . . . CALL
182 214
192 348
193 563
84P 645

BROTHER
HR-15XL-P 359
HR-15XL-S 359
HR-35P 839
HR-35S 839
2024L-P 949
M1009-P 189

JUKI
Juki 6100 344
RS232 Serial Board 55
6100 Tractor 119
6100 Sheet Feeder 209
Juki 6300 757

LEGEND
880 188
1080 . . . . . Call
1380 258
1385 289
808 148

DIABLO
D25 549
630 API 1599
630 ECS 1759
P 80 IF 2395
P 32 CQ1 699
P 38 1749
C 150 999
DX-35 (NEW) . . . . . CALL
AP-80 . . . . . CALL

PANASONIC
109 228
3131 (NEW) 264
1092 325
3151 409
1080 (NEW) 209
1592 (NEW) 439

SILVER REED
EXP400 249
EXP500 295
EXP550 399
EXP770 749

STAR MICRONICS
SG-10 205
SG-10C 219
SG-15 367
SD-10 319
SD-15 438
SR-10 469
SR-15 578
SB-10 589
Powertype 297

DUST COVERS

Atari
520ST 11.95
130XE 6.99
800XL 6.99
1050 . . . . . 6.99
1025 . . . . . 7.99

Commodore
C128 . . . . . 7.99
1571/1541 . . . . . 6.99
1902 . . . . . 10.95
1702 . . . . . 8.99
C64/Vic20 . . . . . 6.99

Panasonic
1090/1091 . . . . . 8.99
1092 . . . . . 8.99
1093 . . . . . 9.99

Star Micronics
SG/SD10 . . . . . 8.99
SG/SD15 . . . . . 9.99
SR10 . . . . . 9.99
SR15 . . . . . 9.99

DRIVES

INDUS
Indus GT Atari 195
GT Commodore 209

TYMAC
TCD-640 3½" Apple Drive 640K 289

TANDON
320K 5¼" Drive 115

MSD
SD1 Drive (C-64) 229
SD2 Drive (C-64) 469

INTERFACING

AXIOM
AT846 (Atari) 65
Parallax (C-64) 65

MICROBITS
MPP-1150 (Atari) 54
MPP-1150XL (Atari) 59
MicroPrint (Atari) 39
64K RAM Board (600xl) 65

DIGITAL DEVICES
ApeFace XLP (Atari) 49
ApeFace 12XLP (1200xl) 49
U-Print A (Atari) 54
U-Print C (C-64) 49
U-Print A16K Buffer 74
U-Print A64K Buffer 99
U-Print AP16K (Apple) 99
U-Print AP64K (Apple) 139
U-CALL RS232 (Atari) 37

MICROTEK
Dumpling GX (Apple) 59
Dumpling 16K (Apple) 89
RV-611C (Apple) 49

TYMAC
Connection (C-64) 55
Tackler (Apple) 49
PPC-100 (Apple) 39

ORANGE MICRO
GRAPPLER + (Apple) 85
Grappler 16K (Apple) 149
ORANGE (Apple) 59
Grappler CD (C-64) 79

CARDCO
G-Wiz (C-64) 54
C/? + G (C-64) 45
C/?PS (C-64) 49
C/?B (C-64) 39

DISKETTES

DENNISON
ELEPHANT 5¼" SSSD 11.99
ELEPHANT 5¼" SSDD 12.99
ELEPHANT 5¼" DSDD 14.99
PREMIUM 5¼" SSDD 13.99
PREMIUM 5¼" DSDD 15.99

VERBATIM
5¼" SSDD 13.99
5¼" DSDD 19.99

BONUS
5¼" SSDD 8.99
5¼" DSDD 12.99

SUNKYONG
SKC 5¼" SSDD 11.99
SKC 5¼" DSDD 13.99

MAXELL
5¼" MD1 . . . . . 13.99

NO LABEL DISKETTES
NL 5¼" SSDD . . . 10.99 (Box 10)
NL 5¼" DSDD . . . 15.99 (Box 10)
*Free Diskette Writer Pen!
*Free Storage Case!

IBM-PC SOFT-WARE

ACTIVISION (IBM)
Borrowed Time 24.75
Ghostbusters 24.75
MindShadows 24.75

MICROPROSE (IBM)
F-15 Strike Eagle 20.75
Solo Flight 20.75
Hellcat Ace 20.75
Spitfire Ace 18.75

IBM-PC COMPATABLE

BRODERBUND (IBM)
Bank St. Writer 48.95
The Print Shop 34.95
Graphics Library I 22.95
Lode Runner 22.95
Ancient Art of War 27.95
Champ Lode Runner 22.95

SSI (IBM)
Battle for Normandy 24.95
Knights of Desert 24.95
Tigers in Snow 24.95
Computer Baseball 24.95
Epidemic 22.95
Warp Factor 24.95
Cartels & Cutthroats 24.95

SYNAPSE (IBM)
Synstock 64.95
Essex 28.95
Wizard of Wall St 28.95
Brimstone 28.95

SUB LOGIC (IBM)
Jet Simulator 34.95

*LEADING EDGE
Nutshell 69.95
LEWP Basic 65.00
LEWP Merge Print 99.00
LE Spell Correction 169.00

*QUADRAM
Quad Jr. Exp. Chassis . . . 519.00
Quad Jr. Exp. Memory . . . 209.00
Quad Memory Jr 209.00

*LOTUS
Lotus 1-2-3 . . . . . 309.00
Symphony 439.00

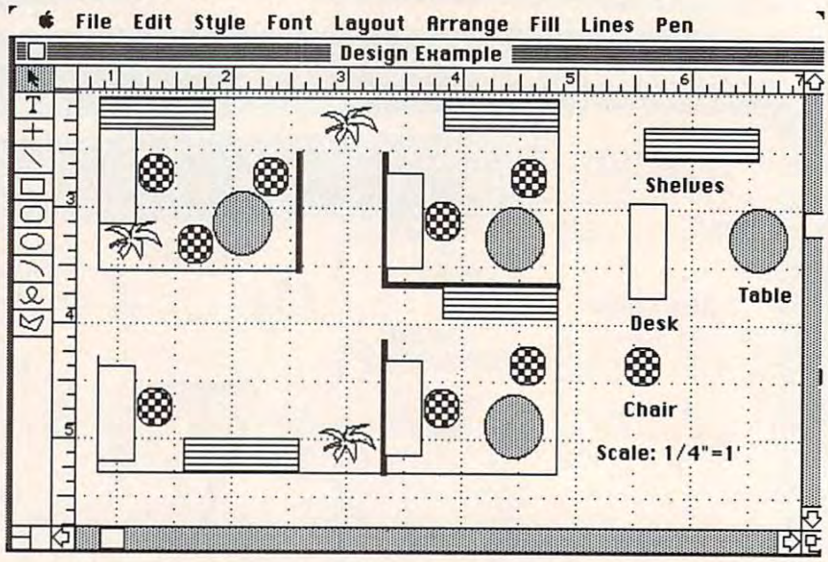
TOLL FREE 1-800-233-8760

TO ORDER
CALL TOLL FREE 1-800-233-8760
In PA 717-494-1030
Customer Service 717-494-1670

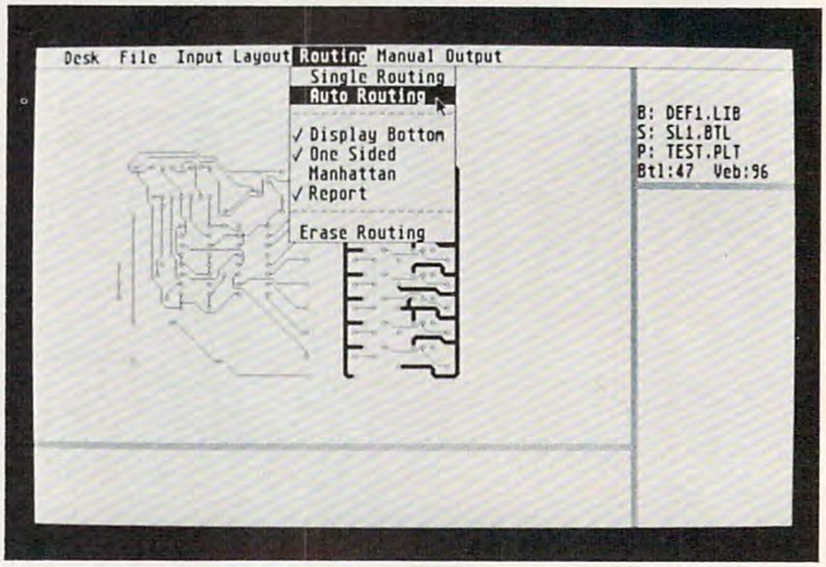
or send order to
Lyco Computer
P.O. Box 5088
Jersey Shore, PA
17740

RISK FREE POLICY
In-stock items shipped within 24 hours of order. No deposit on C.O.D. orders. Free shipping on prepaid cash orders within the continental U.S. Volume discounts available. PA residents add sales tax. APO, FPO, and international orders add $5.00 plus 3% for priority mail service. Advertised prices show 4% discount for cash, add 4% for MasterCard or Visa. Personal checks require 4 weeks' clearance before shipping. Ask about UPS Blue and Red label shipping. All merchandise carried under manufacturer's warranty. Free catalog with order. All items subject to change without notice.

www.commodore.ca



An office floorplan created with Apple Computer's MacDraw *for the Macintosh.*



Creating a printed circuit board layout is greatly simplified with Abacus Software's PCBoard Design *for the Atari ST.*

CAD software doesn't require the user to be an artist. Most CAD packages incorporate functions for creating lines, polygons, arcs, circles, rectangles, rounded corners, angles, and pattern fills, and for duplicating and resizing objects. In addition, built-in rulers and grid patterns take the guesswork out of precisely placing objects and text.

• **Object-based versus pixel-based programs:** With a CAD program, you can create and name a shape, then move it around the screen, recall it from memory, and reuse it at will. The program treats the shape as an object rather than as a collection of individual pixels. Most paint programs can't identify an image as an object, while CAD packages must have this capability to be useful.

Obviously, there are many more features that make up a CAD program, and they vary from package to package, just as prices range from less than a hundred dollars to several thousand dollars. But even the most expensive CAD programs for personal computers cost considerably less than high-end CAD hardware and software; advanced solid-modeling, three-dimensional packages can cost tens of thousands of dollars.

One of the most popular and powerful CAD packages for personal computers is *AutoCad* from Autodesk, Inc., used by many professional designers. So popular has the program become that there are dozens of support packages which add a variety of specific architectural and engineering tools. In February, the first national convention and trade show devoted to CAD systems for microcomputers was held near Chicago—and not surprisingly, the show was called *AutoCad*Con. It focused primarily on the *AutoCad* package and its support programs. Response to the show exceeded expectations, and the *AutoCad*Con sponsors even arranged for as many as 1,200 universities across the nation to receive satellite downlinks from the conference's 20 technical sessions.

Whether your interest in CAD is professional or casual, there are a growing number of software packages at every level of price and performance. However, it's difficult to understand how revolutionary and how widely applicable CAD can be until you've actually worked with one of these packages. What is it that CAD actually does for the designer?

"A word processor for drawing"—that's how software designer William Volk describes a good CAD program. Just as a word processor helps a writer assemble and edit words with unprecedented flexibility, CAD programs offer drafting tools that are versatile, fast, and simple to use. And, just as a word processor can change the way a writer approaches writing, so a CAD package can alter your view of design.

Volk is the creator of *Aegis Draw*, a sophisticated CAD program from Aegis Development, Inc., for the Amiga. *Aegis Draw* represents the direction in which CAD developers are headed today.

www.commodore.ca

Get the jump on the weatherman by accurately forecasting the local weather yourself!

A scientifically proven way to develop an awesome memory.

You are trapped in a five-story, 125-room structure made entirely of ice. Find the exit before you freeze!

Take control of your personal finances in less than one hour a month.

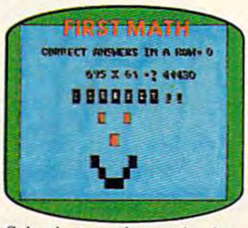
The beautiful princess is held captive by deadly dragons. Only a knight in shining armor can save her now!

Cut your energy costs by monitoring your phone, electric and gas bills.

Computerize car maintenance to improve auto performance, economy and resale value.

Create multi-colored bar graphs with a surprisingly small amount of memory.

A time-saving organizer for coupons, receipts and more.



School-age and pre-school children are rewarded for right answers, corrected on their wrong ones.

A real brainflexer. Deflect random balls into targets on a constantly changing playfield.

A fun way to dramatically increase typing speed and accuracy.

Get up to 20 new programs and games every month in COMPUTE!

Every month, COMPUTE! readers enjoy up to 20 brand-new, ready-to-run computer programs, even arcade-quality games.

And when you subscribe to COMPUTE!, you'll get them all for less than 15 cents each!

You'll find programs to help you conserve time, energy and money. Programs like Cash Flow Manager, Retirement Planner, Coupon Filer, Dynamic Bookkeeping.

You'll enjoy games like Air Defense, Boggler, Slalom, and High Speed Mazer.

Your children will find learning fast and fun with First Math, Guess That Animal, and Mystery Spell.

Looking for a challenge? You can write your own games. Customize BASIC programs. Even make beautiful computer music and pictures.

It's all in COMPUTE!. All ready to type in and run on your Atari, Apple, Commodore, TI-99/4A, IBM PC, or PCjr computers.

What's more, you get information-packed articles, product reviews, ideas and advice that add power and excitement to all your home computing.

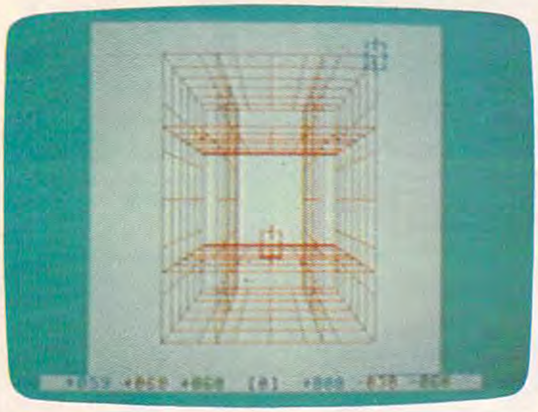
And when it's time to shop for peripherals or hardware, check COMPUTE! first. Our product evaluations can save you money and costly mistakes. We'll even help you decide what to buy: Dot-matrix or daisy-wheel printer? Tape storage or disk drive? What about modems? Memory expansion kits? What's new in joysticks, paddles, and track balls?

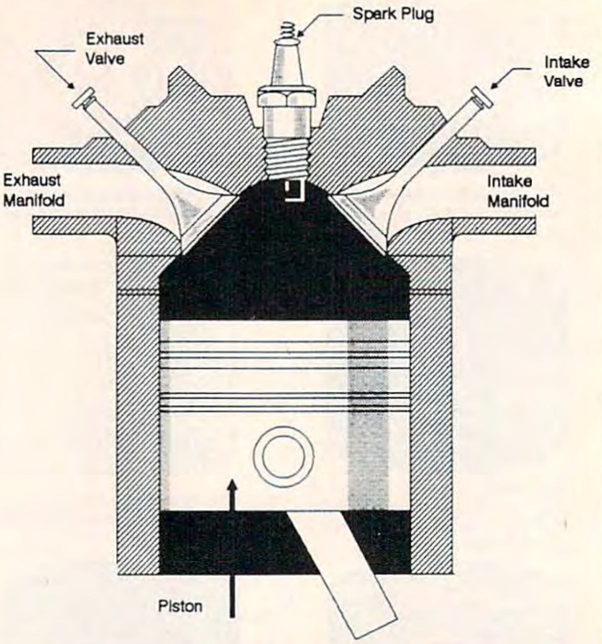
Order now! Mail the postpaid card attached to this ad and start receiving every issue of COMPUTE!.

For Faster Service Call Toll-Free 1-800-247-5470 (in Iowa 1-800-532-1272)

COMPUTE! P.O. Box 10954, Des Moines, IA 50340

www.commodore.ca





A three-dimensional image constructed with CAD-3D *for the Commodore 64 (above).* Easy-Draw *for the Atari ST is a structured drawing program with many CAD features (on right).*

"High performance and ease of use," says Volk. "That's the philosophy of the program. We wanted to break new ground on that."

Volk and other CAD developers are taking advantage of the new generation of personal computers—chiefly the Amiga, ST, and Macintosh—to create CAD packages that are far more sophisticated and yet less expensive than previous programs. Faster, more powerful microprocessors, more memory, multitasking operating systems, and custom graphics chips are all contributing to an expanded universe for CAD developers and users.

On the Amiga, Volk was able to take advantage of the machine's multitasking in ways that would be very difficult or impossible on other computers. "The idea being that you should be able to have multiple windows on the same drawing so that you can actually work in detail on a drawing and still maintain an overall view," he says. In other words, while one window on the Amiga screen shows the entire design, a second window can zoom onto a tiny area of the image for detailed work. Make a change in one window and it's reflected in the other window as well—two windows, both active.

Aegis Draw has other advanced features as well, including an almost infinite level of zoom. For example, you could zoom from an image of the Earth down to the level at which you could read a car's license plate. Additional features include object rotation as fine as .001 degree, angling, variable line thicknesses, and line patterning.

Volk sees two markets for his program and other CAD software: "One is the vertical application people—the engineers and the architects. And the other market is the people who aren't artistic enough to use [a computer paint program] accurately. In other words, if you want to create an organizational chart [with a paint program], typically people are not going to do a great job. But if you're using *Draw*, you can be rather clumsy and still end up with really accurate-looking drawings."

Although *Aegis Draw* can be used by both professionals and hobbyists, Aegis Development has a still higher-end package, *Aegis Draw Professional (Pro)*, that's upwardly compatible with *Aegis Draw*.

One of the most popular and easy to use design programs is Apple Computer's *MacDraw* for the Macintosh. Although not aimed at the professional design market, *MacDraw* and its companion program, *MacPaint*, broke new ground when they were introduced. Their use of icons, pull-down menus, command bars, and mouse-controlled cursor movement are all very well suited to CAD work. Similar user interfaces are becoming a virtual standard among programs for the Amiga and Atari ST.

One interesting new CAD package, both from a professional and educational standpoint, is a program for the Atari ST that automates printed circuit board layouts. Called *PCBoard Designer*, this $395 package was originally developed in West Germany and is now marketed in the U.S. by Abacus Software.

Aimed primarily at the narrow market of printed circuit designers, *PCBoard Designer* also offers valuable hands-on experience for high school and college electronics students. *PCBoard Designer* is a good example of how thousands of hours of development time can be eliminated by a computer with CAD software.

In printed circuit design, there are two phases of work that require large amounts of time, explains Arnie Lee, president of Abacus Software. One phase, called *tracing*, is the layout of circuit traces from one point to another on the board. None of the traces can overlap, or a short circuit would result. With *PCBoard Designer*, the tracing is handled by *autorouting*—automatically routing points that need to be connected on the PC board. The computer program determines the best layout for the traces in seconds.

The second time-consuming phase of printed circuit design is

www.commodore.ca

when last-minute changes force the technician to redraw all traces from scratch. Again, what might have taken a couple of days of tedious tracing and retracing can now be done by the computer in seconds.

CAD programs can work similar wonders for all kinds of projects. Whether you're designing a printed circuit, compiling a family tree, or planning an office flowchart, CAD packages are becoming as easy to use and as powerful as the new computers they run on. Even the most advanced three-dimensional solid modeling will one day be as common on personal computers as two-dimensional graphics are now.

For More Information

While space does not permit a comprehensive listing of all CAD programs now available, the following should help you get started:

Aegis Draw
for Commodore Amiga
Aegis Development
2210 Wilshire Blvd., #277
Santa Monica, CA 90403
$199.95

AutoCad
for IBM PC family, compatibles, and many business computers
Autodesk, Inc.
2320 Marinship Way
Sausalito, CA 94965
Basic package: $1,000
With ADE2 *(Advanced Drafting Extension) and* ADE3: *$2,500*

CadApple
for Apple II family (minimum 64K)
T&W Systems, Inc.
7372 Prince Drive
Suite 106
Huntington Beach, CA 92647
Entry-level package: $495

CAD-3D
for Commodore 64
IHT Software
2269 Chestnut Street
Suite 162
San Francisco, CA 94123
$39.95

CadPak *(Enhanced Version)*
for Commodore 64
CadPak 128
for Commodore 128
Abacus Software
2201 Kalamazoo S.E.
P.O. Box 7211
Grand Rapids, MI 49510
64 version $39.95
128 version $59.95

CadPlan
for IBM PC family and compatibles (minimum 320K)
CalComp
2411 West La Palma Avenue
P.O. Box 3250
Anaheim, CA 92803
$2,000

Computer-Aided Design for the C-128 and C-64
(Hands-on introductory book on CAD for Commodore 64 (using Simons' BASIC*) and 128)*
Abacus Software
2201 Kalamazoo S.E.
P.O. Box 7211
Grand Rapids, MI 49510
Book: $19.95
Optional disk: $14.95

Easy-Draw
for Atari ST
Migraph, Inc.
720 S. 333rd Street
Suite 201
Federal Way, WA 98003
$149.95

MacDraw
for Apple Macintosh
Apple Computer, Inc.
20525 Mariani Avenue
Cupertino, CA 95014
$195

PCBoard Designer
for the Atari ST
Abacus Software
2201 Kalamazoo S.E.
P.O. Box 7211
Grand Rapids, MI 49510
$395

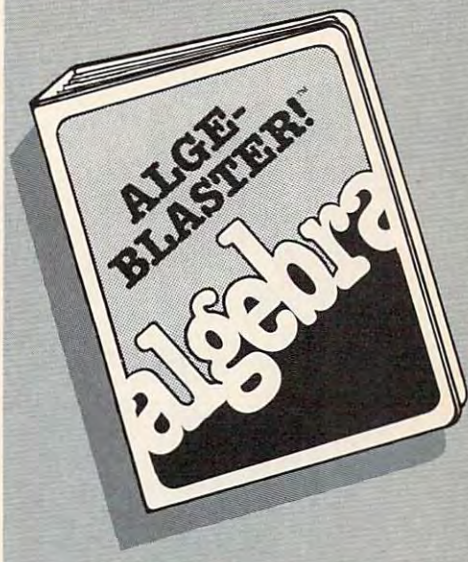
PC-Draw
for IBM PC family and compatibles (minimum 256K)
Micrografx, Inc.
1701 North Greenville
Suite 305
Richardson, TX 75801
$395

Robo Graphics CAD-1+ *and* CAD-2
for Apple II family (minimum 64K)
Chessell-Robocom Corporation
Robo Systems
111 Pheasant Run
Newtown, PA 18940
CAD-1+: $695
CAD-2 (without RAM card): $1,095
CAD-2 (including RAM card): $1,320

VersaCad
for IBM PC family and compatibles
T&W Systems, Inc.
7372 Prince Drive
Suite 106
Huntington Beach, CA 92647
Entry-level package: $495

©

NOW...from the creators of MATH BLASTER!™



ALGE-BLASTER!™

Learn the abc's of

$a^2 + b^2 = c^2$

ALGE-BLASTER! is the most complete algebra program ever put on one disk. Master all the fundamentals: positive and negative numbers, monomials and polynomials, factoring, and equations—670 problems in all! Receive step-by-step tutoring... earn graphic rewards for right answers...add new problems with Davidson's easy-to-use editor...and enjoy sound effects, score-keeping and print features, and much, much more. 7th–12th grade. Apple™ II family (64K). IBM™ version available 11/85.

Educational Software That Works.

Davidson & Associates, Inc.
800-556-6141
(In Calif., 213-534-4070)

Davidson.

Davidson & Associates, Inc.
3135 Kashiwa Street
Torrance, CA 90505

teacher certified NEA™

Please send me a FREE COLOR BROCHURE and the name of my nearest Davidson Dealer.

Name__________

Address__________

City______ State______ Zip______

www.commodore.ca

Hickory, Dickory, Dock

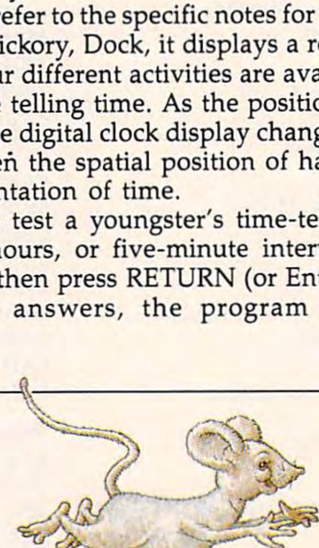
Barbara H. Schulak

This fun, educational program helps children learn the concepts of telling time by relating a digital clock display to a conventional clock face. The original program is written for the Commodore 64 (and 128 in 64 mode). We've added new versions for Apple II-series computers, the IBM PC/ PCjr, Amiga, Atari 520ST, and Atari 400/800, XL, and XE computers.

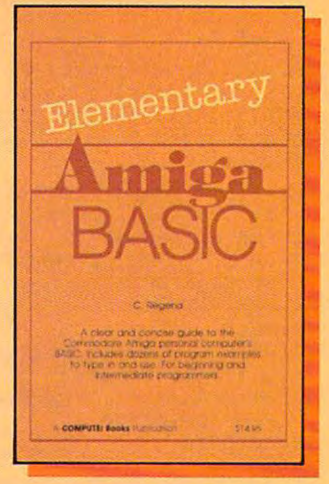
"Hickory, Dickory, Dock" offers an enjoyable way for children to learn how to tell time. Type in the program for your computer, then save a copy before running it. Because every version works much the same, read the general instructions first, then refer to the specific notes for your computer.

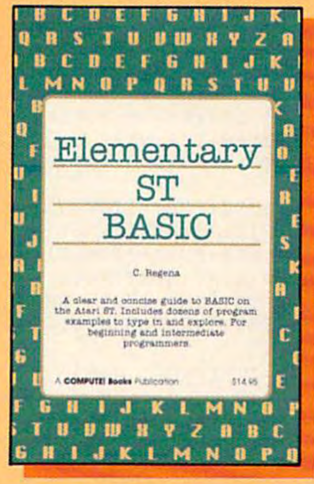
When you run Hickory, Dickory, Dock, it displays a round clock face as well as a digital display. Four different activities are available. The first option lets youngsters practice telling time. As the positions of the clock hands change on the screen, the digital clock display changes as well. This shows the relationship between the spatial position of hands on a clock face and the numeric representation of time.

The other three activities test a youngster's time-telling ability for hours only, hours and half-hours, or five-minute intervals. Move the hands to the correct position, then press RETURN (or Enter) to enter the answer. After five correct answers, the program plays a brief



www.commodore.ca





Elementary Amiga BASIC
C. Regena
0-87455-041-6, $14.95
Disk $15.95

Elementary ST BASIC
C. Regena
0-87455-034-3, $14.95
Disk $15.95

BASIC programming at its best!

Two new programming guides from COMPUTE! Books.

Written by the author of the bestselling *Programmer's Reference Guide to the TI-99/4A*, these books introduce you to the new and powerful BASIC on the Amiga and Atari ST personal computers. The computers' impressive graphics, animation, and sound can be unlocked with the right commands, and BASIC is the place to start. Regena shows you how— in the clear, concise language that's made her such a popular writer.

Complete descriptions of the Amiga's and ST's BASIC commands, syntax, and organization take you from novice to full-power programming. Sample programs and subroutines, all ready to type in, are included. Plus, both books offer you working software while showing you how to write your own programs. *A disk is also available for each book which includes all the programs from the book in an easy, ready-to-load format.*

SPECIAL COMBINATION OFFER
Order the book and disk together for only $29.95!

You'll find these new programming guides and many more useful, entertaining COMPUTE! books at your local computer and book stores. Or you can order directly from COMPUTE! Books.

For the fastest service, call toll free 1-800-346-6767 (in NY 212-887-8525). Or mail the attached coupon with your payment to COMPUTE! Books, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

Please add shipping and handling charges to all orders: In U.S., $2.00 per book, disk, or combination package; $5.00 per item airmail.

I want the best in BASIC programming guides. Please send me:

______	***Elementary ST BASIC,*** (034-3), $14.95 each	______
______	***Elementary ST BASIC Disk,*** $15.95 each	______
______	***Elementary ST BASIC*** disk and book combination, $29.95	______
______	***Elementary Amiga BASIC,*** (041-6), $14.95 each	______
______	***Elementary Amiga BASIC Disk,*** $15.95 each	______
______	***Elementary Amiga BASIC*** disk and book combination, (057-2), $29.95	______
	Subtotal	______
	NC residents add 4.5% sales tax	______
	Shipping charges ($2.00 per item in U.S. and surface mail; $5.00 per item airmail)	______
	Total amount enclosed	______

ALL ORDERS MUST BE PREPAID IN U.S. FUNDS

☐ Payment enclosed (check or money order)

☐ Charge ☐ MasterCard ☐ Visa ☐ American Express

Account No. ______________ Exp. Date ______________ (Required)

Name ______________

Address ______________

City ______________

State ______________ Zip ______________

Allow 4–6 weeks for delivery.

36512111

COMPUTE! Publications, Inc. abc
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor, New York, NY 10019
Publishers of COMPUTE!, COMPUTE!'s Gazette, COMPUTE!'s Gazette Disk, COMPUTE! Books, and COMPUTE!'s Apple Applications

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, England and in Canada from McGraw-Hill, Ryerson Ltd., 330 Progress Ave., Scarborough, Ontario, Canada M1P 2Z5.

www.commodore.ca

song and displays some graphics as a reward. After three incorrect choices, the program automatically moves the clock hands to the correct position.

In the Commodore 64 version (which also works on the 128 in 64 mode), press the f7 function key to move the minute hand (the long green one), and the f5 function key to move the hour hand (short and yellow). If you press M at any point, the program returns to the main menu.

For all Atari versions (including the ST), press the H key to move the hours hand, press M to move the minutes hand, and press Q to return to the main menu. Before running the Atari ST version, you should select Low Resolution from the Preferences menu and turn off buffered graphics. (If your ST has the TOS operating system in Read Only Memory, there may be enough Random Access Memory left to run Hickory, Dickory, Dock without turning off buffered graphics.)

The Amiga version includes speech synthesis and uses the mouse pointer instead of keyboard controls.

In the Apple and IBM PC/PCjr versions, press the 1 key to move the minutes hand, press 2 to move the hours hand, and press ESC to return to the main menu.

Program 1: Hickory, Dickory, Dock For Commodore 64

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!

```
AF 1Ø  POKE53281,Ø:POKE5328Ø,11
CP 2Ø  PRINT"{CLR}{WHT}{8 DOWN}
       "TAB(8)"HICKORY, DICKORY
       , DOCK!
QR 3Ø  PRINT"{1Ø DOWN}"TAB(14)"
       PLEASE WAIT
XC 4Ø  POKE56,48:CLR
DC 5Ø  POKE65Ø,128:X=RND(-TI)
GJ 6Ø  DIM N(12),LO(12,6),CH(12
       ,6),NU(1Ø,15),T1(12),T2(
       12)
MB 7Ø  S=54272:FORI=ØTO24:POKES
       +I,Ø:NEXT:POKES+24,15
MM 8Ø  FORI=1TO3Ø:READA$,A,B,C:
       NEXT
DB 9Ø  V=53248:POKE2Ø4Ø,13
RM 1ØØ FORI=ØTO63:READA:POKE83
       2+I,A:NEXT
FQ 11Ø POKEV+39,2:POKEV+37,1:P
       OKEV+38,6
SF 12Ø POKEV,15Ø:POKEV+1,15Ø
XM 13Ø POKEV+23,1:POKEV+29,1
MD 14Ø POKEV+28,1 :POKEV+21,1
PF 15Ø FORI=49152TO49152+81:RE
       ADA:POKEI,A:NEXT
CS 16Ø SYS49152
JJ 17Ø FORI=ØTO143:READA:POKE1
       2288+1Ø1*8+I,A:NEXT
CB 18Ø FORI=1TO12:READ N(I):NE
       XT
JR 19Ø FORI=1TO12:FORJ=1TON(I)
       :READLO(I,J):NEXT:NEXT
QS 2ØØ FORI=1TO12:FORJ=1TON(I)
       :READCH(I,J):NEXT:NEXT
SM 21Ø FORI=ØTO9:FORJ=1TO15:RE
       ADA$:IFA$="+"THENNU(I,J
       )=16Ø
KH 22Ø IFA$="-"THENNU(I,J)=32
ES 23Ø NEXT:NEXT
RJ 24Ø FORI=1TO12:READT1(I),T2
       (I):NEXT
DA 25Ø RESTORE:POKEV+21,Ø
DR 26Ø PRINT"{CLR}{3 DOWN}
       {WHT}CHOOSE ONE OF THE
       {SPACE}FOLLOWING:
AX 27Ø PRINTTAB(5)"{2 DOWN}1.
       {SPACE}TEST - HOURS
SB 28Ø PRINTTAB(5)"{DOWN}2. TE
       ST - HOURS AND HALF HOU
       RS
AK 29Ø PRINTTAB(5)"{DOWN}3. TE
       ST - 5 MIN. INTERVALS
BR 3ØØ PRINTTAB(5)"{DOWN}4. PR
       ACTICE
JJ 31Ø PRINTTAB(5)"{DOWN}5. EN
       D PROGRAM"
JK 32Ø GETKK$:IFKK$=""THEN32Ø
KA 33Ø KK=VAL(KK$):IFKK<1ORKK>
       5THEN32Ø
JE 34Ø IFKK=5THENSYS2Ø48:END
HX 35Ø POKE53272,(PEEK(53272)A
       ND24Ø)+12
SD 36Ø RA=Ø:WA=Ø
CD 37Ø PRINT"{CLR}{3 DOWN}"
PJ 38Ø PRINTTAB(19)"12"
XM 39Ø PRINTTAB(14)"11
       {9 SPACES}1"
HF 4ØØ PRINT:PRINTTAB(11)"1Ø
       {15 SPACES}2"
HR 41Ø PRINT:PRINT:PRINTTAB(11
       )"9{8 SPACES}Q
       {8 SPACES}3"
HS 42Ø PRINT:PRINT:PRINTTAB(12
       )"8{15 SPACES}4"
CP 43Ø PRINT:PRINTTAB(15)"7
       {9 SPACES}5"
DD 44Ø PRINTTAB(2Ø)"6"
DP 45Ø PRINT"{3 DOWN}{YEL}F5=*
       ***
BB 46Ø PRINT"{GRN}F7=<*****
QX 47Ø PRINT"{WHT}RETURN=ANS.
GP 48Ø PRINT"M=MENU
JS 49Ø H=2:M=12:GOSUB95Ø:GOSUB
       1Ø6Ø
BH 5ØØ POKE1844,16Ø:POKE1844+S
       ,2
GE 51Ø POKE1924,16Ø:POKE1924+S
       ,2
XB 52Ø IFKK=4THEN247Ø
EM 53Ø C=H:D=M:PRINT"{WHT}
       {HOME}SET THE HANDS ON
       {SPACE}THE CLOCK:"
FB 54Ø GOSUB159Ø:H=A:M=B:GOSUB
       123Ø:GOSUB145Ø:H=C:M=D
KD 55Ø :
RP 56Ø POKE198,Ø
JF 57Ø K$="":GETK$:IFK$=""THEN
       57Ø
DR 58Ø IFK$="{F5}"THENGOSUB111
       Ø
KG 59Ø IFK$="{F7}"THENGOSUB116
       Ø
KB 6ØØ IFK$=CHR$(13)THEN64Ø
JE 61Ø IFK$="M"THEN26Ø
MR 62Ø GOTO57Ø
SJ 63Ø :
RK 64Ø IFH<>AORM<>BTHEN77Ø
HP 65Ø PRINT"{HOME}{2 DOWN}"TA
       B(17)"{RVS}[7]RIGHT!"
KQ 66Ø GOSUB231Ø
DH 67Ø FORI=1TO1Ø
FD 68Ø PRINT"{HOME}{2 DOWN}"TA
       B(17)"{RVS}RIGHT!"
QX 69Ø FORT=1TO5Ø:NEXT
MH 7ØØ PRINT"{HOME}{2 DOWN}"TA
       B(17)"RIGHT!"
DS 71Ø FORT=1TO5Ø:NEXT
CS 72Ø NEXT
DB 73Ø PRINT"{HOME}{2 DOWN}"TA
       B(17)"{6 SPACES}"
CK 74Ø WA=Ø:RA=RA+1:IFRA=5THEN
       RA=Ø:GOTO17ØØ
MC 75Ø GOTO53Ø
GB 76Ø :
HJ 77Ø PRINT"{HOME}{2 DOWN}"TA
       B(17)"{PUR}WRONG!"
FF 78Ø GOSUB239Ø:WA=WA+1
SF 79Ø FORT=1TO1ØØØ:NEXT
CR 8ØØ PRINT"{HOME}{2 DOWN}"TA
       B(17)"{6 SPACES}"
GC 81Ø IFWA<3THEN57Ø
KE 82Ø WA=Ø:PRINT"{HOME}{WHT}T
       HE CORRECT TIME IS:
       {7 SPACES}"
ME 83Ø GOSUB91Ø:GOSUB1Ø2Ø
GK 84Ø H=A:M=B:GOSUB95Ø:GOSUB1
       Ø6Ø
XK 85Ø FORT=1TO2ØØØ:NEXT
HH 86Ø PRINT"{HOME}{2Ø SPACES}
       "
HM 87Ø GOTO53Ø
FK 88Ø :
FD 89Ø :CHANGE MINUTE HAND
AK 9ØØ :
DQ 91Ø FORI=1TON(M)
CH 92Ø POKELO(M,I),32
PA 93Ø NEXT:RETURN
AQ 94Ø :
AD 95Ø FORI=1TON(M)
ME 96Ø POKELO(M,I),CH(M,I)
SX 97Ø POKELO(M,I)+S,5
JJ 98Ø NEXT:GOSUB225Ø:RETURN
HA 99Ø :
BB 1ØØØ :CHANGE HOUR HAND
SX 1Ø1Ø :
BH 1Ø2Ø FORI=1TON(H)-1
PK 1Ø3Ø POKELO(H,I),32
MJ 1Ø4Ø NEXT:RETURN
PC 1Ø5Ø :
RP 1Ø6Ø FORI=1TON(H)-1
KS 1Ø7Ø POKELO(H,I),CH(H,I)
PK 1Ø8Ø POKELO(H,I)+S,7
GP 1Ø9Ø NEXT:GOSUB225Ø:RETURN
GF 11ØØ :
KK 111Ø GOSUB91Ø:GOSUB1Ø2Ø
FA 112Ø H=H+1:IFH=13THENH=1
MB 113Ø GOSUB1Ø6Ø:GOSUB95Ø
KG 114Ø RETURN
FK 115Ø :
PF 116Ø GOSUB91Ø:GOSUB1Ø2Ø
KD 117Ø M=M+1:IFM=13THENM=1
KR 118Ø GOSUB1Ø6Ø:GOSUB95Ø
MM 119Ø RETURN
AP 12ØØ :
QR 121Ø : CHANGE HOUR NUMBER
GR 122Ø :
CP 123Ø IFH<1ØTHEN13ØØ
EX 124Ø K=Ø
AB 125Ø FORI=1796TO1956STEP4Ø
SP 126Ø FORJ=ØTO2:K=K+1
XH 127Ø POKEI+J,NU(1,K):POKEI+
        J+S,2
FJ 128Ø NEXT:NEXT
FX 129Ø TP=H-1Ø:GOTO131Ø
EG 13ØØ TP=H
GD 131Ø K=Ø
```


www.commodore.ca

240K Apple® Compatible Computer System

NEW SALE

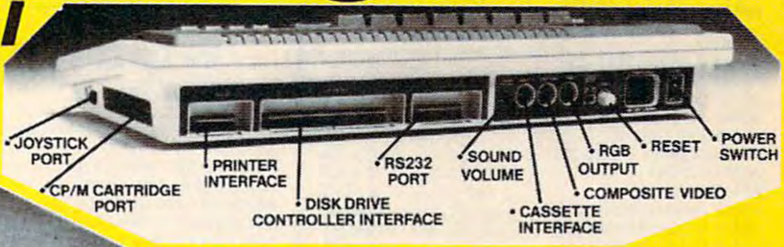
APlus 3000 computer system includes 192K RAM, 48K ROM (32K Microsoft Basic plus 16K ROM Emulator), 160K Laser 5¼" Disk Drive (Runs Apple II Software), Magic Window Wordprocessor, MagiCalc spreadsheet, Magic Memory Database. All for only $399.00

Complete System

$399.00

• *15 Day Free Trial*

"Aplus 3000" System



Double Immediate Replacement Warranty
If any of the Aplus 3000 computer system equipment fails due to faulty workmanship or material within 180 days of purchase we will REPLACE it immediately with no service charge!!

- **Over 10,000 existing Apple® programs**
- **Centronics printer interface included**
- **240K (192K RAM, 48K ROM)**
- **ArtSci's Magic Window II, Magic Memory, and MagiCalc included**
- **160K Laser 5¼" Disk Drive (Runs Apple II software)**
- **RGB (80 columns in color) and composite included**

SPECIFICATIONS

A plus 3000 is a complete, self-contained computer based on the popular 6502A microprocessor and can tap into the tremendous software library of Apple II. Features include 192K Bytes RAM, 32KB Enhanced Microsoft BASIC, 80 column text, 560H X 192V color graphic display, 81 key sculptured keyboard and high efficiency switching power supply. Also included as standard are Centronics bus printer interface, Cassette interface, 4 channel sound generator, and 5¼" Apple Compatible Disk Drive.

• TEXT
- 40 columns X 24 rows or 80 columns X 24 rows software selectable.
- 5 X 7 characters in 7 X 8 matrix.
- Upper and lower case characters.
- One of Eight colors for characters/graphics and background, Red, Green, Blue, Cyan, Magenta, Yellow, Black and White.
- Character set with normal, inverse and flashing capabilities.

• GRAPHICS
- 280H X 192V 6 colors — Black, White, Violet, Green, Blue, Orange.
- 280H X 192V 8 colors bit image — Black, White, Red, Green, Blue, Cyan, Magenta, Yellow.
- 560H X 192V 6 colors — Black, White, Violet, Green, Blue, Orange. (High resolution color monitor required)

More Features than Apple® for less than Commodore®

Features	Aplus 3000	Apple IIe	Commodore C-128
RAM	192K	64K	128K
Runs Apple II Software	Yes	Yes	No
Function Keys	24	None	16
4 Voice, 6 Octave Sound	Yes	No	Yes
Composite Video	Yes	Yes	Yes
Disk Drive	Included	Extra Cost	Extra Cost
Numeric Keypad	Included	Extra Cost	Included
Video Cable	Included	Extra Cost	Extra Cost
RGB Color Card	Included	Extra Cost	Included
80 Column Card	Included	Extra Cost	Included
Centronics Printer Interface	Included	Extra Cost	Extra Cost
Drive Controller	Included	Extra Cost	Included
$150 Wordprocessor (Magic Window)	Included	Extra Cost	Extra Cost
$150 Spreadsheet (MagiCalc)	Included	Extra Cost	Extra Cost
$60 Database prg. (Magic Memory)	Included	Extra Cost	Extra Cost
Your Cost	**$399.00**	$1745.00	$1117.90

Super Apple Compatible Disk Drive Sale $149.95.
Quieter, Cooler, Better Disk Drives for your Apple II plus, IIe, IIc (specify when ordering). List $299.95. Sale $149.95.

15 Day Free Trial — If it doesn't meet your expectations within 15 days of receipt, just send it back to us UPS prepaid and we will refund your purchase price!!

ACCESSORIES

ACCESSORIES	LIST	SALE
2nd Disk Drive	$299.95	$149.95
2 professional analog joysticks	$ 39.95	$ 24.95
Z-80 cart. allows CP/M use	$ 99.95	$ 59.95
RS232 adapter	$ 99.95	$ 59.95
R/F Modulator (TV hookup)	$ 29.95	$ 19.95
RGB cable (RGB Monitor hookup)	$ 24.95	$ 19.95
Centronics cable (for Centronics printer)	$ 34.95	$ 24.95
Technical reference manual	$ 29.95	$ 19.95
80 columns Hi-Res Green Monitor	$199.00	$ 79.95
80 column Hi-Res RGB Monitor	$399.00	$259.00

Add $25.00 for shipping and handling!!
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery. 2 to 7 days for phone orders. 1 day express mail! We accept Visa and MasterCard. We ship C.O.D. to continental U.S. addresses only. Add $10 more if C.O.D.

COMPUTER DIRECT

We Love Our Customers
22292 N. Pepper Rd., Barrington, Ill. 60010
312/382-5050 to order

APPLE and COMMODORE are registered trademarks of Apple Computer Inc. and Commodore Business Machines, Inc., respectively

www.commodore.ca

CALL ATARI CALL

312-382-5050 — **Software Sale** — **312-382-5050**

(T) Tape, (C) Cartridge, (D) Disk.

GAMES

Electronic Arts

- A0684 DR. J & LARRY BIRD GO 1 ON 1 (D) $24.95
- A0685 MOVIE MAKER (D) $23.95
- A0686 SEVEN CITIES OF GOLD (D) $23.95
- A0687 PINBALL CONSTRUCTION SET (D) $16.95
- A0688 MUSIC CONSTRUCTION SET (D) $16.95
- A0689 FINANCIAL COOKBOOK (D) $27.95
- A0690 M.U.L.E. (D) $16.95
- A0691 MURDER ON THE ZINDERNUEF (D) $16.95

Atari

- A0544 STAR RAIDERS (C) $14.95
- A0545 MISSILE COMMAND (C) $14.95
- A0546 GALAXIAN (C) $14.95
- A0547 DEFENDER (C) $14.95
- A0548 DIG DUG (C) $16.95
- A0549 DONKEY KONG (C) $16.95
- A0555 PENGO (C) $16.95
- A0556 MILLIPEDE (C) $16.95
- A0557 JUNGLE HUNT (C) $16.95
- A0559 MOON PATROL (C) $16.95
- A0560 FINAL LEGACY (C) $16.95
- A0561 FOOTBALL (C) $14.95
- A0562 TENNIS (C) $14.95
- A0563 TRACK & FIELD (C) $29.95
- A0564 EASTERN FRONT (C) $14.95
- A0566 ADVENTURE WRITER (D) $19.95
- A0567 STAR VOYAGER (D) $14.95
- A0568 SARGON II (D) $16.95
- A0569 MS PACMAN (C) $16.95
- A0570 DONKEY KONG JR (C) $16.95
- A0571 POLE POSITION (D) $18.95

Broderbund

- A0514 MASK OF THE SUN (D) $24.95
- A0515 OPERATION WHIRLWIND (D) $24.95
- A0516 SPELUNKER (D) $18.95
- A0517 LODE RUNNER (D) $20.95
- A0518 WHISTLERS BROTHER (D) $18.95
- A0502 STEALTH (D) $18.95
- A0670 CHAMPIONSHIP LODE RUNNER (D) $26.95
- A0671 KARATEKA (D) $23.95

Activision

- A0597 PAST FINDER (D) $20.95
- A0598 SPACE SHUTTLE (D) $19.95
- A0599 GHOSTBUSTERS (D) $18.95
- A0665 HACKER (D) $19.95
- A0666 MIND SHADOW (D) $18.95
- A0667 MASTER OF THE LAMPS (D) $19.95
- A0668 GR AMER CROSS CNTRY RD RACE (D) $18.95

Suncom

- A0190 PARTY QUIZ (D) $14.95
- A0193 GENL EDITION 2 (D) $14.95
- A0194 GENL EDITION 3 (D) $14.95
- A0195 SPORTS EDITION (D) $14.95
- A0196 EDUCATION EDITION (D) $14.95
- A0197 BIBLE EDITION (D) $14.95
- A0198 ENTERTAINMENT EDITION (D) $14.95

Avalon Hill

- A0573 TGIF (D) $16.95
- A0574 FLYING ACE (D) $22.95
- A0575 MIDWAY CAMPAIGN (T) $12.95
- A0576 B-1 NUCLEAR BOMBER (T) $18.95
- A0577 LEGIONNAIRE (D) $20.95
- A0578 TAC (D) $26.95
- A0579 MARKET FORCES (D) $14.95
- A0580 PANZER JAGD (D) $20.95
- A0604 FREE TRADER (D) $19.95
- A0605 EMPIRE OF THE OVERMIND (D) $26.95
- A0606 QUEST OF THE SPACE BEAGLE (D) $22.95
- A0607 CLEAR FOR ACTION (D) $26.95
- A0608 PARIS IN DANGER (D) $26.95
- A0609 GULF STRIKE (D) $22.95
- A0504 GALAXY (D) $16.95
- A0505 ANDROMEDA CONQUEST (D) $16.95
- A0506 COMPUTER STOCKS & BONDS (D) $18.95

EDUCATION

Xerox

- A0412 STICKYBEAR BOP (D) $19.95
- A0413 STICKYBEAR NUMBERS (D) $19.95
- A0414 STICKYBEAR BKST BOUNCE (D) $19.95
- A0415 STICKYBEAR OPPOSITES (D) $19.95
- A0416 STICKYBEAR ABC (D) $19.95
- A0417 STICKYBEAR SHAPES (D) $19.95

BUSINESS

- A0201 ATARI WRITER (C) $39.95
- A0203 VISICALC (D) $29.95
- A0204 HOME FILING MANAGER (D) $19.95
- A0206 FILEWRITER (D) $20.95
- A0207 REPORT WRITER (D) $20.95
- A0208 MENU WRITER (D) $19.95
- A0209 FAMILY FINANCE (D) $19.95
- A0210 HOME INTEGRATOR (D) $19.95
- A0211 SMALL BUS INVENTORY (D) $11.95
- A0212 SALESMAN'S EXPENSES (D) $11.95
- A0214 RETAIL INVOICE (D) $11.95
- A0215 TIMEWISE (D) $14.95
- A0216 PEACHTREE G/L (D) $49.00
- A0217 PEACHTREE A/R (D) $49.00
- A0218 PEACHTREE A/P (D) $49.00
- A0717 SYN CALC (D) $32.95
- A0718 SYN CALC TEMPLATES (D) $14.95
- A0672 APPT PLNR/WKLY SCHEDULE (D) $12.95
- A0673 ACCOUNTS RECEIVABLE (D) $11.95
- A0674 ACCOUNTS PAYABLE (D) $11.95

Synapse

- A0534 ENCOUNTER (D) $14.95
- A0535 BLUE MAX 2001 (D) $18.95
- A0536 QUASIMODO/AIR SUPPORT (D) $16.95
- A0537 NEW YORK CITY/ELECTRICIAN (D) $16.95
- A0538 RAINBOW WALKER/COUNTDOWN (D) $16.95
- A0539 FORT APOCALYPSE (D) $20.95
- A0540 BLUE MAX (D) $20.95
- A0715 MIND WHEEL (D) $25.95
- A0716 ESSEX (D) $25.95

Epyx

- A0520 JUMPMAN (D) $15.95
- A0521 DRAGON RIDERS OF PERN (D) $18.95
- A0522 SUMMER OLY GAMES (D) $24.95
- A0523 PITSTOP II (D) $24.95
- A0524 BALL BLAZER (D) $24.95
- A0525 RESCUE ON FRACTULUS (D) $24.95
- A0693 KORONIS RIFT (D) $24.95
- A0692 THE EIDOLON (D) $24.95

Strategic Simulations, Inc.

- A0601 SHOOTOUT AT OK GALAXY (D) $17.95
- A0602 DNIEPER RIVER LINE (D) $24.95
- A0603 SPACE COWBOY (D) $18.95
- A0526 KNIGHTS OF THE DESERT (D) $24.95
- A0527 FIELD OF FIRE (D) $24.95
- A0528 FORTRESS (D) $22.95
- A0529 COSMIC BALANCE (D) $24.95
- A0530 IMPERIUM GALATUM (D) $24.95
- A0531 RAILS WEST (D) $24.95
- A0532 TIGERS IN THE SNOW (D) $24.95
- A0533 50 MISSION CRUSH (D) $24.95
- A0590 BROADSIDES (D) $24.95
- A0591 COMPUTER QUARTERBACK (D) $24.95
- A0592 COMPUTER AMBUSH (D) $34.95
- A0593 COMPUTER BASEBALL (D) $24.95
- A0712 COLONIAL CONQUEST (D) $24.95
- A0713 COMBAT LEADER (D) $23.95
- A0714 KAMPFGRUPPE (D) $34.95

Atari

- A0420 ATARI MUSIC I (D) $19.95
- A0421 ATARI MUSIC II (D) $19.95
- A0422 INTRO PROG I (T) $14.95
- A0423 INTRO PROG II (T) $14.95
- A0424 INTRO PROG III (T) $14.95
- A0425 ATARI LAB STARTER (C) $44.95
- A0426 ATARI LAB LIGHT MOD (C) $33.95
- A0428 SKYWRITER (C) $16.95
- A0429 CONVERSATIONAL FRENCH (T) $16.95
- A0430 CONVERSATIONAL SPANISH (T) $16.95
- A0431 MY FIRST ALPHABET (D) $16.95
- A0432 SPEED READING (T) $19.95
- A0433 TYPO ATTACK (C) $16.95
- A0435 VERBAL MODULE SAT (D) $29.95
- A0436 SAT SAMPLE PRETEST (D) $17.95
- A0437 MATH MODULE SAT (D) $29.95
- A0438 TOUCH TYPING (T) $14.95
- A0439 JUGGLES RAINBOW (D) $16.95
- A0440 JUGGLES HOUSE (D) $16.95
- A0442 TOUCH TABLET/SOFTWARE $49.00
- A0443 PAINT (D) $19.95
- A0315 PILOT/TURTLE GRAPHICS (C) $29.95
- A0316 LOGO (C) $39.95
- A0318 ASSEMBLER/EDITOR (C) $19.95
- A0319 MACRO ASSEMBLER (C) $19.95

Spinnaker

- A0444 LINKING LOGIC (C) $16.95
- A0445 DANCE FANTASY (C) $16.95
- A0446 MEMORY MANOR (C) $16.95
- A0447 LOGIC LEVELS (C) $16.95
- A0448 KINDERCOMP (D) $16.95
- A0449 FACEMAKER (D) $16.95
- A0450 KIDS ON KEYS (D) $16.95
- A0451 GRANDMAS HOUSE (D) $16.95
- A0452 KIDWRITER (D) $16.95
- A0453 FRACTION FEVER (D) $18.95
- A0454 IN SEARCH AMAZ THING (D) $22.95
- A0455 TRAINS (D) $18.95
- A0456 ALPHABET ZOO (D) $16.95
- A0457 AEROBICS (D) $22.95
- A0710 DELTA DRAWING (C) $16.95
- A0711 ADVENTURE CREATOR (C) $16.95

American Educational Computer

- A0459 VOCABULARY WORD BLDR (D) $16.95
- A0460 GRAMMAR WRK USE SKILLS (D) $16.95
- A0461 WORLD GEOGRAPHY FACTS (D) $16.95
- A0462 SPANISH VOCAB SKILLS (D) $16.95
- A0463 FRENCH VOCAB SKILLS (D) $16.95
- A0464 WORLD HISTORY FACTS (D) $16.95
- A0465 US HISTORY FACTS (D) $16.95
- A0466 US GEOGRAPHY FACTS (D) $16.95
- A0467 US GOVERNMENT FACTS (D) $16.95
- A0468 A PLUS LEARN TO READ (D) $24.95
- A0470 A PLUS READING COMPREHENSION (D) $24.95
- A0471 COMPUTER LEARNING PAD $37.95
- A0418 BIOLOGY FACTS (D) $16.95
- A0493 ELEM SCIENCE 3 & 4 (D) $16.95
- A0494 ELEM SCIENCE 5 & 6 (D) $16.95
- A0495 ELEM SCIENCE 7 & 8 (D) $16.95

DLM

- A0680 SPELLING WIZ (D) $19.95
- A0681 ALIEN ADDITION (D) $19.95
- A0682 METEOR MULTIPLICATION (D) $19.95
- A0683 ALLIGATOR MIX (D) $19.95

Artworx

- A0738 LINKWORD LANGUAGE-SPANISH (D) $17.95
- A0739 LINKWORD LANG-FRENCH (D) $17.95
- A0740 LINKWORD LANG-GERMAN (D) $17.95
- A0741 LINKWORD LANG-ITALIAN (D) $17.95
- A0663 MONKEYMATH (D) $15.95
- A0664 MONKEY NEWS (D) $15.95

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

ORDER TODAY

COMPUTER DIRECT

We Love Our Customers

22292 N. Pepper Rd., Barrington, Ill. 60010

312/382-5050 to order

www.commodore.ca

ATARI®

152K *Lowest Price In The USA!* 152K

Computer System Sale

• Students • Word Processing • Home • Business

LOOK AT ALL YOU GET FOR ONLY **$379** SYSTEM PRICE

LIMITED QUANTITIES

	LIST PRICE	INDIVIDUAL SALE PRICE
① Atari 130XE 152K Computer	$249.00	$134.95
② Atari 1050 127K Disk Drive	299.00	159.95
③ Atari 1027 Letter Quality 20 CPS Printer	299.00	159.95
Atari Writer Plus Word Processer with Spell Checker	59.95	49.95
Atari BASIC Tutorial Manual	16.95	12.95
TOTALS	**$923.90**	**$517.75**

All connecting cables & T.V. interface included.
* Monitors sold separetly.

SAVE OVER $100
All 5 ONLY
$379.00
SYSTEM SALE PRICE

CALL FOR 1027 PRINTER REPLACEMENT OPTIONS

Other Accessories	List	Sale	
☆ 12" Hi Resolution Green Screen Monitor	$199.00	$79.95	Add $9.95 for Connection Cables
☆ 13" Hi Resolution Color Monitor	$399.00	$159.95	Add $10 for UPS

15 DAY FREE TRIAL. We give you 15 days to try out this ATARI COMPUTER SYSTEM!! If it doesn't meet your expectations, just send it back to us prepaid and we will refund your purchase price!! **90 DAY IMMEDIATE REPLACEMENT WARRANTY.** If any of the ATARI COMPUTER SYSTEM equipment or programs fail due to faulty workmanship or material within 90 days of purchase we will replace it IMMEDIATELY with no service charge!!

Best Prices • Over 1000 Programs and 500 Accessories Available • Best Service • One Day Express Mail • Programming Knowledge • Technical Support

Add $25.00 for shipping and handling!!
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery. 2 to 7 days for phone orders. 1 day express mail! We accept Visa and MasterCard. We ship C.O.D. to continental U.S. addresses only. Add $10 more if C.O.D., add $25 if Air Mail.

COMPUTER DIRECT
We Love Our Customers
22292 N. Pepper Rd., Barrington, Ill. 60010
312/382-5050 to order

www.commodore.ca

```
DC 1320 FORI=1800TO1960STEP40
XX 1330 FORJ=0TO2:K=K+1
QS 1340 POKEI+J,NU(TP,K):POKEI
        +J+S,2
GQ 1350 NEXT:NEXT
DX 1360 IFH>9THENRETURN
AK 1370 FORI=1796TO1956STEP40
KD 1380 FORJ=0TO2:K=K+1
XR 1390 POKEI+J,32:POKEI+J+S,2
BS 1400 NEXT:NEXT
RK 1410 RETURN
FK 1420 :
EX 1430 :CHANGE MINUTE NUMBER
XP 1440 :
SQ 1450 T1=T1(M):T2=T2(M)
XS 1460 K=0
PJ 1470 FORI=1806TO1966STEP40
HE 1480 FORJ=0TO2:K=K+1
MH 1490 POKEI+J,NU(T1,K):POKEI
        +J+S,2
EE 1500 NEXT:NEXT
FA 1510 K=0
GD 1520 FORI=1810TO1970STEP40
QM 1530 FORJ=0TO2:K=K+1
FP 1540 POKEI+J,NU(T2,K):POKEI
        +J+S,2
FJ 1550 NEXT:NEXT
KD 1560 RETURN
RE 1570 :
HB 1580 :GET RANDOM TIME
HF 1590 :
HD 1600 G=INT(RND(1)*12)+1:IFG
        =ATHEN1600
CR 1610 A=G
FA 1620 J=INT(RND(1)*12)+1
MQ 1630 IFKK=1THENB=12:RETURN
JA 1640 IFKK=2ANDJ<6THENB=12:R
        ETURN
MC 1650 IFKK=2THENB=6:RETURN
HJ 1660 B=J:RETURN
DM 1670 :
FX 1680 :HICKORY
RQ 1690 :
DC 1700 POKE53281,0:POKE53280,
        11:PRINT"{CLR}"
JJ 1710 FORI=0TO24:POKEI+S,0:N
        EXT
FX 1720 POKES+24,15:POKES+6,24
        0
DE 1730 POKEV,250:POKEV+1,184
JG 1740 POKEV+21,1
SX 1750 PRINT"{3 DOWN}"TAB(30)
        "{RVS}E2}{7 SPACES}"
PC 1760 PRINTTAB(30)"{RVS} E8}
        {2 SPACES}B{2 SPACES}
        E2} "
XM 1770 PRINTTAB(30)"{RVS} E8}
        {2 SPACES}-{2 SPACES}
        E2} "
KC 1780 PRINTTAB(30)"{RVS} E8}
        {2 SPACES}Q*CE2} "
RP 1790 PRINTTAB(30)"{RVS} E8}
        {5 SPACES}E2} "
RQ 1800 PRINTTAB(30)"{RVS}E2}
        {7 SPACES}"
HB 1810 FORI=1TO9
XX 1820 PRINTTAB(30)"{RVS}
        {2 RIGHT}{OFF}{YEL}-
        {2 SPACES}{RVS}E2} "
RF 1830 NEXT
AM 1840 PRINTTAB(30)"{RVS}
        {2 RIGHT}{OFF}{YEL}Q
        {2 RIGHT}{RVS}E2} "
GC 1850 PRINTTAB(30)"{RVS}
        {5 RIGHT} "
DC 1860 PRINTTAB(30)"{RVS}
        {7 SPACES}{WHT}"
RF 1870 PRINT"{HOME}{5 DOWN}"
RX 1880 Z=7:GOSUB1970:PRINT
BK 1890 Z=8:GOSUB1970:PRINT
KS 1900 FORI=186TO50STEP-2:POK
        EV+1,I:NEXT
```

"Hickory, Dickory, Dock" for the Commodore 64 and 128, an amusing educational program for children.

```
DX 1910 Z=4:GOSUB1970:PRINT
KM 1920 POKE1338,160:POKE1257,
        194:POKE1298,206:POKE1
        338+S,15:POKE1298+S,15
FS 1930 POKE1339,160:Z=4:GOSUB
        1970:PRINT
KS 1940 FORI=50TO186STEP2:POKE
        V+1,I:NEXT
DQ 1950 Z=7:GOSUB1970:GOTO250
FQ 1960 :
JE 1970 FORI=1TOZ
SM 1980 READA$,A,B,C:POKES+4,1
        7
XJ 1990 PRINTA$;:POKES+1,A:POK
        ES,B
QB 2000 FORT=1TOC:NEXT:POKES+4
        ,16
EE 2010 NEXT:RETURN
ES 2020 :
BX 2030 :
AS 2040 DATA "HICK",12,32,125,
        "O",12,216,125,"RY, ",
        14,107,125,"DICK",14,1
        07,125
HX 2050 DATA "O",16,47,125,"RY
        , ",18,42,125,"DOCK!",
        19,63,1000
JS 2060 DATA "THE ",12,32,125,
        "MOUSE",12,32,125," ",
        12,216,125,"RAN ",14,1
        07,125
KH 2070 DATA "UP ",14,107,125,
        ,16,47,125,"THE ",18,4
        2,125,"CLOCK.",19,63,1
        000
QP 2080 DATA "THE ",14,107,125
        ,"CLOCK ",19,63,250,"S
        TRUCK ",19,63,125
SE 2090 DATA "ONE,",18,42,250,
        "THE ",18,42,125,"MOUS
        E ",16,47,250
HG 2100 DATA "RAN ",16,47,125,
        "DOWN.",14,107,1000
PC 2110 DATA "HICK",14,107,125
        ,"OR",16,47,124,"Y, ",
        14,107,125,"DICK",12,2
        16,125
JS 2120 DATA "O",12,32,125,"RY
        , ",10,205,125,"DOCK!"
        ,9,159,1000
MG 2130 :
DH 2140 DATA 5,64,84,5,64,84,5
        ,85
AK 2150 DATA 84,0,85,80,0,68,6
        4,0
FJ 2160 DATA 85,64,16,81,65,16
        ,21,1
XX 2170 DATA 16,4,1,31,255,253
        ,15,255
SA 2180 DATA 252,0,255,192,0,2
        55,192,0
EM 2190 DATA 255,192,65,170,12
        8,68,162,128
BD 2200 DATA 80,162,128,0,162,
        128,0,17
GD 2210 DATA 0,0,17,0,1,81,80,
        255
BP 2220 :
HA 2230 :MUSIC SUBS
XR 2240 :
CR 2250 POKES+5,0:POKES+6,2
DQ 2260 POKES+4,129
DC 2270 POKES,100:POKES+1,100
DJ 2280 FORT=1TO10:NEXT
SH 2290 POKES+4,128:RETURN
MA 2300 :
SA 2310 POKES+5,0:POKES+6,240
PB 2320 FORI=1TO10
XC 2330 POKES+4,17
KF 2340 POKES,100:POKES+1,20
XA 2350 FORT=1TO50:NEXT
GA 2360 POKES+4,16
DQ 2370 NEXT:RETURN
CG 2380 :
KX 2390 POKES+6,240:POKES+5,0
QG 2400 POKES+4,17
CR 2410 POKES,108:POKES+1,6
BS 2420 FORT=1TO300:NEXT
BQ 2430 POKES+4,16:RETURN
QM 2440 :
JR 2450 :LEARNING MODULE
GP 2460 :
XJ 2470 PRINT"{2 UP}
        {11 SPACES}"
QX 2480 GOSUB1230:GOSUB1450
MP 2490 K$="":GETK$:IFK$=""THE
        N2490
DD 2500 IFK$="{F5}"THENGOSUB11
        10:GOSUB1230
CA 2510 IFK$="{F7}"THENGOSUB11
        60:GOSUB1450
DA 2520 IFK$="M"THEN260
KE 2530 GOTO2490
PA 2540 :
MG 2550 :ML DATA
RC 2560 :
HK 2570 DATA 173,14,220,41,254
        ,141,14,220
RF 2580 DATA 165,1,41,251,133,
        1,160,0
QJ 2590 DATA 185,0,208,153,0,4
        8,185,0
KA 2600 DATA 209,153,0,49,185,
        0,210,153
PA 2610 DATA 0,50,185,0,211,15
        3,0,51
SF 2620 DATA 185,0,212,153,0,5
        2,185,0
HA 2630 DATA 213,153,0,53,185,
        0,214,153
HS 2640 DATA 0,54,185,0,215,15
        3,0,55
AG 2650 DATA 200,208,205,165,1
        ,9,4,133
CJ 2660 DATA 1,173,14,220,9,1,
        141,14
XG 2670 DATA 220,96
RM 2680 :
AM 2690 :
AS 2700 :CHAR DATA
GM 2710 :
XE 2720 DATA 1,2,4,8,16,32,64,
        128
XG 2730 DATA 128,64,32,16,8,4,
        2,1
JE 2740 DATA 0,0,0,0,0,3,28,22
        4
KF 2750 DATA 0,0,1,14,112,128,
        0,0
FD 2760 DATA 7,56,192,0,0,0,0,
        0
RQ 2770 DATA 192,56,7,0,0,0,0,
        0
```


www.commodore.ca

```
FK 278Ø DATA Ø,Ø,128,112,14,1,
        Ø,Ø
GH 279Ø DATA Ø,Ø,Ø,Ø,Ø,224,28,
        3
GS 28ØØ DATA 8,12,14,255,255,1
        4,12,8
FD 281Ø DATA 24,24,24,24,24,12
        6,6Ø,24
JQ 282Ø DATA 31,15,15,31,57,11
        2,224,192
DH 283Ø DATA 192,224,112,57,31
        ,15,15,31
SC 284Ø DATA 3,7,14,156,248,24
        Ø,24Ø,248
BP 285Ø DATA 248,24Ø,24Ø,248,1
        56,14,7,3
QX 286Ø DATA Ø,Ø,254,62,126,13
        4,2,Ø
RH 287Ø DATA Ø,Ø,127,124,126,9
        7,64,Ø
FS 288Ø DATA Ø,Ø,Ø,2,134,126,6
        2,254
FC 289Ø DATA Ø,Ø,Ø,64,97,126,1
        24,127
EG 29ØØ :
QK 291Ø :HOUR HAND DATA
MJ 292Ø :
CB 293Ø DATA 3,5,6,5,3,4,3,5,6
        ,5,3,4
AG 294Ø DATA 14Ø5,1366,1327
EJ 295Ø DATA 14Ø5,14Ø6,14Ø7,13
        68,1369
KM 296Ø DATA 1445,1446,1447,14
        48,1449,145Ø
CG 297Ø DATA 1485,1486,1487,15
        28,1529
FA 298Ø DATA 1485,1526,1567
JE 299Ø DATA 1484,1524,1564,16
        Ø4
XG 3ØØØ DATA 1483,1522,1561
GD 3Ø1Ø DATA 1483,1482,1481,15
        2Ø,1519
SQ 3Ø2Ø DATA 1443,1442,1441,14
        4Ø,1439,1438
HC 3Ø3Ø DATA 14Ø3,14Ø2,14Ø1,13
        6Ø,1359
QS 3Ø4Ø DATA 14Ø3,1362,1321
SP 3Ø5Ø DATA 14Ø4,1364,1324,12
        84
CD 3Ø6Ø DATA 78,78,111
RE 3Ø7Ø DATA 1Ø3,1Ø4,1Ø5,1Ø3,1
        15
CP 3Ø8Ø DATA 67,67,67,67,67,1Ø
        9
AG 3Ø9Ø DATA 1Ø6,1Ø7,1Ø8,1Ø6,1
        17
QB 31ØØ DATA 77,77,112
FC 311Ø DATA 93,93,93,11Ø
SF 312Ø DATA 78,78,113
MS 313Ø DATA 1Ø5,1Ø4,1Ø3,1Ø5,1
        18
QR 314Ø DATA 67,67,67,67,67,31
PD 315Ø DATA 1Ø8,1Ø7,1Ø6,1Ø8,1
        16
QJ 316Ø DATA 77,77,114
CJ 317Ø DATA 93,93,93,3Ø
EJ 318Ø :
JP 319Ø :NUMBER DATA
GJ 32ØØ :
AX 321Ø DATA +,+,+,+,-,+,+,-,+
        ,+,-,+,+,+,+
SE 322Ø DATA -,+,-,-,+,-,-,+,-
        ,-,+,-,-,+,-
JD 323Ø DATA +,+,+,-,-,+,+,+,+
        ,+,-,+,+,+
KG 324Ø DATA +,+,+,-,-,+,-,+,+
        ,-,-,+,+,+,+
KR 325Ø DATA +,-,+,+,-,+,+,+,+
        ,-,-,+,-,-,+
BG 326Ø DATA +,+,+,+,-,-,+,+,+
        ,-,-,+,+,+,+
MD 327Ø DATA +,+,+,+,-,-,+,+,+
        ,+,-,+,+,+,+
AF 328Ø DATA +,+,+,-,-,+,-,-,+
        ,-,-,+,-,-,+
GD 329Ø DATA +,+,+,+,-,+,+,+,+
        ,+,-,+,+,+,+
JS 33ØØ DATA +,+,+,+,-,+,+,+,+
        ,-,-,+,-,-,+
DX 331Ø :
JJ 332Ø :MORE DATA
GA 333Ø :
HJ 334Ø DATA Ø,5,1,Ø,1,5,2,Ø,2
        ,5,3,Ø
KQ 335Ø DATA 3,5,4,Ø,4,5,5,Ø,5
        ,5,Ø,Ø
```

Program 2: Hickory, Dickory, Dock For Atari 400/800/XL/XE

Version by Kevin Mykytyn, Editorial Programmer

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
PN 1Ø  GOTO 3Ø
IG 2Ø  X=INT(XC-XR*COS(A)):Y=
       INT(YC-YR*SIN(A)):RETU
       RN
PC 3Ø  DIM A$(5),K$(1),TIME$(
       5):OPEN #1,4,8,"K:":PO
       KE 82,Ø
BM 4Ø  GRAPHICS 18:PRINT #6;"
       1. TEST hours":PRINT #
       6:PRINT #6;"2. TEST ho
       urs and{6 SPACES}half
       hours"
AL 5Ø  PRINT #6:PRINT #6;"3.
       TEST five minute
       {4 SPACES}intervals":P
       RINT #6:PRINT #6;"4. P
       RACTICE"
JB 6Ø  PRINT #6:PRINT #6;"5.
       QUIT"
NL 7Ø  GET #1,K:K$=CHR$(K):IF
        K$<"1" OR K$>"5" THEN
        7Ø
OA 8Ø  N=VAL(K$):IF N=5 THEN
       GRAPHICS Ø:END
HN 9Ø  IF N=4 THEN 25Ø
KM 1ØØ GOSUB 29Ø
LH 11Ø HR=INT(RND(1)*12)+1:M
       N=Ø:IF N=2 THEN MN=IN
       T(RND(1)+Ø.5)*3Ø:GOTO
        13Ø
HF 12Ø IF N=3 THEN MN=INT(RN
       D(1)*12)*5
EA 13Ø GOSUB 48Ø:AMN=MN:AHR=
       HR:MN=Ø:HR=12
JN 14Ø GOSUB 44Ø:GOSUB 54Ø:I
       F K$="Q" THEN 4Ø
LP 15Ø IF K$<>CHR$(155) THEN
        14Ø
EF 16Ø POKE 656,Ø:POKE 657,8
       :IF AMN<>MN OR AHR<>H
       R THEN 19Ø
BE 17Ø NR=NR+1:NW=Ø:PRINT "R
       IGHT!";:FOR A=15 TO Ø
        STEP -1
FE 18Ø SOUND 1,121,1Ø,A:SOUN
       D 2,81,1Ø,A:FOR T=1 T
       O 1Ø:NEXT T:NEXT A:GO
       SUB 23Ø:GOTO 21Ø
ON 19Ø MN=AMN:HR=AHR:PRINT "
       WRONG!";:FOR A=1Ø TO
       15Ø:SOUND 1,A,12,15:N
       EXT A:GOSUB 23Ø:NW=NW
       +1:IF NW<>3 THEN 13Ø
```

This version of "Hickory, Dickory, Dock" runs on all Atari 400/800, XL, and XE computers.

```
PC 2ØØ NW=Ø:GOSUB 44Ø:GOSUB
       48Ø:FOR TD=1 TO 15ØØ:
       NEXT TD:GOTO 11Ø
MI 21Ø IF NR=5 THEN NR=Ø:GOS
       UB 59Ø
FP 22Ø GOTO 11Ø
PB 23Ø SOUND 2,Ø,Ø,Ø:SOUND 1
       ,Ø,Ø,Ø:FOR TD=1 TO 6Ø
       Ø:NEXT TD:PRINT "
       {6 LEFT}{7 SPACES}";:
       RETURN
EM 24Ø FOR T=1 TO 15ØØ:NEXT
       T:GOTO 4Ø
LC 25Ø GOSUB 29Ø
GH 26Ø GOSUB 44Ø:GOSUB 48Ø:G
       OSUB 54Ø:IF K$<>"Q" T
       HEN 26Ø
DG 27Ø GOTO 4Ø
PJ 28Ø REM DRAW THE CLOCK
LC 29Ø GRAPHICS 6:SETCOLOR 2
       ,Ø,Ø:GOSUB 41Ø:COLOR
       1:XC=8Ø:YC=38:YR=28:X
       R=YR*1.2:GOSUB 34Ø:XR
       =XR+8:YR=YR+6:YC=YC-4
OH 3ØØ NW=Ø:NR=Ø:POKE 656,Ø:
       POKE 657,25:PRINT "(H
       )OURS{5 SPACES}(M)INU
       TES{5 SPACES}(Q)UIT"
MM 31Ø FOR Q=1 TO 12:A$=STR$
       (Q):A=Q*3Ø+9Ø:GOSUB 2
       Ø:X=INT(X/8):GOSUB 38
       Ø:NEXT Q:XR=XR-1Ø:YR=
       YR-1Ø:YC=YC+4
DN 32Ø HR=12:MN=Ø:OMX=XC:OHX
       =XC:OMY=YC:OHY=YC:RET
       URN
EI 33Ø REM CIRCLE DRAWING SU
       BROUTINE
CL 34Ø DEG :FOR A=Ø TO 36Ø S
       TEP 1Ø:GOSUB 2Ø
CA 35Ø IF A=Ø THEN PLOT X,Y
BL 36Ø DRAWTO X,Y:NEXT A:RET
       URN
IM 37Ø REM CHARACTER PLOTTIN
       G SUBROUTINE
GD 38Ø CHROM=57344:SCREEN=PE
       EK(88)+256*PEEK(89)+X
       +Y*2Ø-(LEN(A$)=2)
ND 39Ø FOR A=1 TO LEN(A$):CS
       =CHROM+ASC(A$(A,A))*8
       -256
KN 4ØØ FOR B=Ø TO 7:POKE SCR
       EEN+B*2Ø,PEEK(CS+B):N
       EXT B:SCREEN=SCREEN+1
       :NEXT A:RETURN
LK 41Ø DL=PEEK(56Ø)+256*PEEK
       (561):POKE DL+85,71:P
       OKE DL+88,2:POKE DL+8
       9,65
AM 42Ø POKE DL+9Ø,PEEK(56Ø):
       POKE DL+91,PEEK(561):
       RETURN
```


www.commodore.ca

```
AH 43Ø REM OUTPUT CLOCK TIME
DF 44Ø COLOR Ø:PLOT XC,YC:DRAWTO OMX,OMY:PLOT XC,YC:DRAWTO OHX,OHY
FC 45Ø COLOR 1:A=MN*6+9Ø:GOSUB 2Ø:PLOT XC,YC:DRAWTO X,Y:XR=XR/2:YR=YR/2:OMX=X:OMY=Y
MC 46Ø A=HR*3Ø+9Ø+MN/2:GOSUB 2Ø:PLOT XC,YC:DRAWTO X,Y:XR=XR*2:YR=YR*2:OHX=X:OHY=Y:RETURN
EJ 47Ø REM OUPUT DIGITAL TIME
NK 48Ø POKE 656,Ø:POKE 657,8:POKE 658,Ø:IF HR<1Ø THEN PRINT " ";
HE 49Ø TIME$=STR$(HR):GOSUB 52Ø:PRINT CHR$(154);
DD 5ØØ IF MN<1Ø THEN PRINT CHR$(144);
CP 51Ø TIME$=STR$(MN)
IP 52Ø FOR A=1 TO LEN(TIME$):PRINT CHR$(ASC(TIME$(A,A))+96);:NEXT A:RETURN
HH 53Ø REM CHANGE TIME ROUTINE
AN 54Ø GET #1,K:K$=CHR$(K):IF K$="M" THEN MN=MN+5:IF MN=6Ø THEN MN=Ø
FN 55Ø IF K$="H" THEN HR=HR+1
JM 56Ø IF HR=13 THEN HR=1
GM 57Ø IF K$="M" OR K$="H" OR K$=CHR$(155) OR K$="Q" THEN RETURN
GP 58Ø GOTO 54Ø
FK 59Ø RESTORE 62Ø
OJ 6ØØ READ NT,L:IF NT=-1 THEN SOUND 1,Ø,Ø,Ø:RETURN
F6 61Ø SOUND 1,NT,1Ø,15:FOR A=1 TO L*4Ø:NEXT A:SOUND 1,Ø,Ø,Ø:GOTO 6ØØ
I6 62Ø DATA 81,1,72,1,81,1,91,1,96,1,1Ø8,1,121,5,96,1,81,1,72,1,81,1,91,1,96,1,1Ø8,1,121,5
KH 63Ø DATA 96,1,1Ø8,2,1Ø8,1,72,3,81,2,81,1,6Ø,3,81,1,72,1,81,1,91,1,96,1,1Ø8,1,121,6,-1,-1
```

Program 3: Hickory, Dickory, Dock For Apple

Version by Tim Victor, Editorial Programmer

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
6F 1ØØ PI = 4 * ATN (1)
ED 11Ø DIM DP(4,7),CX(12),CY(12)
77 12Ø GOSUB 84Ø: GOSUB 96Ø: GOSUB 112Ø: GOSUB 116Ø: POKE 6,Ø: POKE 7,138
25 13Ø IF PEEK (19Ø * 256) = 76 THEN PRINT CHR$ (4);"PR#A$3ØØ": GOTO 15Ø
BA 14Ø POKE 54,Ø: POKE 55,3: CALL 1ØØ2
32 15Ø HGR : GOSUB 51Ø
39 16Ø HGR : HOME : GOSUB 6ØØ
1E 17Ø VTAB 19: HTAB 16: PRINT "1": HTAB 16: PRINT "2"
1C 18Ø FOR I = Ø TO 1: HCOLOR= 5 + I: FOR J = Ø TO 1: HPLOT 18,147 + J + 8 * I TO 52 + 8 * I,147 + J + 8 * I: NEXT : NEXT
B5 19Ø VTAB 21: HTAB 3: PRINT "TO MOVE";: HTAB 14: PRINT "PRESS": HTAB 2: PRINT "THIS HAND";: HTAB 13: PRINT "THIS KEY"
BB 2ØØ CH = 213: FOR CV = 138 TO 15Ø STEP 12: GOSUB 77Ø: NEXT
3C 21Ø FOR I = 1 TO 4:DD(I) = 1Ø: FOR J = 1 TO 7:DP(I,J) = Ø: NEXT : NEXT
E5 22Ø HH = 84:HV = 35:MH = 83:MV = 24
DA 23Ø IF GM = 4 THEN 41Ø
46 24Ø NW = Ø:TH = INT ( RND (1) * 12) + 1:TM = INT ( RND (1) * 6Ø)
1E 25Ø IF GM = 1 THEN TM = Ø
B6 26Ø IF GM = 2 THEN TM = 3Ø * INT (TM / 3Ø)
5C 27Ø IF GM = 3 THEN TM = 5 * INT (TM / 5)
77 28Ø HR = TH:MN = TM: GOSUB 61Ø
```

"Hickory, Dickory, Dock" for Apple computers offers children a fun way to learn about telling time.

```
C5 29Ø HR = 1:MN = Ø
4C 3ØØ VTAB 24: HTAB 2: PRINT "PRESS RETURN TO ANSWER, ESC FOR MENU";
4F 31Ø GOSUB 67Ø
CF 32Ø GOSUB 46Ø: IF A$ = CHR$ (27) THEN 15Ø
33 33Ø IF A$ = CHR$ (13) THEN 35Ø
96 34Ø GOTO 31Ø
23 35Ø IF TH = HR AND TM = MN THEN 39Ø
47 36Ø NW = NW + 1: IF NW < 3 THEN VTAB 24: HTAB 2: PRINT SPC( 38);: HTAB 5: PRINT "THAT IS NOT CORRECT, TRY AGAIN";: FOR I = 1 TO 1ØØØ: NEXT : GOTO 3ØØ
C8 37Ø HR = TH:MN = TM: GOSUB 67Ø: VTAB 24: HTAB 2: PRINT SPC( 38);: HTAB 7: PRINT "THIS IS THE CORRECT ANSWER";
DE 38Ø FOR I = 1 TO 15ØØ: NEXT : GOTO 24Ø
37 39Ø VTAB 24: HTAB 2: PRINT SPC( 38);: HTAB 1Ø: PRINT "CORRECT! GOOD ANSWER";
7F 4ØØ FOR I = 1 TO 1ØØØ: NEXT : GOTO 24Ø
DD 41Ø VTAB 24: HTAB 11: PRINT "PRESS ESC FOR MENU";
B9 42Ø HR = 1:MN = Ø
89 43Ø GOSUB 61Ø: GOSUB 67Ø
D4 44Ø GOSUB 46Ø: IF A$ = CHR$ (27) THEN 15Ø
1C 45Ø GOTO 43Ø
2A 46Ø VTAB 24: HTAB 1: GET A$: IF A$ = CHR$ (27) THEN RETURN
94 47Ø IF A$ = CHR$ (13) THEN RETURN
8E 48Ø IF A$ = "1" THEN HR = HR + 1 - 12 * (HR = 12): RETURN
EF 49Ø IF A$ = "2" THEN MN = MN + 5 - 6Ø * (MN = 55): RETURN
16 5ØØ GOTO 46Ø
56 51Ø TEXT : HOME : VTAB 6: HTAB 8: PRINT "PRESS KEY TO CHOOSE GAME:"
76 52Ø VTAB 1Ø: HTAB 7: PRINT "1: HOURS TEST"
E8 53Ø VTAB 12: HTAB 7: PRINT "2: HOURS AND HALF HOURS TEST"
82 54Ø VTAB 14: HTAB 7: PRINT "3: 5 MINUTE INTERVALS TEST"
38 55Ø VTAB 16: HTAB 7: PRINT "4: PRACTICE"
59 56Ø VTAB 18: HTAB 7: PRINT "5: QUIT"
26 57Ø VTAB 24: GET A$: IF A$ < "1" OR A$ > "5" THEN 57Ø
58 58Ø IF A$ = "5" THEN END
6B 59Ø GM = VAL (A$): RETURN
69 6ØØ FOR I = 1 TO 12: VTAB CY(I): HTAB CX(I): PRINT I: NEXT : RETURN
CE 61Ø DE = 1:HC = 16Ø:DC = INT (HR / 1Ø): IF DC = Ø THEN DC = 1Ø
AC 62Ø IF DC < > DD(1) THEN GOSUB 79Ø
45 63Ø DD(1) = DC:HC = 186:DE = 2:DC = HR - 1Ø * INT (HR / 1Ø): IF DC < > DD(2) THEN GOSUB 79Ø
6F 64Ø DD(2) = DC:HC = 22Ø:DE = 3:DC = INT (MN / 1Ø): IF DC < > DD(3) THEN GOSUB 79Ø
84 65Ø DD(3) = DC:HC = 246:DE = 4:DC = MN - DC * 1Ø: IF DC < > DD(4) THEN GOSUB 79Ø
13 66Ø DD(4) = DC: RETURN
95 67Ø HCOLOR= Ø: GOSUB 7ØØ: GOSUB 72Ø
A7 68Ø GOSUB 69Ø: GOSUB 71Ø: RETURN
E3 69Ø A = (HR / 6 + MN / 36Ø) * PI:HV = 68 - 33 * COS (A):HH = 84 + 44 * SIN (A): HCOLOR= 5
F6 7ØØ HPLOT 84,68 TO HH,HV: HPLOT 83,67 TO HH - 1,HV - 1: RETURN
94 71Ø A = MN / 3Ø * PI:MV = 68 - 44 * COS (A):MH = 83 + 59 * SIN (A): HCOLOR= 6
DC 72Ø HPLOT 84,68 TO MH,MV: HPLOT 83,67 TO MH - 1,MV - 1: RETURN
65 73Ø FOR I = - 2 TO 2: HPLOT CH + I,CV - 4 + ( ABS (I) = 2) TO CH + I,CV + 4 - ( ABS (I) = 2): NEXT
1F 74Ø RETURN
5B 75Ø FOR I = - 1 TO 1: HPLOT CH - 5 - (I = Ø),CV + I TO CH + 5 + (I = Ø),CV + I: NEXT
```


www.commodore.ca

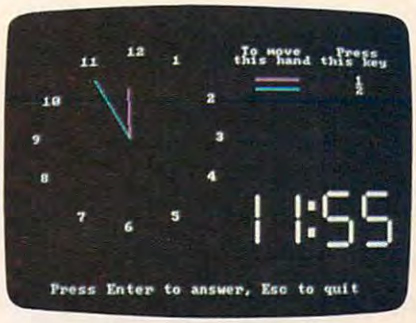
```
23 760 RETURN
27 770 HCOLOR= 2: FOR I = - 2 TO 2: HPLOT CH + I,CV - 2 + ( ABS (I) < 2) TO CH + I ,CV + 2 - ( ABS (I) < 2): NEXT
27 780 RETURN
0B 790 CI = 0:CH = HC + 10: FOR CV = 132 TO 156 STEP 12: GOSUB 830: IF DP(DE,CI) < > PC THEN GOSUB 750:DP(DE,CI) = PC
FF 800 NEXT
7A 810 FOR CV = 138 TO 150 STEP 12: FOR CH = HC + 2 TO HC + 18 STEP 16: GOSUB 830: IF DP(DE,CI) < > PC THEN GOSUB 730:DP(DE,CI) = PC
5E 820 NEXT : NEXT : RETURN
CB 830 CI = CI + 1:PC = VAL ( MID$ (SS$(DC),CI,1)): HCOLOR= 2 * PC: RETURN
F0 840 FOR I = 35456 TO I + 72 STEP 8: POKE I,128: POKE I + 7,128
DB 850 FOR J = 1 TO 6: READ A: POKE I + J,A: NEXT : NEXT : RETURN
90 860 DATA 188,230,246,238,230,188
B1 870 DATA 152,156,152,152,152,188
75 880 DATA 188,230,176,140,230,254
88 890 DATA 188,230,176,224,230,188
9F 900 DATA 176,184,180,254,176,176
5D 910 DATA 254,134,190,224,230,188
79 920 DATA 188,134,190,230,230,188
8A 930 DATA 254,224,176,152,140,140
14 940 DATA 188,230,188,230,230,188
80 950 DATA 188,230,230,252,176,152
E8 960 FOR I = 768 TO I + 87: READ A: POKE I,A: NEXT : RETURN
A4 970 DATA 216,120,133,69,134,70
50 980 DATA 132,71,166,7,10,10
5B 990 DATA 176,4,16,62,48,4
A2 1000 DATA 16,1,232,232,10,134
4A 1010 DATA 27,24,101,6,133,26
8D 1020 DATA 144,2,230,27,165,40
7F 1030 DATA 133,8,165,41,41,3
6B 1040 DATA 5,230,133,9,162,8
28 1050 DATA 160,0,177,26,36,50
73 1060 DATA 48,2,73,127,164,36
57 1070 DATA 145,8,230,26,208,2
AF 1080 DATA 230,27,165,9,24,105
1F 1090 DATA 4,133,9,202,208,226
71 1100 DATA 165,69,166,70,164,71
5C 1110 DATA 88,76,240,253
A1 1120 FOR DC = 0 TO 10: READ SS$(DC): NEXT : RETURN
02 1130 DATA 1011111,0000101,1110110,1110101
14 1140 DATA 0101101,1111001,1111011,1000101
8E 1150 DATA 1111111,1111101,0000000
42 1160 FOR I = 1 TO 12: READ CY(I),CX(I): NEXT : RETURN
AC 1170 DATA 2,18,5,22,9,23
23 1180 DATA 13,22,16,18,17,12
5B 1190 DATA 16,6,13,2,9,1
65 1200 DATA 5,2,2,6,1,12
```

Program 4: Hickory, Dickory, Dock For IBM PC/PCjr

Version by Tim Victor, Editorial Programmer

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
PK 10 KEY OFF:RANDOMIZE TIMER
MN 20 DIM DD(4),DS(4,7),SS$(10),CH(12),CV(12)
HO 30 DIM DH(5),DV(5)
FB 40 PI=4*ATN(1)
HP 50 GOSUB 900:GOSUB 940:GOSUB 840
GA 60 GOSUB 300
KC 70 SCREEN 1
CH 80 GOSUB 530:GOSUB 540:GOSUB 400
CH 90 FOR CP=0 TO 3:DD(CP)=10:FOR SN=1 TO 7:DS(CP,SN)=0:NEXT:NEXT
JM 100 HH=84:HV=36:MH=84:MV=21
BI 110 IF GM<4 THEN 160
DA 120 HR=1:MN=0:GOSUB 700:GOSUB 550
IA 130 LOCATE 25,11:PRINT "Press Esc to quit";
PH 140 GOSUB 500:IF K$=CHR$(27) THEN 60
GC 150 GOSUB 700:GOSUB 550:GOTO 140
PA 160 NC=0
NE 170 NW=0:HR=INT(12*RND(1))+1
LD 180 IF GM=1 THEN MN=0
PM 190 IF GM=2 THEN MN=30*INT(2*RND(1))
OO 200 IF GM=3 THEN MN=5*INT(12*RND(1))
BM 210 GOSUB 700:TH=HR:TM=MN:HR=1:MN=0
LG 220 LOCATE 25,3:PRINT "Press Enter to answer, Esc to quit";
EJ 230 K$="":WHILE K$<>CHR$(27) AND K$<>CHR$(13):GOSUB 550:GOSUB 500:WEND
PL 240 IF K$=CHR$(27) THEN 60
CA 250 LOCATE 25,3:IF HR=TH AND MN=TM THEN 280
IH 260 NW=NW+1:IF NW<3 THEN PRINT SPACE$(8) "That's not correct"SPACE$(8);:NC=0:FOR DLAY=1 TO 500:NEXT:GOTO 220
LB 270 HR=TH:MN=TM:GOSUB 550:PRINT SPACE$(4) "This is the correct answer" SPACE$(4);:FOR I=1 TO 1500:NEXT:GOTO 170
JD 280 PRINT SPACE$(10) "You're right!" SPACE$(11);:NC=NC+1:IF NC=5 THEN NC=0:GOSUB 460
GB 290 FOR DLAY=1 TO 1500:NEXT:GOTO 170
NO 300 SCREEN 0:WIDTH 40:CLS
BB 310 LOCATE 6,8:PRINT "Press key to select game:"
ON 320 LOCATE 8,5:PRINT "1. Hours test"
IB 330 LOCATE 9,5:PRINT "2. Hours and half hours test"
LH 340 LOCATE 10,5:PRINT "3. Five minute intervals test"
DH 350 LOCATE 11,5:PRINT "4. Practice"
NC 360 LOCATE 12,5:PRINT "5. Quit"
NA 370 K$=INPUT$(1):IF K$<"1" OR K$>"5" THEN 370
```



"Hickory, Dickory, Dock" for the IBM PC/PCjr.

```
JN 380 IF K$="5" THEN END
PA 390 GM=VAL(K$):RETURN
IE 400 LOCATE 1,23:PRINT " To move    Press";
DG 410 LOCATE 2,23:PRINT "this hand and this key";
CG 420 LOCATE 4,37:PRINT "1":LOCATE 5,37:PRINT "2"
DF 430 LINE (196,27)-(235,28),2,BF
PA 440 LINE (196,35)-(235,36),1,BF
NI 450 RETURN
CP 460 PLAY "mnt180o3l8gggaaal2gl4n0l8gl4el8el4fl8fl2el4n0"
LC 470 PLAY "l8el4cl8cl4el8el4dl8dl4a."
KA 480 PLAY "mll8gagfedl4c."
NA 490 RETURN
FI 500 K$=INPUT$(1):IF K$="1" THEN HR=HR+1+12*(HR=12)
EA 510 IF K$="2" THEN MN=MN+5+60*(MN=55)
MD 520 RETURN
OM 530 FOR I=1 TO 12:LOCATE CV(I),CH(I):PRINT MID$(STR$(I),2):NEXT:RETURN
BA 540 LH=241:FOR LV=128 TO 149 STEP 21:LINE (LH,LV)-(LH+5,LV+6),3,BF:NEXT:RETURN
QK 550 LINE (84,76)-(HH,HV),0
DN 560 LINE (83,75)-(HH-1,HV-1),0
NL 570 LINE (84,76)-(MH,MV),0
BI 580 LINE (83,75)-(MH-1,MV-1),0
IB 590 HH=84+48*SIN(PI*(HR/6+MN/360))
FM 600 HV=76-40*COS(PI*(HR/6+MN/360))
CH 610 LINE (84,76)-(HH,HV),2
GC 620 LINE (83,75)-(HH-1,HV-1),2
DO 630 MH=84+63*SIN(PI*MN/30)
IJ 640 MV=76-55*COS(PI*MN/30)
OC 650 LINE (84,76)-(MH,MV),1
CD 660 LINE (83,75)-(MH-1,MV-1),1
NO 670 RETURN
HO 680 LINE (83,75)-(HH-1,HV-1),2
EK 690 MH=84+63*SIN(PI*MN/30)
MK 700 CP=0:DC=INT(HR/10):IF DC=0 THEN DC=10
MK 710 GOSUB 760
DL 720 CP=1:DC=HR-10*INT(HR/10):GOSUB 760
CJ 730 CP=2:DC=INT(MN/10):GOSUB 760
MM 740 CP=3:DC=MN-10*DC:GOSUB 760
NL 750 RETURN
IP 760 IF DD(CP)=DC THEN RETURN ELSE DD(CP)=DC
```

www.commodore.ca

```
LL 770 SN=Ø:FOR SV=Ø TO 4Ø STEP
       2Ø:GOSUB 81Ø:IF DS(CP,SN)
       <>HC THEN GOSUB 82Ø:DS(CP
       ,SN)=HC
OD 780 NEXT
DM 790 FOR SV=Ø TO 2Ø STEP 2Ø:FO
       R SH=Ø TO 24 STEP 24:GOSU
       B 81Ø:IF DS(CP,SN)<>HC TH
       EN GOSUB 83Ø:DS(CP,SN)=HC
FL 8ØØ NEXT:NEXT:RETURN
FP 81Ø SN=SN+1:HC=(MID$(SS$(DC),
       SN,1)="1"):RETURN
JC 82Ø PUT (174+CP*36-12*(CP>1),
       12Ø+SV),DH,XOR:RETURN
ID 83Ø PUT (17Ø+CP*36-12*(CP>1)+
       SH,123+SV),DV,XOR:RETURN
LM 84Ø SCREEN 1:CLS
GG 85Ø FOR LV=Ø TO 3:LINE (1+(LV
       =1 OR LV=2),LV)-(18-(LV=1
        OR LV=2),LV),3:NEXT
LG 86Ø GET (Ø,Ø)-(19,3),DH:PUT (
       Ø,Ø),DH,XOR
LO 87Ø FOR LH=Ø TO 3:LINE (LH,1+
       (LH=1 OR LH=2))-(LH,16-(L
       H=1 OR LH=2)),3:NEXT
ON 88Ø GET (Ø,Ø)-(3,17),DV:PUT (
       Ø,Ø),DV,XOR
NE 89Ø RETURN
HB 9ØØ FOR DC=Ø TO 1Ø:READ SS$(D
       C):NEXT:RETURN
BI 91Ø DATA "1Ø11111","ØØØØ1Ø1",
       "111Ø11Ø","111Ø1Ø1"
QO 92Ø DATA "Ø1Ø11Ø1","1111ØØ1",
       "1111Ø11","1ØØØ1Ø1"
NN 93Ø DATA "1111111","11111Ø1",
       "ØØØØØØØ"
OD 94Ø FOR I=1 TO 12:READ CV(I),
       CH(I):NEXT:RETURN
NN 95Ø DATA 2,16,6,2Ø,1Ø,21
IG 96Ø DATA 14,2Ø,18,16,19,11
FD 97Ø DATA 18,6,14,2,1Ø,1
MG 98Ø DATA 6,2,2,6,1,11
```

Program 5: Hickory, Dickory, Dock For Atari ST

Version by Kevin Mykytyn, Editorial Programmer

```
10 randomize 0:goto 30
20 z=a*.0175:x=int(xc-xr*cos(z)):y=int(y
   c-yr*sin(z)):return
30 ch=16:gosub 470:openw 2:clearw 2:ful
   lw 2:ohx=130:omx=130:ohy=75:omy
   =75
40 nr=0:xr=70:yr=60:xc=130:yc=75:got
   oxy 0,5:color 6:print " 1. Test - hours"
50 print:print " 2. Test - hours and half ho
   urs"
60 print:print " 3. Test - five minute inter
   vals"
```

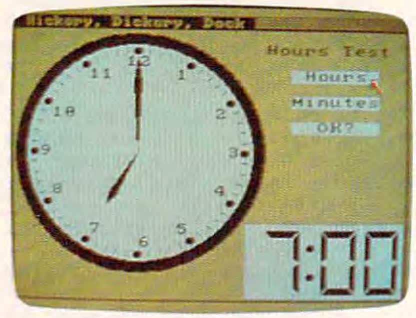
The Atari ST version of "Hickory, Dickory, Dock."

```
70 print:print " 4. Practice":print:print " 5
   . Quit"
80 k=inp(2):if k<49 or k>53 then 80
90 if k=53 then ch=6:gosub 470:end els
   e if k=52 then 250
100 n=k-48:gosub 280
110 hr=int(rnd(1)*12)+1:mn=0:if n=2 th
   en mn=int(rnd(1)+.5)*30:goto 130
120 if n=3 then mn=int(rnd(1)*12)*5
130 gosub 380:amn=mn:ahr=hr:mn=0:h
   r=12
140 gosub 340:gosub 420:if k=113 then 30
150 if k<>13 then 140
160 color 2:if amn<>mn or ahr<>hr the
   n 190
170 nr=nr+1:gotoxy 26,15:print "Right!":
   b=1:c=5:gosub 560
180 nw=0:gosub 240:goto 220
190 gotoxy 26,15:print "Wrong!":b=5:c=1
   :gosub 560
200 gosub 240:hr=ahr:mn=amn:nw=nw
   +1:if nw<>3 then 130
210 gosub 340:for td=1 to 4000:next:nw=
   0:goto 230
220 if nr=5 then nr=0:gosub 500
230 for t=1 to 500:next:goto 110
240 for td=1 to 300:next:sound 1,0:gotox
   y 26,15:print "            ":return
250 gosub 280
260 gosub 340:gosub 380:gosub 420:if k=1
   13 then 30 else 260
270 ' draw the clock
280 clearw 2:color 1
290 ch=6:gosub 470:gotoxy 3,17:print "(H)
   ours      (M)inutes      (Q)uit";
300 color 6:ch=16:gosub 470:nr=0:nw=0:
   for q=1 to 12:a=q*30+90:gosub 20
310 x=int(x/8):y=int(y/8):if q=11 or q=
   12 then x=x-1
320 gotoxy x,y:print q:next
330 xr=xr-7:yr=yr-7:xc=xc+28:yc=yc+
   8:mn=0:hr=12:return
340 color 1,1,0:linef xc,yc,omx,omy:linef xc
   ,yc,ohx,ohy
350 color 1,1,2:a=mn*6+90:gosub 20:line
   f xc,yc,x,y:xr=xr/2:yr=yr/2
360 color 1,1,4:omx=x:omy=y:a=hr*30+
   90+mn/2:gosub 20:linef xc,yc,x,y
370 xr=xr*2:yr=yr*2:ohx=x:ohy=y:retu
   rn
380 color 5:gotoxy 2,15:if hr<10 then prin
   t " ";
390 q=hr:gosub 410:print q$;":";:if mn<1
   0 then print "0";
400 q=mn:gosub 410:print q$:return
410 q$=right$(str$(q),len(str$(q))-1):return
420 k=inp(2):if k=109 then mn=mn+5:i
   f mn=60 then mn=0
430 if k=104 then hr=hr+1
440 if hr=13 then hr=1
450 if k=104 or k=109 or k=13 or k=11
   3 then return
460 goto 420
470 poke contrl,12:poke contrl+2,1:pok
   e contrl+6,0
480 poke ptsin,0:poke ptsin+2,ch
490 vdisys (0):return
500 restore 520:for nt=1 to 29:read a,b,c
510 sound 1,15,a,b,c*7:next:sound 1,0:retu
   rn
520 data 8,5,1,10,5,1,8,5,1,6,5,1,5,5,1,3,5,1,1,5
   ,5
530 data 5,5,1,8,5,1,10,5,1,8,5,1,6,5,1,5,5,1,3,5
   ,1,1,5,5
540 data 5,5,1,3,5,2,3,5,1,10,5,3,8,5,2,8,5,1,1,6
   ,3
550 data 8,5,1,10,5,1,8,5,1,6,5,1,5,5,1,3,5,1,1,5
   ,6
560 for a=b to c step 2*sgn(c-b):sound 1,1
   5,a,6
570 wave 1,1,14,5,5:next:return
```

Program 6: Hickory, Dickory, Dock For Amiga

Version by John Krause

```
CLEAR,30000←
GOSUB init←
loop:←
MENU 1,0,1←
IF MENU(0)=1 THEN ON MENU(1) GOS
UB practice,hours,half,five,quit←
IF p THEN GOSUB SetClock←
GOTO loop←
practice:←
COLOR 3,0:LOCATE 2,17:PRINT " Practi
ce "←
SAY TRANSLATE$("praktis.")←
p=1:hour=0:minute=0:GOSUB Updat
eAnalog←
HourDigital=0:MinuteDigital=0:GOSU
B UpdateDigital←
RETURN←
hours:←
p=0:COLOR 3,0:LOCATE 2,17:PRINT '
' Hours Test "←
SAY TRANSLATE$("hours test.")←
FOR count=1 TO 5←
MinuteDigital=0:HourDigital=INT(R
ND*12)*12←
GOSUB GetAnswer←
NEXT←
GOSUB music←
RETURN←
half:←
p=0:COLOR 3,0:LOCATE 2,17:PRINT "H
alf Hours Test"←
SAY TRANSLATE$("half hours test.")←
FOR count=1 TO 5←
MinuteDigital=CINT(RND)*72:HourDi
gital=INT(RND*12)*12←
GOSUB GetAnswer←
NEXT←
GOSUB music←
RETURN←
five:←
p=0:COLOR 3,0:LOCATE 2,17:PRINT
" 5 Minute Test"←
SAY TRANSLATE$("five minut test.")←
FOR count=1 TO 5←
MinuteDigital=INT(RND*12)*12:Hour
Digital=INT(RND*12)*12←
GOSUB GetAnswer←
NEXT←
GOSUB music←
RETURN←
```



Speech synthesis and mouse control are featured in the Amiga version of "Hickory, Dickory, Dock."

www.commodore.ca

```
quit:←
SYSTEM←
SetClock:←
answer=0←
IF MOUSE(0)=1 THEN←
IF MOUSE(3)>220 AND MOUSE(3)<29
0 THEN←
IF MOUSE(4)>27 AND MOUSE(4)<36 TH
EN←
GOSUB IncHour←
IF p THEN GOSUB IncHourDigital←
END IF←
IF MOUSE(4)>45 AND MOUSE(4)<54 TH
EN←
GOSUB IncMinute←
IF p THEN GOSUB IncMinuteDigital←
END IF←
IF MOUSE(4)>63 AND MOUSE(4)<72 AN
D p=0 THEN←
COLOR 1,0←
IF INT(hour/12)=HourDigital/12 AN
D minute=MinuteDigital THEN←
LOCATE 10,23:PRINT "Correct!"←
answer=2:SAY TRANSLATE$("corekt.'
')←
ELSE←
LOCATE 10,23:PRINT "Wrong!"←
answer=1:SAY TRANSLATE$("wrong.'
')←
END IF←
FOR i=0 TO 2000:NEXT←
LOCATE 10,23:PRINT SPACE$(9)←
END IF←
END IF←
END IF←
RETURN←
GetAnswer:←
MENU 1,0,0←
GOSUB UpdateDigital:wrong=0←
loop1: GOSUB SetClock←
IF answer=0 THEN loop1←
IF answer=2 THEN←
RETURN←
ELSE←
wrong=wrong+1←
IF wrong<3 THEN←
GOTO loop1←
ELSE←
WHILE MinuteDigital<>minute:GOSU
B IncMinute:WEND←
WHILE HourDigital/12<>INT(hour/12)
:GOSUB IncHour:WEND←
count=count-1←
FOR t=0 TO 4000:NEXT←
END IF←
END IF←
RETURN←
music:←
FOR i=0 TO 27:SOUND f(i),t(i):SOUND 0,
1:NEXT:RETURN←
IncHourDigital:←
HourDigital=(OldHourDigital+12) MO
D 144←
GOSUB UpdateDigital←
RETURN←
IncMinuteDigital:←
MinuteDigital=(OldMinuteDigital+12
) MOD 144←
HourDigital=(OldHourDigital+1) MO
D 144←
GOSUB UpdateDigital←
RETURN←
IncHour:←
hour=(OldHour+12) MOD 144←
GOSUB UpdateAnalog←
RETURN←
IncMinute:←
minute=(OldMinute+12) MOD 144←
hour=(OldHour+1) MOD 144←
GOSUB UpdateAnalog←
RETURN←
number:←
COLOR 1,2←
LOCATE r(i),c(i)←
IF i>9 THEN PRINT "1";←
PRINT CHR$(48+(i MOD 10))←
RETURN←
UpdateDigital:←
GOSUB DrawDigital←
OldHourDigital=HourDigital:OldMinu
teDigital=MinuteDigital←
GOSUB DrawDigital←
RETURN←
DrawDigital:←
IF OldHourDigital<12 THEN OldHourDi
gital=OldHourDigital+144←
IF OldHourDigital>119 THEN PUT (165,
139),d(0,1)←
PUT (200,139),d(0,(OldHourDigital\12
) MOD 10)←
PUT (245,139),d(0,INT(OldMinuteDigit
al*5/120))←
PUT (280,139),d(0,(OldMinuteDigital*5
\12) MOD 10)←
RETURN←
ClrAnalog:←
GOSUB MinuteHand←
COLOR 2:AREAFILL←
GOSUB HourHand←
AREAFILL←
i=CINT(OldMinute/12):IF i=0 THEN i=
12←
GOSUB number←
RETURN←
UpdateAnalog:←
GOSUB ClrAnalog←
OldMinute=minute:GOSUB MinuteHan
d←
COLOR 1:AREAFILL←
OldHour=hour:GOSUB HourHand←
AREAFILL←
RETURN←
MinuteHand:←
AREA (cx,cy)←
AREA (cx+60*x((OldMinute+143) MO
D 144),cy+60*y((OldMinute+143) MO
D 144))←
AREA (cx+80*x(OldMinute MOD 144),c
y+80*y(OldMinute MOD 144))←
AREA (cx+60*x((OldMinute+1) MO
D 144),cy+60*y((OldMinute+1) MOD 14
4))←
RETURN←
HourHand:←
AREA (cx,cy)←
AREA (cx+40*x((OldHour+142) MO
D 144),cy+40*y((OldHour+142) MO
D 144))←
AREA (cx+50*x(OldHour MOD 144),cy
+50*y(OldHour MOD 144))←
AREA (cx+40*x((OldHour+2) MOD 144
),cy+40*y((OldHour+2) MOD 144))←
RETURN←
init:←
SAY ""←
SCREEN 2,320,200,2,1←
WINDOW 2," Hickory, Dickory, Dock ",,
2,2←
PALETTE 0,.6,.5,.3←
PALETTE 1,0,0,0←
PALETTE 2,.7,.7,.7←
PALETTE 3,0,0,0←
DIM s(300,6),d(300,9),x(143),y(143),r(1
2),c(12),f(27),t(27)←
MENU 1,0,1,"Test "←
MENU 1,1,1,"Practice "←
MENU 1,2,1,"Hours "←
MENU 1,3,1,"Half hours"←
MENU 1,4,1,"5 minute "←
MENU 1,5,1,"Quit "←
MENU 2,0,0,"":MENU 3,0,0,"":MENU 4,
0,0,""←
RANDOMIZE TIMER←
pi=4*ATN(1):p=1←
FOR i=0 TO 143←
x(i)=COS(pi*(i/72-.5))←
y(i)=SIN(pi*(i/72-.5))*.84←
NEXT←
FOR i=0 TO 6←
CLS:READ k←
FOR j=1 TO k:READ x,y:AREA (x,y):NE
XT←
AREAFILL:GET (0,0)-(32,48),s(0,i)←
NEXT←
FOR j=0 TO 9←
CLS:READ a$←
FOR i=1 TO 7←
IF MID$(a$,i,1)="1" THEN PUT (0,0),s(
0,i-1)←
NEXT←
GET (0,0)-(32,48),d(0,j)←
NEXT←
CLS:ERASE s←
FOR i=1 TO 12:READ r(i),c(i):NEXT←
FOR i=0 TO 27:READ f(i),t(i):NEXT←
cx=100:cy=85←
CIRCLE (cx,cy),100,1:PAINT (cx,cy),1←
CIRCLE (cx,cy),90,2:PAINT (cx,cy),2←
COLOR 1,2←
FOR i=1 TO 12:GOSUB number:NEXT←
FOR i=0 TO 143 STEP 12←
CIRCLE (cx+84*x(i),cy+84*y(i)),3←
PAINT (cx+84*x(i),cy+84*y(i))←
NEXT←
FOR i=0 TO 59←
dx=COS(pi*(i/30-.5)):dy=SIN(pi*(i/3
0-.5))←
LINE (cx+82*dx,cy+69*dy)-STEP(4*dx,
3*dy)←
NEXT←
LINE (185,135)-(320,200),2,bf←
CIRCLE (236,151),3:PAINT (236,151)←
CIRCLE (236,171),3:PAINT (236,171)←
GOSUB DrawDigital←
LOCATE 4,23:PRINT " Hours "←
LOCATE 6,23:PRINT "Minutes"←
LOCATE 8,23:PRINT " OK? "←
SAY TRANSLATE$("welcome to hickor
y dickory dock.")←
GOSUB practice←
RETURN←
DATA 4,0,1,0,21,4,19,4,5←
DATA 4,1,0,26,0,22,4,5,4←
DATA 4,27,1,27,21,23,19,23,5←
DATA 4,27,23,27,43,23,39,23,25←
DATA 4,26,44,1,44,5,40,22,40←
DATA 4,0,43,0,23,4,25,4,39←
DATA 6,1,22,6,20,22,20,26,22,21,24,5,2
4←
DATA 1111110,0011000,0110111,0111
101,1011001←
DATA 1101101,1101111,0111000,1111
111,1111101←
DATA 4,14,7,17,10,18,13,17,16,14,17,1
1←
DATA 16,7,13,4,10,3,7,4,4,7,3,10←
DATA 783.99,2,783.99,2,783.99,2,880,2
,880,2,880,2,783.99,11←
DATA 783.99,2,659.26,5,659.26,2,698.4
6,5,698.46,2,659.26,11←
DATA 659.26,2,523.25,5,523.25,2,659.2
6,5,659.26,2,587.33,5,587.33,2,880,8←
DATA 783.99,2,880,2,783.99,2,698.46,2
,659.26,2,587.33,2,523.25,11←
```
©

www.commodore.ca

Philips CD-ROM And *The Electronic Encyclopedia* For IBM

Tony Roberts, Production Director

Requirements: IBM PC with at least 256K RAM. Versions for other personal computers expected soon.

Recent years have unleashed an information explosion—an uncoordinated, unmanaged proliferation of data. New developments, however, indicate that we stand poised to harness this data and place the immense power of the information age at the fingertips of anyone with access to a personal computer.

The harbinger of this new era is the hardware/software combination of a CD-ROM (Compact Disc–Read Only Memory) player and a compact disc containing up to 600 megabytes of digitized data. The first such combination available is the Philips CM100 player and Grolier's *The Electronic Encyclopedia.* Available now for the IBM PC and compatibles, this package is an exciting look at the future of information retrieval.

The Electronic Encyclopedia is a 20-volume, nine-million-word encyclopedia with cross-referenced index encoded on a five-inch plastic disc (with about two-thirds of the disc to spare). Using this disc, a CD-ROM player, and an IBM PC running Activenture Corporation's *Knowledge Retrieval System,* it's possible to access any article in the encyclopedia in seconds. In fact, you can find every occurrence of any keyword throughout the entire encyclopedia.

The System

The Philips CM100 system consists of the CD-ROM drive itself, a tan box about 14 inches wide, 6 inches high, and 10 inches deep; an interface card, which occupies one of the full slots in the IBM PC; and a connecting cable. Everything can be set up in a matter of minutes. The disc player, incidentally, cannot play audio compact discs, although the technology is quite similar.

Activenture's *Knowledge Retrieval System* is the link between you and the CD-ROM. It's so simple to operate that the 94-page instruction manual is practically superfluous.

The left quarter of the screen shows the commands available by pressing one of the ten function keys on the IBM. For instance, the F1 key, labeled About Keys, displays a map showing you where you are in the program and offers help on any of the functions available. The remainder of the screen displays options for your searches and the articles you call up.

A Simple Search

After viewing the title screen and pressing the function key labeled Word Search, you're ready to begin sifting through the encyclopedia. Let's say you're looking for references to the subject "information age."

Following the onscreen prompts, you can ask the computer to search for the word *information* along with the word *age* occurring anywhere in the same article. Within seconds, a message is flashed on the screen that 1,472 occurrences of the word *information* have been found. Then the display indicates that 3,221 occurrences of the word *age* have been located. Putting the two words together, the program finally reports that there are 228 occurrences of the words *information* and *age* in 117 articles.

That's probably more than you bargained for. To narrow things down, you can instruct the computer to look only for occurrences of the two words within the same paragraph. Seconds later, the program reports finding 33 occurrences in 15 articles, including pieces on animal behavior, insurance, census, and poison.

Narrowing the search parameters further, you can specify that *information* and *age* must appear in exact order: that is, "information age." This time, the program finally reports that there are two occurrences in one article. When you press F4, Show Titles, the program loads and displays the titles of the articles located during the search. In this case, the article you're probably interested in is "Information Science."

Pressing F2 loads the article and displays the paragraph containing the first reference to the search words. The search words are highlighted throughout the article, so it's easy to scan it using the function keys labeled Next Page and Previous Page to find your information quickly. As it turns out, the "Information Science" article has only one relevant paragraph relating to "information age" (the other occurrence of the phrase appears in the bibliography at the end of the article). To print out a hardcopy, you can press F7, Make Copy. The result is the following paragraph:

> It is common to speak of the present as the Information Age, or to refer to the information explosion. About 50 percent of all workers in the United States today are in some way involved in information processing. Many people do not receive the right information at the right time, however, because they are not aware the information exists, because they do not know where to look for it, or because it is buried in a mass of extraneous information and is difficult to find.

Plenty Of Options

There are many options available after choosing Make Copy. You can copy the article or parts of it to a printer, or save it to disk. You can select printer margins, line spacing, hyphenation, and justification, if you like. If you're saving to disk, you can save it in ASCII text format, in a *WordStar*-compatible format, or in a PRINT format.

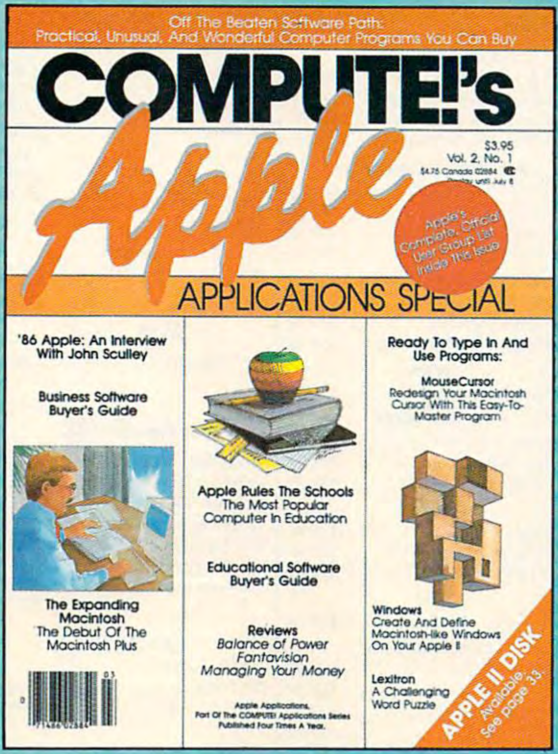
Only A Beginning

This description of using *The Electronic Encyclopedia,* and the encyclopedia itself, are only beginnings. Much more complex searches are possible by using wildcards and negating certain words. For example, you could search for *horse* but not *iron* to eliminate articles about steam engines from your research on equestrians. Another timesaving feature is the Outline option. If you find yourself mired in a complex article,

www.commodore.ca

COMPUTE!'s *All New* Apple Applications Special

COMPUTE!'s latest *Apple Applications Special* features in-depth articles and interviews, all the inside news about Apple, clearly written tutorials, software buyer's guides, new product information, and valuable ready-to-type-in programs for all Apple users.



Apple owners find these special Apple issues the most understandable, complete, and valuable resources available today.

PLUS

All the programs in *COMPUTE!'s Apple Applications Special* are also available on a timesaving disk, ready to run on your Apple II, II+, IIe, and IIc. The *Disk* costs only $12.95 (plus $2.00 shipping and handling) and gives you immediate access to all the great programs in this special issue.

Look for the Spring/Summer 1986 issue of *COMPUTE!'s Apple Applications Special* on sale where you buy other COMPUTE! publications, or order directly from COMPUTE!. This special issue goes on sale April 8, 1986.

Send in the attached order card or call toll free 800-346-6767 (in NY call 212-887-8525).

Features

• Business Applications

'86 Apple: An Interview with John Scully

A wide-ranging interview with the president of Apple. The company's plans for the coming year, its markets, the new Macintosh, and the viability of the Apple II.

Business Software Buyer's Guide

A buyer's guide to the newest Apple II and Macintosh word processors, databases, spreadsheets, and more.

The Expanding Mac

• Education

Apple Rules the Schools

Why does Apple have a lock on educational computing? Comments from teachers, administrators, and Apple.

Computers and the Humanities

Educational Software Buyer's Guide

• The Expanding Apple

It's New II

A multitude of new hardware and software for the Apple II line—from color printers to Mac-like software—is evaluated.

Weirdware: Off the Beaten Software Path

Weirdware—out of the ordinary software—can turn the Apple II or Macintosh into a telescope, astrological fortune-teller, baby evaluator, and much, much more.

MacAdds: More for the Macintosh

Applications

• Utilities and Tutorials

Windows

Create Macintosh-style windows on any Apple II-series computer. Set window size, open, close, and retrieve information.

Mouse Cursor

A Macintosh BASIC utility for altering the mouse pointer. Design data can be saved, then used in other BASIC programs.

Your Personal Ledger

A complete personal financial application for tracking expenses, income, and assets. Easy to use, and packed with features from report generation to customized category codes.

Personal Publishing With Your Macintosh

Tutorial and guide to using such software as *MacPaint* and *MacWrite* to customize letterheads, cards, banners, and more.

Keynote

• Education and Recreation

Lexitron

Entertaining word game where players try to beat the clock while finding as many hidden words as possible.

Backgammon

Play the computer in this classic game. This version observes all the rules of standard backgammon.

Apple Automatic Proofreader

COMPUTE!™ Publications, Inc. abc

Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor, New York, NY 10019
Publishers of COMPUTE!, COMPUTE!'s Gazette, COMPUTE!'s Gazette Disk, COMPUTE! Books, and COMPUTE!'s Apple Applications

www.commodore.ca

COMPUTER MAIL ORDER

THE CMO ADVANTAGE

- THE BEST PRICES!
- Next day shipping on all in stock items.
- Free easy access order inquiry.
- Orders from outside Pennsylvania save state sales tax.
- Free technical support with our factory trained technical staff.
- There is no limit and no deposit on C.O.D. orders.
- There's no extra charge for using your MasterCard or Visa. Your card is not charged until we ship.
- No waiting period for cashiers checks.
- We accept purchase orders from qualified corporations. Subject to approval.
- Educational discounts available to qualified institutions.
- FREE CATALOG MEMBERSHIP.

ORDER LINE

CALL TOLL-FREE
1-800-233-8950
Educational Institutions Call Toll-Free
1-800-221-4283
CUSTOMER SERVICE & TECH SUPPORT
1-717-327-1450
DEPT. A205

MAILING ADDRESS

Computer Mail Order
DEPT. A205
477 East Third Street
Williamsport, PA 17701

MEMBER DIRECT MARKETING ASSOCIATION

CREDIT CARDS

VISA MasterCard American Express Diners Club

SHIPPING

Add 3%, minimum $7.00 shipping and handling on all orders. Larger shipments may require additional charges.

All items subject to availability and price change.

Returned shipments may be subject to a restocking fee.

CANADIAN ORDERS

1-800-268-3974
Ontario/Quebec
1-800-268-4559
Other Provinces
1-416-828-0866
In Toronto

TELEX: 06-218960

2505 Dunwin Drive,
Mississauga, Ontario
Canada L5L1T1

All prices shown are for U.S.A. orders.
Call The Canadian Office for Canadian prices.

HOME COMPUTERS

APPLE

- APPLE IIe ... CALL
- APPLE IIc ... CALL
- IIc LCD Display ... CALL

ATARI

- **65XE (64K) ... CALL**
- **130XE (128K) ... CALL**
- **520ST (512K) ... CALL**
- 800XL 64K ... CALL
- 1010 Recorder ... $49.99
- 1050 Disk Drive ... CALL
- 1020 Printer ... $29.99
- 1027 Letter Quality Printer ... $129.00
- 1030 Direct Connect Modem ... $59.99
- Comrex 220 Atari ... $89.99

COMMODORE

C64 Package

- C64, C1541, C1802 ... $559.00
- C128 Computer ... $269.00
- C1571 (Disk Drive for C128) ... $249.00
- C1902 (RGB 13" Monitor for C128 ... CALL
- C1670 (Modem for C128) ... $179.00
- C1530 Datasette ... $39.99
- C1660 Auto Modem ... $59.99
- DPS 1101 Daisy Printer ... $339.00
- Comrex 220 (C64 Interface) ... $89.99
- Xetec SuperGraphix 8K ... $69.99

GRAPHICS

Polaroid

- Palette ... $1399.00
- Power Processor ... $229.00
- Illuminated Slide Mounter ... $39.99
- Polacolor 2 Pack film ... $18.99

ACCESSORIES

CURTIS POWER STRIPS

- Diamond SP-1 ... $32.99
- Emerald SP-2 ... $39.99
- Sapphire SPF-1 ... $52.99
- Ruby SPF-2 ... $59.99
- Safe Strip SP-3 ... $19.99

KENSINGTON

- MasterPiece ... $99.99
- MasterPiece Plus ... $119.00

PORTABLE COMPUTERS

HEWLETT PACKARD

- 41CV ... $139.00
- 41CX ... $199.00
- HP 11C ... $49.99
- HP 12C ... $75.99
- HP 15C ... $75.99
- HP 16C ... $89.99
- HPIL Module ... $98.99
- HPIL Cassette or Printer ... $359.99
- Card Reader ... $143.99
- Extended Function Module ... $63.99
- Time Module ... $63.99

We stock the full line of HP calculator products

NEC

- PC-8401 LS ... CALL
- PC-8201 Portable Computer ... $339.00
- PC-8231 Disk Drive ... $599.00
- PC-8221A Thermal Printers ... $149.00
- PC-8281A Data Recorder ... $99.99
- PC-8201-06 8K RAM ... $79.99

SHARP

- PC-1350 ... $149.00
- PC-1261 ... $149.00
- PC-1500A ... $169.00
- PC-1250A ... $89.99
- CE-125 Printer/Cassette ... $129.00
- CE-150 Color Printer Cassette ... $149.00
- CE-161 16K RAM ... $129.00

SOFTWARE

ARTWORX

- Bridger 4.0 (All) ... $17.99
- Strip Poker (All) ... $19.99

ATARI

ROMS FOR ATARI ONLY

- 8036 Atari Writer ... $24.99
- Star Raiders ... $4.99
- Missile Command ... $4.99
- Defender ... $4.99
- Galaxian ... $4.99
- Asteroids ... $4.99
- Centipede ... $4.99

BRODERBUND

- Print Shop (All) ... $29.99
- Graphics Library I, II, III (All) ... $17.99
- Bank Street Writer (64, AT) ... $34.99
- Bank Street Writer (IIe/IIc) ... $44.99

BATTERIES INCLUDED

- Paperclip (AT, 64/128) ... $37.99
- Paperclip (IIe/IIc) ... $44.99
- Homepack (64/AT) ... $29.99
- D.E.G.A.S. (520) ... $27.99

CONTINENTAL SOFTWARE

- Home Accountant (All) ... $44.99
- Tax Advantage (All) ... $44.99
- Home Acct./Tax Bundle (All) ... $59.99

JANE

- Wordpro/dBase (C64) ... $39.99

MICROPROSE

- Kennedy Approach (All) ... $24.99
- Acrojet (All) ... $24.99
- Silent Service (All) ... $24.99

PFS

- File (64) ... $39.99
- File, Report, Graph (IIe/IIc) ... $79.99
- Write w/Proof (IIe/IIc) ... $79.99

PSI

- Fleet Systems III (C128) ... $59.99

SPRINGBOARD

- Newsroom (64) ... $34.99
- Clip Arts (64) ... $22.99
- Newsroom (Apple) ... $39.99
- Clip Arts (Apple) ... $22.99

SUBLOGIC

- Flight Simulator (All) ... $37.99
- Jet (64) ... $37.99

DISKETTES

maxell

- 3½" SS/DD (10) ... $18.99
- 3½" DS/DD (10) ... $29.99
- 5¼" MD-1 w/Hardcases (10) ... $11.99
- 5¼" MD-2 w/Hardcases (10) ... $16.99
- 5¼" MD-2-HD for AT (10) ... $32.99
- 3½" 5 pack SS/DD/Case ... $10.99

Verbatim.

- 5¼" SS/DD ... $12.99
- 5¼" DS/DD ... $24.99
- Disk Analyzer ... $24.99

Dennison

- Elephant 5¼" SS/SD ... $13.99
- Elephant 5¼" SS/DD ... $14.99
- Elephant 5¼" DS/DD ... $16.99
- Elephant Premium DS/DD(50) ... $79.99
- Elephant 3½" SS/DD ... $29.99

IBM

- 5¼" DS/DD floppy disks (Box of 10) ... $26.99

DISK HOLDERS

INNOVATIVE CONCEPTS

- Flip'n File 10 ... $2.99
- Flip'n File 50 ... $14.99
- Flip'n File 50 w/lock ... $19.99
- Flip'n File 100 ... $19.99
- Flip'n File Data Case ... $9.99

AMARAY

- 50 Disk Tub 5¼" ... $9.99
- 30 Disk Tub 3½" ... $9.99

MODEMS

ANCHOR

- Volksmodem ... $59.99
- Volksmodem 300/1200 ... $169.99
- Signalman Express ... $259.00
- Lightning 2400 Baud ... $399.00
- Expressi (PC Halfcard) ... $169.00
- 6470 (64/128) 300/1200 Baud ... $139.00

DIGITAL DEVICES

- AT300 - 300 Baud (Atari) ... $99.99

Hayes

- Smartmodem 300 ... $139.00
- Smartmodem 1200 ... $389.00
- Smartmodem 1200B ... $359.00
- Smartmodem 2400 ... $599.00
- Micromodem IIe ... $149.00
- Smart Com II ... $89.99
- Chronograph ... $199.00
- Transet 1000 ... $309.00

AST

- Reach 1200 Baud Half Card ... $399.00

SUPRA

- MPP-1064 AD/AA (C-64) ... $69.99

Novation

- Smart Cat Plus ... $299.00
- J-Cat ... $99.99
- Novation 2400 ... $549.00
- Apple Cat II ... $229.00
- 212 Apple Cat II ... $379.00
- Apple Cat 212 Upgrade ... $229.00
- Macmodem ... $279.00

QUADRAM

Quadmodem II

- 300/1200 ... $339.00
- 300/1200/2400 ... $499.00

EVEREX

- 1200 Baud Internal (IBM/PC) ... $179.00

DRIVES

HARD

I-MEGA

- A110H Single 10 ... CALL
- A210H 10 + 10 ... CALL
- A120H Surge 20 ... CALL
- A220H 20 + 20 ... CALL
- Save on 10 & 20 Carts ... CALL

TALLGRASS TECHNOLOGIES

COMMITTED TO MEMORY

- 25, 35, 50, 80 meg (PC) ... from $1299.00

IRWIN

- Tape Backup ... CALL

EVEREX

- 60 Meg Internal Backup System $799.00

CORE

- AT20-AT72MB ... CALL

PRIAM

- 40, 60 MB (PC, XT, AT) ... CALL

KITS

- 10 Meg with controller ... $379.00
- 20 Meg with controller ... $499.00

FLOPPY

ALLIED TECHNOLOGY

- Apple II,II+,IIe ½ height ... $109.00

INDUS

- Atari GT ... $199.00
- C-64 /128 GT ... $199.00

MSD

- SD1 C-64 Single ... $219.00
- SD2 C-64 Dual ... $469.00

Tandon

- 320K 5¼" (PC) ... $119.00

TEAC

- 320K 5¼" ... $119.00

DEPT. A205 CALL TOLL-FREE

www.commodore.ca

MONITORS

AMDEK

Video 300 Green....................$119.00
Video 300A Amber....................$129.00
Video 310A Amber TTL....................$159.00
Color 600 Hi-Res. RGB....................$399.00
Color 710 Ultra Hi-Res....................$439.00
Color 722 Dual Mode....................$529.00

MAGNAVOX

8562 RGB/Composite....................$279.00
613 TTL Green....................$99.99
623 TTL Amber....................$99.99

NEC

JB1205A....................$79.99
JB1270G/1275A..........(ea.) $99.99
JB1280G TTL Green....................$129.00
JB1285A TTL Amber....................$129.00
JC1460 RGB....................$229.00
JC1225 Composite....................$179.00
JC1401 Multi Sync RGB....................$549.00

PRINCETON

MAX-12 Amber....................$179.00
HX-9 9" RGB....................$469.00
HX-9E Enhanced....................$519.00
HX-12 12" RGB....................$469.00
HX-12E Enhanced....................$559.00
SR-12 Hi-Res....................$599.00
SR-12P Professional....................$699.00

TAXAN

115 12" Green....................$119.00
116 12" Amber....................$129.00
121 TTL Green....................$139.00
122 TTL Amber....................$149.00
620 640x200 RGB....................$NEW
630 640x200 RGB....................$NEW
640 720x400 RGB....................$NEW

QUADRAM

8400 Quadchrome I....................$499.00
8410 Quadchrome II....................$339.00
8420 Amberchrome....................$179.00
8500 Quad Screen....................$1449.00

ZENITH

ZVM 1220 Amber....................$99.99
ZVM 1230 Green....................$99.99
ZVM 1240 IBM Amber....................$149.00
ZVM 135 RGB....................$459.00
ZVM 1330 RGB....................$459.00
ZVM 1360 RGB....................CALL
ZVM 1380 E G Comp....................CALL

INTERFACES

AST

Multi I/O (Apple II)....................$149.00

PRACTICAL PERIPHERALS

Graphcard....................$79.99
Seriall Card....................$99.99
Microbuffer II+....................$169.00
Microbuffer 32K....................$189.00

QUADRAM

Microfazer....................from $139.00
Efazer (Epson)....................from $79.99

Orange Micro

Grappler CD (C64)....................$89.99
Grappler Plus (IIe, IIc)....................$89.99
Grappler C (IIc)....................$89.99
Grappler 16K (IIe, II+)....................$149.00

DIGITAL DEVICES

Ape Face (Atari)....................$49.99
U-Print A (Atari)....................$54.99
U-A16/Buffer (Atari)....................$74.99
U-Call Interface (Atari)....................$39.99
U-Print C (C64)....................$49.99
P-16 Print Buffer....................$74.99
U-Print 16 apple IIc....................$89.99

PRINTERS

Canon

A40,A50,A55....................CALL
LBP-8A1 Laser....................CALL

CITIZEN

MSP-10 (80 col.)....................$279.00
MSP-15 (132 col.)....................$389.00
MSP-20 (80 col.)....................$349.00
MSP-25 (132 col.)....................$509.00

C.ITOH

Prowriter 7500....................$169.00
Prowriter 1550P....................$349.00
Starwriter 10-30....................$399.00
3500 Tri Printer....................$1499.00

corona

Lazer LP-300....................$2799.00

DIABLO

620 Daisywheel....................$299.00
D25 Daisywheel....................$549.00
635 Daisywheel....................$1099.00
D80IF Daisywheel....................CALL

daisywriter

2000....................$699.00

EPSON

Homewriter 10, LX-80....................CALL
FX-85, FX-286, RX-100....................CALL
DX-10, DX-20, DX-35....................CALL
SQ-2000, Hi-80, HS-80, AP-80....................CALL
LQ-800, LQ-1000....................CALL

JUKI

6000 Letter Quality....................CALL
6100 Letter Quality....................CALL
6200 Letter Quality....................CALL
6300 Letter Quality....................CALL
5510 Dot Matrix....................CALL

LEGEND

808 Dot Matrix 100 cps....................$179.00
1080 Dot Matrix 100 cps....................$259.00
1380 Dot Matrix 130 cps....................$289.00
1385 Dot Matrix 165 cps....................$339.00

NEC

3000 Series....................$779.00
8000 Series....................$1099.00
ELF 360....................$399.00
Pinwriter 560....................$999.00

OKIDATA

182, 183, 192, 193, 2410, 84....................CALL
Okimate 10 (Specify C64/Atari)$189.00
Okimate 20 (IBM)....................CALL

Panasonic

KX1080....................NEW
KX1091....................$259.00
KX1092....................$389.00
KX1592....................$469.00
KX1595....................$659.00

QUADRAM

Quadjet....................$399.00
Quad Laser....................CALL

SILVER-REED

500 Letter Quality....................$219.00
550 Letter Quality....................$419.00
800 Letter Quality....................$699.00

star

SG-10A (Atari)....................CALL
SG-10C (C64 Interface)....................CALL
SB/SD/SG/SR Series....................CALL
Powertype Letter Quality....................CALL

Texas Instruments

TI850....................$529.00
TI855....................$639.00
TI865....................$799.00

TOSHIBA

1340 (80 column)....................$389.00
P341 (132 column)....................$799.00
P351 (132 column)....................$1049.00

SOFTWARE FOR IBM

ANSA SOFTWARE

Paradox....................$499.00

ASHTON-TATE

Framework II....................$389.00
dBase III Plus....................$389.00

BORLAND

Lightening....................$59.99
Sidekick (unprotected)....................$57.99
Reflex....................$59.99
Newspack....................$57.99

CENTRAL POINT

Copy II PC-Backup....................$29.99

DECISION RESOURCES

Chartmaster....................$229.00
Signmaster....................$169.00
Diagram Master....................$209.00

FIFTH GENERATION

Fast Back....................99.99

FOX AND GELLER

Quickcode III....................$169.00

FUNK SOFTWARE

Sideways....................$44.99

HARVARD SOFTWARE INC.

Total Project Manager....................$269.00

INFOCOM

Cornerstone....................$269.00

LIFETREE

Volkswriter III....................$159.00

LIVING VIDEOTEXT

Think Tank....................$109.00
Ready....................$64.99

LOTUS

Symphony....................CALL
1-2-3 Version 2....................CALL

MECA SOFTWARE

Managing Your Money 2.0....................$99.99

MICROPRO

Easy....................$94.99
WordStar 2000....................$239.00
WordStar 2000+....................$289.00
WordStar Professional....................$189.00

MICRORIM SOFTWARE

R:Base 4000....................$249.00
R:Base 5000....................$389.00
Clout 2.0....................$129.00

MICROSOFT

Flight Simulator....................$34.99
MultiPlan....................$129.00
Word....................$249.00
Mouse....................$139.00

MICROSTUF

Crosstalk XVI....................$89.99
Crosstalk Mark IV....................$149.00
Remote....................$89.99

MULTIMATE

Multi Mate Word Proc....................$219.00
Advantage....................$289.00
On File....................$89.99
Just Write....................$89.99

NOUNEMON

Intuit....................$69.99

NORTON

Norton Utilities 3.1....................$59.99

ONE STEP

Golf's Best....................$37.99

PFS: IBM

Proof....................$59.99
File/Graph..........(ea.) $79.99
Report....................$74.99
Write/Proof Combo....................$79.99

PROFESSIONAL SOFTWARE

Write-N-Spell....................$89.99

THE SOFTWARE GROUP

Enable....................$329.00

SATELLITE SYSTEMS

Word Perfect 4.1....................$219.00

SORCIM/IUS

Accounting
AP/AR/GL/INV/OE..........(ea.) $299.00
SuperCalc III....................$199.00
EasyWriter II System....................$239.00
Super Project....................$199.00

SPI SOFTWARE

Open Access....................$299.00

SUBLOGIC

Jet....................$37.99

IBM

IBM PC SYSTEMS
Configured to your specifications.
Call for Best Price!
IBM-PC, IBM-XT, IBM-AT

ZENITH

PC-138 Series, PC-148 Series, PC-158 Series, PC-160 Series, PC-171 Series, AT-200 Series....................CALL

SANYO

MBC 550-2, MBC 555-2, MBC 675 Portable, MBC775, MBC 880 DesktopCALL

AT&T

Safari (7300)....................CALL
6300....................CALL

corona

PPC400 Dual Portable....................$1289.00
PPCXT 10 meg Portable....................$1989.00
PC40022 Dual Desktop....................$1389.00
PC400-HD2 10 meg....................$1989.00

ITT

ITT X-TRA
256K, 2 Drive System....................CALL
256K,10 meg Hard Drive System CALL
XP5, 20 meg....................CALL

SPERRY

Sperry-AT..........as low as $1749.00
Sperry-IT..........as low as $2699.00
Call for Specific Configuration!
All Models....................CALL

KAYPRO

KP-2000 Portable....................CALL
Kaypro PC....................CALL

MULTIFUNCTION CARDS

AST

RamVantage....................$349.00
Rampage-PC....................$379.00
Rampage-AT....................CALL
Six Pack Plus....................$229.00
I/O Plus II....................$139.00
Advantage-AT....................$399.00
Preview Mono....................$299.00
PC Net Cards....................$379.00
5251/11 On-line....................$669.00
5251/12 Remote....................$579.00

dca

IRMA 3270....................$879.00
IRMA Print....................$999.00
IRMA Smart Alec....................$779.00

EVEREX

Edge Card....................$259.00
Graphics Edge....................$229.00
Magic Card II....................$159.00
Magic Card I....................$99.99

HERCULES

Graphics....................$299.00
Color....................$159.00

IDEAssociates

IDEA 5251....................$589.00

MYLEX

The Chairman....................$439.00

PARADISE

Color/Mono Card....................$149.00
Modular Graphics Card....................$259.00
Multi Display Card....................$199.00
Five Pack C, S....................$129.00

PERSYST

Bob Board....................$359.00

TECMAR

Captain - 64....................$199.00
Graphics Master....................$469.00

QUADRAM

Quadport-AT....................$119.00
Liberty-AT (128K)....................$349.00
The Gold Quadboard....................$449.00
The Silver Quadboard....................$239.00
Expanded Quadboard....................$199.00
Liberty....................$309.00
QuadSprint....................$499.00
QuadLink....................$399.00
QuadColor....................$199.00
Quadboard-AT....................CALL
8600 E.G.A. card....................$399.00

INTEL

PCNC8087 5MHz.................... *CALL*
PCNC8087-2 8 MHz.................... *FOR*
PCNC80287 6 MHz.................... *YOUR*
1010 PC-Above Board.................... *PC*
1110 PS-Above Board....................
2010 AT-Above Board....................

COMPUTER MAIL ORDER

1-800-233-8950

www.commodore.ca

press the function key labeled Outline. Within seconds, an outline of the entire article is before you. Move the cursor to the topic that interests you, press Enter, and you'll go directly to that section.

Though it contains more than 30,000 articles, *The Electronic Encyclopedia* is, by its very nature, general. It is not detailed enough to be particularly helpful in serious research, though it does quite well at answering general questions. Thanks to the compact disc format, however, it is practical and economical to update regularly, another advantage over its paper counterparts. Grolier promises to update the encyclopedia each year for $24.95.

Having all the information in 20 volumes available on a single disc is exciting, but even more thrilling is what it promises for the future. Imagine a whole library of compact disc databases—indexes to law libraries, census data, technical journals—and imagine what it might mean to have the key to all that information—the key to the information age.

Philips CM100 CD-ROM
The Electronic Encyclopedia
Grolier Electronic Publishing, Inc.
Sherman Turnpike
Danbury, CT 06816
$1,495
The Electronic Encyclopedia
(without drive) $199

The Body In Focus

Larry Krengel

Requirements: Commodore 64; Apple II-series computer with at least 64K RAM; IBM PC with color/graphics adapter; or an IBM PCjr. Disk only.

I'm always excited when I see computer software that displays impressive graphics. And *The Body in Focus* is exciting software. What's more, this human anatomy program is educational, accurate, and engaging.

I've been teaching biology for 15 years. When I first examined *The Body in Focus,* I found it to be technically correct. But why did I continue looking at the program long after I had assured myself it was biologically valid? Because I didn't want to miss any of the great graphic presentations.

For example, the designers must have burned the midnight oil to make the simulated body sneeze and even raise goose bumps. My kids thought they were seeing things when the skeleton swung its head around to demonstrate a pivot joint. (When you can't believe what you're seeing, one keypress repeats the action.)

With The Body in Focus, *anyone can have X-ray vision. (Commodore 64 version.)*

The Body in Focus comes on two disks. The first contains tutorials, and the second presents questions based on the tutorials. Eight vital body systems are covered—including the circulatory, respiratory, endocrine, skeletal, nervous, digestive, muscular, and integumentary (skin, that is). A tour of each system takes 10 to 15 minutes.

To illustrate each body system, the disk contains three "body closeups": a closeup view of each system within the head, torso, and arm. The closeups let you strip away the body layer by layer, going deeper and deeper. It's like a graphic dissection. If you find one of the screens particularly interesting, there's a "tell me more" key which calls up a more complete discussion of what you see.

Body Trivia

The second disk contains a library of more than 200 questions based on the tutorials. If you think you already know a lot about the human body, try the "body I.Q. test" before using the tutorials. Do you know how many taste buds there are on a human tongue? Or how long it takes your body to pump 3,000 gallons of blood? Or why you get goose bumps?

The Body in Focus is very simple to use. Which key do you press to view the digestive system? Or the skeletal system? You don't have to memorize these details—a soft plastic keyboard overlay lists all the key functions.

Another strong point of the program is its use of sound. With some software, you can often tell that sound was added as an afterthought. But with *The Body in Focus,* the sound is very functional. You quickly learn the different sounds for "your turn" and "that key doesn't do anything." Sounds are used intelligently for getting your attention at the right time and for signifying right and wrong responses.

As a parent, I would recommend *The Body in Focus* for a youngster who is taking biology. As a teacher, I probably wouldn't use *The Body in Focus* as part of the curriculum because it is not sequential (some students may choose not to press "tell me more"). However, this program is exciting enough that I think students would invest their own time to travel through *The Body in Focus.* It would be great for enrichment.

The Body in Focus
CBS Interactive Learning
One Fawcett Place
Greenwich, CT 06836
$39.95

One-On-One For Amiga

Charles Brannon, Program Editor

Requirements: Amiga with 256K RAM. Joystick and 512K RAM recommended.

As promised, Electronic Arts has successfully translated for the Amiga several of its popular games originally written for the Commodore 64, Atari, and Apple. These programs are showcase pieces of game design. On computers like the Amiga, they can be even better, using the power of the machine to enhance the realism with additional color, detail, and smoother motion. The Amiga's stereo sound system can also be exploited for more realistic music and unusual sound effects.

Not all games can easily incorporate these new features without being redesigned, however. Because *One-on-One* is an adaptation, it is very similar to the original version running on, say, a Commodore 64. Although the Amiga version has a more colorful screen with true-to-life color schemes, basically the original game's graphics have been retouched. For example, the basketball court is convincingly colored to look like a polished wood floor. This is possible due to the Amiga's ability to display up to 32 colors simultaneously on its low-resolution graphics screen. (The term *low resolution* is relative; the same resolution of 320 × 200 pixels is called *high resolution* on the Commodore 64.) Each color is chosen from a wide range of possible hues (4,096 in all), so it's easier to approximate real-life colors.

On the other hand, the sound effects in *One-on-One* are considerably enhanced, since the Amiga can play back digitally recorded sounds. You can hear the actual background noises of a basketball game, with the crowd murmuring, cheering, booing, catcalling,

www.commodore.ca

Famous Smith Corona National Brand

10" PRINTER SALE

Below Wholesale Cost Prices!!!

• ONE YEAR IMMEDIATE REPLACEMENT WARRANTY

- **Speed: 120 or 160 characters per second**
- **Friction Feed/Tractor Feed — Standard**
- **80 character print line at 10 CPI**
- **1 Line Buffer, 2K Buffer on 160 CPS Plus LQM**
- **Six pitches**
- **Graphics capability**
- **Centronics compatible parallel interface**
- **Features Bidirectional Print, Shortline Seek, Vertical And Horizontal Tabs**

SUPER GRAPHICS

This is a sample of our near-letter-quality print.

emphasized

italic print.

There is standard data processing quality print

Check These Features & Prices

120 CPS 10" Printer

List $429.00

SALE $159

160 CPS + Letter Quality Mode 10" Printer

List $499.00

SALE $199

SPECIFICATIONS

(IBM — Commodore)

Size/Weight
Height 5.04" Width 16.7"
Depth 13.4" Weight 18.7 lbs.

Internal Char. Coding
ASCII Plus ISO

Print Buffer Size
120 CPS: 132 Bytes (1 line)
120/160 CPS Plus LQM: 2K

No. of Char. in Char. Set
96 ASCII Plus International

Graphics Capability
Standard 60, 72, 120 DPI
Horizontal 72 DPI Vertical

Pitch
10, 12, 16.7, 5, 6, 8.3, Proportional Spacing

Printing Method
Impact Dot Matrix

Char. Matrix Size
9H x 9V (Standard) to 10H x 9V
(Emphasized & Elongate)

Printing Features
Bi-directional, Short line seeking, Vertical Tabs, Horizontal Tabs

Forms Type
Fanfold, Cut Sheet, Roll (optional)

Max Paper Width
11"

Feeding Method
Friction Feed Std.; Tractor Feed Std.

Ribbon
Cassette — Fabric inked ribbon

Ribbon Life
4 million characters

(Apple — Atari — Etc.)

Interfaces
Parallel 8 bit Centronics compatible
120/160 CPS Plus NLQ: RS232 Serial inc.

Character Mode
10 x 8 Emphasized; 9 x 8 Standard; 10 x 8 Elongated; 9 x 8 Super/Sub Script (1 pass)

Character Set
96 ASCII
11 x 7 International Char.

Line Spacing
6/8/12/72/144 LPI

Character Spacing
10 cpi normal; 5 cpi elongated normal; 12 cpi compressed; 6 cpi elongated compressed; 16.7 cpi condensed; 8.3 cpi elongated condensed; 5.12.5 cpi elongated proportional

Cartridge Ribbon — List $19.95. **Sale $12.95.**

Interfaces

IBM $89.00 **Apple $59.00** **Atari $59.00** **Commodore $39.95**

Add $14.50 for shipping, handling and insurance. Illinois residents please add 6¼ % tax. Add $29.00 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 delivery. 2 to 7 days for phone orders. 1 day express mail! Prices & Availability subject to change without notice.
VISA — MASTERCARD — C.O.D. No C.O.D. to Canada or APO-FPO

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

www.commodore.ca

COMMODORE 64 COMPUTER

(Order Now)

$139.95

- C128 Disks 79¢ ea.*
- Paperback Writer 64 $39.95
- 10" Comstar 10X Printer $148.00
- 14" Color Monitor $149.95

CALL BEFORE YOU ORDER

COMMODORE 64 SYSTEM SALE

Commodore 64
Com. 1541 Disk Drive
14" Color Monitor

$457

Plus $30.00 S&H

PLUS FREE $49.95 Oil Barons Adventure Program

C128 COMMODORE COMPUTER

NEW

(Order Now)

***$229.05** (SEE BELOW)

With $59.95 Timeworks Wordwriter Wordprocessor savings applied

- 340K 1571 Disk Drive $259.00
- Voice Synthesizer $39.95
- 12" Monitor $79.95

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.).

14" COLOR MONITOR $149.95

You pay only $149.95 when you order this 14" 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).

Premium Quality 120-140 CPS Comstar 10X Printer $148.00

The COMSTAR 10X gives you a 10" carriage, 120-140 CPS, 9 x 9 dot matrix with double strike capability for 18 x 18 dot matrix (near letter quality), high resolution bit image (120 x 144 dot matrix), underlining, back spacing, left and right margin setting, true lower decenders with super and subscripts, prints standard, italic, block graphics and special characters. It gives you print quality and features found on printers costing twice as much!! (Centronics Parallel Interface) List $399.00 **Sale $148.00.**

4 SLOT EXPANDER & 80 COLUMN BOARD $49.95

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! ***Limited Quantities*** **Sale $49.95.** *Coupon $39.95.*

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
Voice Command Module	$79.95	$39.95	$34.95
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
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
File Writer (by Codewriter)	$39.95	$29.95	$24.95
C64 Troubleshoot & Repair Guide	$24.95	$15.95	$12.95
Financial Planner — Sylvia Porter	$59.95	$38.95	$35.95

(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 $289.00

You pay only $289.00 for the C128 computer and we include the C128 Wordwriter Wordprocessor by Timeworks (Sale $59.95). Thus, your net cost for the C128 computer is only $229.05.
List $349.00. **SALE $289.00.**

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 1541, plus runs all 1541 formats.
List $349.00. **Sale $259.00.**

SUPER AUTO DIAL MODEM $29.95

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 $99.00. **SALE $29.95.** *Coupon $24.95.*

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!! PLUS ($19.95 value) TEXT TO SPEECH program included FREE, 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" MAGNAVOX (NAP) 80 COLUMN MONITOR WITH SOUND $79.95

Super High Resolution green screen monitor. 80 columns x 24 lines, easy to read, plus speaker for audio sound included. Fantastic value List $129.00 **Sale $79.95.** (C128 cable $19.95. C64, Atari cable $9.95)

PRINTER/TYPEWRITER COMBINATION $229.95

"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 $229.95. (Ltd. Qty.)**

14" RGB & COMPOSITE COLOR MONITOR $259.95

Must be used to get 80 columns in color with 80 column computers (C128 - IBM - Apple).(RGB Cable $19.95) 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. C.S.T. Weekdays
9 a.m. - 12 noon C.S.T. Saturdays

- 90 DAY FREE REPLACEMENT WARRANTY
- OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6¼% 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 Cashier Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Prices & Availability subject to change without notice.
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

www.commodore.ca

Commodore Software Sale

ORDER TODAY!

Name ______________________________
Address ______________________________
City ____________ State ______ Zip ______
Visa/Mastercard No. ______________________________
Exp. Date ________ Phone (______) ______________

(T) Tape, (C) Cartridge, (D) Disk.

GAMES

Accolade

- ☐ 5950 HARDBALL (D) $29.95 $18.95
- ☐ 5952 LAW OF THE WEST (D) 29.95 18.95
- ☐ 5954 FIGHT NIGHT (D) 29.95 18.95
- ☐ 5956 PSI 5 TRADING CO. (D) 29.95 18.95
- ☐ 5958 THE DAM BUSTERS (D) 29.95 18.95

Activision

- ☐ 0757 RIVER RAID (D) $39.95 $18.95
- ☐ 0761 PITFALL II — LOST CAVERNS (D) . 39.95 18.95
- ☐ 0900 SPACE SHUTTLE (D) 37.95 18.95
- ☐ 0932 ON FIELD FOOTBALL (D) 39.95 18.95
- ☐ 0936 ON COURT TENNIS (D) 39.95 18.95
- ☐ 0940 GHOSTBUSTERS (D) 39.95 22.95
- ☐ 3580 GREAT AMERICAN RD. RACE (D) . 29.95 20.95
- ☐ 3582 MASTER OF THE LAMPS (D) 29.95 20.95
- ☐ 3584 COUNTDOWN/SHUTDOWN (D) .. 29.95 20.95
- ☐ 3588 MINDSHADOW (D) 29.95 20.95
- ☐ 3590 STAR LEAGUE BASEBALL (D) 29.95 20.95
- ☐ 3592 ALCAZAR (D) 29.95 20.95
- ☐ 5196 LITTLE PEOPLE PROJECT (D) 34.95 24.95
- ☐ 5198 FAST TRACKS (D) 34.95 20.95

Broderbund

- ☐ 2900 MASK OF THE SUN (D) $39.95 $23.95
- ☐ 2901 OPERATION WHIRLWIND (D) 39.95 22.95
- ☐ 2903 LODE RUNNER (D) 34.95 19.95
- ☐ 2904 THE CASTLES OF DR. CREEP (D) .. 29.95 18.95
- ☐ 2906 WHISTLERS BROTHER (D) 29.95 18.95
- ☐ 3039 STEALTH (D) 29.95 22.95
- ☐ 3041 RAID ON BUNGELING BAY (D) .. 29.95 18.50
- ☐ 2905 KARATEKA (D) 29.95 23.95
- ☐ 3038 CHAMPION LODE RUNNER (D) .. 34.95 26.95
- ☐ 5158 BANK STREET WRITER (D) 49.95 32.95
- ☐ 5330 BANK STREET SPELLER (D) 49.95 32.95
- ☐ 5332 BANK STREET FILER (D) 49.95 32.95
- ☐ 5334 BANK STREET MAILER (D) 49.95 32.95

Datasoft

- ☐ 3025 BRUCE LEE (D) $34.95 $18.95
- ☐ 3026 PAC-MAN (D) 34.95 18.00
- ☐ 3027 MIGHTY CONAN (D) 34.95 18.00
- ☐ 3028 MR DO! (D) 34.95 18.95
- ☐ 3029 DIG DUG (D) 34.95 18.95
- ☐ 3032 POLE POSITION (D) 34.95 18.95
- ☐ 5216 ALTERNATE REALITY 39.95 25.95
- ☐ 5218 THE GOONIES (D) 29.95 18.95
- ☐ 5220 ZORRO (D) 29.95 18.95

Electronic Arts

- ☐ 3830 DR. J & LARRY BIRD (D) $29.95 $23.95
- ☐ 3832 FINANCIAL COOKBOOK (D) 39.95 27.95
- ☐ 3834 MAIL ORDER MONSTERS (D) 34.95 22.95
- ☐ 3840 THE SEVEN CITIES OF GOLD (D) .. 29.95 23.95
- ☐ 3842 SKY FOX (D) 29.95 23.95
- ☐ 5176 CARRIERS AT WAR (D) 42.95 32.95
- ☐ 5178 REACH FOR THE STARS II (D) 37.95 28.95
- ☐ 5180 HEART OF AFRICA (D) 29.95 23.95
- ☐ 5182 MOVIE MAKER (D) 29.95 23.95
- ☐ 5184 EUROPE ABLAZE (D) 42.95 34.95
- ☐ 5186 M.U.L.E. (D) 19.95 16.95
- ☐ 5188 MURDER ON ZINDERNEUF (D) ... 19.95 16.95
- ☐ 5190 MUSIC CONSTRUCTION SET (D) . 19.95 16.95
- ☐ 5192 PINBALL CONSTRUCTION SET (D) 19.95 16.95
- ☐ 5194 RACING CONSTRUCTION SET (D) 29.95 22.95

Epyx

- ☐ 0337 WORLD'S GREAT FOOTBALL (D) $39.95 $23.95
- ☐ 0338 WINTER GAMES (D) 39.95 20.95
- ☐ 0339 THE EIDOLON (D) 39.95 20.95
- ☐ 0340 KORONIS RIFT (D) 39.95 20.95
- ☐ 0360 JET COMBAT SIMULATION (D) .. 39.95 20.95
- ☐ 0364 SUMMER OLYMPIC GAMES (D) .. 39.95 18.95
- ☐ 0365 WORLD'S GREAT BASEBALL (D) .. 34.95 20.95
- ☐ 0382 SUMMER OLYMPIC GAMES II (D) . 39.95 20.95
- ☐ 0750 PITSTOP II (D) 39.95 22.95
- ☐ 2046 IMPOSSIBLE MISSION (D) 34.95 16.95
- ☐ 2066 ROBOTS OF DAWN (D) 39.95 15.95
- ☐ 2070 BARBIE (D) 39.95 18.95
- ☐ 2074 G.I. JOE (D) 39.95 18.95
- ☐ 2085 BREAKDANCE (D) 39.95 16.95
- ☐ 2305 SCRABBLE (D) 39.95 24.95
- ☐ 3004 CHIPWITS (D) 34.95 15.95
- ☐ 3005 BALLBLAZER (D) 29.95 24.95
- ☐ 3006 RESCUE ON FRACTALUS! (D) 29.95 20.95

Strategic Simulations, Inc.

- ☐ 2995 RDF 1985 (D) $34.95 $20.95
- ☐ 2996 COMBAT LEADER (D) 39.95 23.95
- ☐ 2997 GEOPOLITIQUE (D) 39.95 24.95
- ☐ 2998 BALTIC 1985 (D) 34.95 20.95
- ☐ 3008 RINGSIDE SEAT (D) 39.95 23.95
- ☐ 3009 THE COSMIC BALANCE (D) 39.95 23.95
- ☐ 3010 IMPERIUM GALACTUM (D) 39.95 23.95
- ☐ 3011 CARTELS AND CUTTHROATS (D) . 39.95 23.95
- ☐ 3012 RAILS WEST (D) 39.95 26.95
- ☐ 3013 TIGERS IN THE SNOW (D) 39.95 23.95
- ☐ 3014 PROFESSIONAL TOUR GOLF (D) .. 39.95 23.95
- ☐ 3015 50 MISSION CRUSH (D) 39.95 23.95
- ☐ 3016 PRESIDENT ELECT (D) 39.95 23.95
- ☐ 3017 BROADSIDES (D) 39.95 24.95
- ☐ 3018 COMPUTER QUARTERBACK (D) .. 39.95 24.95
- ☐ 3020 COMPUTER AMBUSH (D) 59.95 37.95
- ☐ 3021 COMPUTER BASEBALL (D) 39.95 23.95
- ☐ 3030 KNIGHTS IN THE DESERT (D) 39.95 23.95
- ☐ 3031 FIELD OF FIRE (D) 39.95 23.95

Suncom

- ☐ 3876 PARTY QUIZ (D) $49.95 $14.95
- ☐ 3880 GENERAL EDITION (D) 39.95 14.95
- ☐ 3882 GENERAL EDITION III 39.95 14.95
- ☐ 3884 SPORTS EDITION (D) 39.95 14.95
- ☐ 3886 EDUCATION EDITION (D) 39.95 14.95
- ☐ 3888 BIBLE EDITION (D) 39.95 14.95

BUSINESS

Codewriter

- ☐ 0129 FILEWRITER (D) $39.95 $29.95
- ☐ 0706 REPORTWRITER (D) 39.95 29.95
- ☐ 0707 ADVENTUREWRITER (D) 49.95 39.95
- ☐ 0708 ELF (D) 39.95 29.95
- ☐ 0709 DIALOG (D) 49.95 39.95
- ☐ 0784 MENUWRITER (D) 39.95 29.95
- ☐ 2550 SPEEDWRITER (D) 49.95 39.95
- ☐ 2551 HOME INTEGRATOR (D) 29.95 19.95

Softsync

- ☐ 5930 ACCOUNTANT, INC. (D) $99.95 $64.95
- ☐ 5932 PERSONAL ACCOUNTANT (D) 34.95 26.95
- ☐ 5934 MODEL DIET (D) 29.95 23.95
- ☐ 5936 TRIO (D) 49.95 45.95
- ☐ 5938 KID PRO QUO (D) 29.95 23.95
- ☐ 5940 DESK MANAGER (D) 39.95 28.95

Timeworks

- ☐ 0176 INVENTORY MANAGE (D) $69.95 $38.95
- ☐ 0180 ACCOUNTS RECEIVABLE/ INVOICING (D) 69.00 38.95
- ☐ 0182 ACCOUNTS PAYABLE/ CHECKWRITING (D) 69.00 38.95
- ☐ 0184 PAYROLL MANAGEMENT (D) 69.00 38.95
- ☐ 0188 GENERAL LEDGER (D) 69.00 38.95
- ☐ 0207 ELECTRONIC CHECKBOOK (D& T) 29.95 19.95
- ☐ 0231 MONEY MANAGER (D&T) 29.95 19.95
- ☐ 0235 DATA MANAGER (D&T) 29.95 19.95
- ☐ 0928 EVELYN WOOD SPEED READ (D) . 69.96 32.95
- ☐ 3743 SYLVIA PORTER (D) 59.95 38.95

C128 Software From Timeworks

- ☐ 5022 WORD WRITER/ SPELL CHECKER (D) $69.96 $59.95
- ☐ 5024 DATA MANAGER II (D) 69.96 49.95
- ☐ 5026 SWIFTCALC WITH SIDEWAYS (D) . 69.96 49.95

Phone Orders

CALL 8 to 8 C.S.T. - M-F

312-382-5244

EDUCATION

American Educational Computer

- ☐ 2482 ELEM. SCIENCE FACTS (D) $29.95 $14.95
- ☐ 2492 VOCABULARY WORD BUILD (D) .. 29.95 14.95
- ☐ 2493 GRAMMAR WORD SKILLS (D) 29.95 14.95
- ☐ 2494 WORLD GEOGRAPHY FACTS (D) . 29.95 14.95
- ☐ 2495 SPANISH VOCAB. SKILLS (D) 29.95 14.95
- ☐ 2496 FRENCH VOCAB. SKILLS (D) 29.95 14.95
- ☐ 2497 WORLD HISTORY (D) 29.95 14.95
- ☐ 2498 U.S. HISTORY FACTS (D) 29.95 14.95
- ☐ 2499 BIOLOGY FACTS (D) 29.95 14.95
- ☐ 2519 U.S. GEOGRAPHY FACTS (D) 29.95 14.95
- ☐ 2520 U.S. GOVERNMENT FACTS (D) ... 29.95 14.95
- ☐ 2521 AEC SPELLING (D) 39.95 24.95
- ☐ 3745 PHONICS (D) 39.95 24.95
- ☐ 3747 LEARN TO READ (D) 39.95 24.95
- ☐ 3749 READING COMPRENSION (D) ... 39.95 24.95

Designware

- ☐ 0824 GRAMMAR EXAMINER (D) $39.95 $24.95
- ☐ 0828 SPELLAKAZAM (D) 34.95 19.95
- ☐ 0832 STATES & TRAITS (D) 44.95 27.95
- ☐ 0836 SPELLICOPTER (D) 39.95 24.95
- ☐ 0840 CREATURE CREATOR (D) 34.95 19.95
- ☐ 0844 TRAP-A-ZOID (D) 39.95 25.95
- ☐ 2518 THE BODY TRANSPARENT (D) 44.95 27.95
- ☐ 2517 EUROPEAN NATIONS & LOCATIONS (D) 44.95 27.95
- ☐ 2062 MATH MAZE (D) 39.95 26.95
- ☐ 5100 ALGEBRA I (D) 39.95 28.95
- ☐ 5102 REMEMBER (D) 69.96 49.95
- ☐ 5104 WEBSTER'S NUMBERS (D) 39.95 28.95
- ☐ 5105 SPELLING & READ PRIMER (D) 39.95 24.95
- ☐ 5106 ALGEBRA 2 (D) 39.95 28.95
- ☐ 5107 ALGEBRA 3 (D) 39.95 28.95

Mindscape

- ☐ 5108 KEYBOARD CADET (D) 39.95 25.95
- ☐ 5110 BANK STREET MUSIC WRITER (D) . 39.95 25.95
- ☐ 5112 CROSSWORD MAGIC (D) 49.95 29.95
- ☐ 5114 THE PERFECT SCORE (D) 69.96 45.95
- ☐ 5116 COLORME/RAINBOW BRITE (D) .. 34.95 18.95
- ☐ 5118 THE HALLEY PROJECT (D) 39.95 25.95
- ☐ 5120 INDIANA JONES IN THE LOST KINGDOM (D) 29.95 18.95
- ☐ 5122 BANK STREET STORYBOOK (D) .. 39.95 25.95
- ☐ 5910 THE DOLPHIN'S RUNE (D) 29.95 18.95
- ☐ 5912 THE LUSCHER PROFILE (D) 39.95 25.95
- ☐ 5914 QUAKE MINUS ONE (D) 29.95 18.95
- ☐ 5916 THE LORDS OF MIDNIGHT (D) 29.95 18.95
- ☐ 5918 SHADOWFIRE (D) 29.95 18.95

Weekly Reader

- ☐ 2511 STICKYBEAR BOP (D) $34.95 $19.95
- ☐ 2512 STICKYBEAR NUMBERS (D) 34.95 19.95
- ☐ 2513 STICKYBEAR BASKETBOUNCE (D) 34.95 19.95
- ☐ 2514 STICKYBEAR OPPOSITES (D) 34.95 19.95
- ☐ 2515 STICKYBEAR ABC (D) 34.95 19.95
- ☐ 2516 STICKYBEAR SHAPES (D) 34.95 19.95
- ☐ 2600 PIC BUILDER (D) 29.95 19.95
- ☐ 5126 STICKYBEAR SPELLGRABBER (D) . 29.95 19.95
- ☐ 5128 STICKYBEAR TOWN BUILDER (D) . 29.95 19.95
- ☐ 5130 STICKYBEAR MATH (D) 29.95 19.95

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6¼% 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! Prices & availability subject to change without notice.
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

www.commodore.ca

The Amiga version of One-on-One *has more colorful graphics and real-life sound effects.*

and roaring when a player makes a basket (if the Amiga is hooked up to a stereo sound system, you'll notice that the cheering is loudest from the scoring team's side). And if you listen closely, you can even hear a vendor wandering through the crowd ("Hot Dogs! Cold Beer!").

Two Superstars

For those unfamiliar with *One-on-One*, it's a two-man basketball game with a 3-D perspective of the court. The two ballplayers are none other than Larry Bird and Dr. J (Julius Erving). Consulting with Larry Bird and Dr. J, Electronic Arts has modeled the actual playing characteristics of the two superstars. You control the action with the mouse or a joystick. Push forward to move toward the basket, back to move away, and left and right to move laterally. A quick press of the button spins you around (a "360"). If you're holding the ball, a long press sends it flying towards the basket (you have to time it carefully). If you're not holding the ball, a long press sends your player leaping up for a rebound or attempted block.

The computer sometimes adjusts your player's position. When you're facing the basket and the opposing player shoots, you automatically turn around to face the ball so you can jump up and attempt to block it.

The game attempts to be realistic without encumbering arcade-style play. A bar graph at the bottom of the screen shows each player's remaining energy, a sort of exhaustion factor. As your energy drains, from strenuous running, leaping, and blocking, your player becomes sluggish, moves slower, is unable to jump very high, and misses more shots. If you stand still, your energy level builds as you rest. When either player takes a time out or when a quarter ends, both players are refreshed. This is an important part of the game, since if you had infinite energy, you could run the best players off the court.

Every feature of professional basketball is here: the 24-second shot clock, the three-point goal, and penalties for hacking, charging, blocking, and traveling. However, the game makes no attempt to charge you for goaltending—where you try to block a shot on its downward flight into the basket.

Although you have a choice of using the mouse or a joystick in *One-on-One*, the mouse doesn't make a very good controller in this game. You have to keep moving it constantly to keep your player going. This is difficult with limited desk space for the mouse. A joystick affords much better control. (Any Commodore or Atari joystick works with the Amiga.)

Master Of The Slam-Dunk

While playing *One-on-One*, I found that the simulation of the characters really doesn't seem to affect the game much. As in real life, Larry Bird is nearly always able to make a three-point shot and Dr. J. is the master of the slam-dunk, but there doesn't seem to be all that much difference between the players. (However, a 76ers or Celtics fan might instantly notice some subtle nuances.) In the end, it's joystick dexterity coupled with some basketball experience that determines the ultimate winner.

Pull-down menus that work with either the joystick or the mouse let you select the game's difficulty level, loser's outs versus winner's outs, and whether you are competing for points or against time. There are four difficulty levels: Park & Rec, Varsity, College, and Pro. You can also play against the computer, choosing which player the computer controls. If you can beat the computer playing at the Pro level, you can whip most human opponents.

As proof of the careful attention that went into this game, Electronic Arts has included several cute features that give the game a special character. When the computer considers that it's just observed a particularly interesting or amazing shot, it calls for an *Instant Replay* that repeats the last few seconds—quite flattering if your player made the shot, but bound to draw a sneer from your opponent. A sufficiently powerful slam-dunk can shatter the backboard, raining down slivers of glass. A little janitor waddles out with a broom, looks at you and grumbles, then sweeps the fragments into his dustpan. This happened twice within a few hours of play, so it's more likely to happen than in real life.

Although the graphics and sound are uniquely Amiga, the game play is consistent with earlier versions. A testament to careful research and clever programming, perhaps this element of *One-on-One* doesn't really need improving, even on such a powerful computer as the Amiga.

One-on-One
Electronic Arts
1820 Gateway Drive
San Mateo, CA 94404
$39.95

AtariWriter Plus

Tom R. Halfhill, Editor

Requirements: Atari 400, 800, XL, or XE computer with at least 48K RAM, a disk drive, and printer. Special version included for 130XE.

The original *AtariWriter* was by no means the first word processor for Atari computers, nor the most powerful—but it quickly became one of the most popular. Its manageable range of commands, ease of use, and pop-in cartridge convenience rapidly made it the word processor of choice for thousands of beginners and experienced users alike.

Now comes its successor: *AtariWriter Plus*. All new and improved, *AtariWriter Plus* nevertheless retains a familiar resemblance to *AtariWriter*. Significant new features include an integrated spelling checker program with 36,000-word dictionary, an integrated mail-merge program, a utility program that lets you construct your own printer driver in case you have an oddball printer, the ability to take advantage of the 128K RAM in the 130XE, selectable insert/strikeover modes, optional horizontal scrolling up to 249 columns, double-column printing, form-letter printing, support for dual-drive systems, and a larger failsafe buffer.

Buy One, Get One Free

AtariWriter Plus comes on two disks instead of a cartridge, and one of the disks is a flippy with a different version of the program on each side. One version works on all eight-bit Atari computers with at least 48K RAM, and the other version works only on the 130XE. The second disk contains the dictionary for the spelling checker.

There are disadvantages to this new arrangement, especially if you've grown accustomed to the instant start-up convenience and durability of the *AtariWriter* cartridge. *AtariWriter Plus* takes nearly a minute to load, and the program disk is copy-protected. That means if the disk fails after the 90-day warranty expires, you're out of action until you can get another (at the full

www.commodore.ca

replacement cost of the program). Fortunately, the dictionary disk is not copy-protected, and that's the disk which may get the heaviest use, as we'll explain below.

When you boot up *AtariWriter Plus*, the first thing you see after a title screen is the main menu. It's almost identical to the main menu of *AtariWriter* with these additions: Verify Spelling, Global Format, Mail Merge, Index Drive 1, and Index Drive 2. Verify Spelling and Mail Merge load those corresponding programs, which we'll cover in a moment. The Index selections call up disk directories for one- or two-drive systems (*AtariWriter* allowed access to only one drive).

Global Format is a new option that changes the way you format a file for printing. The old *AtariWriter* always places a row of formatting commands across the top of your file; *AtariWriter Plus* eliminates this. Instead, Global Format brings up a screen that lets you adjust these settings for any file currently in memory. Settings include top and bottom margins, page length, paragraph indention and spacing, type font, justification on/off, left and right margins, second left and right margins (for double-column printing), page numbering, line spacing, and page wait on/off (for single-sheet or tractor-feed paper). As before, all of these settings default to common values, and you can imbed additional formatting commands within your file if you need to change a setting partway through a document.

Although Global Format rids the writing/editing screen of distracting formatting codes, it adds them to the disk file. When you save an *AtariWriter Plus* document from the main menu, the program no longer saves the text in standard ASCII format as *AtariWriter* did. Instead, the formatting codes are tacked onto the file as a header. This restores the settings when you reload the file later, but also causes problems if you try to load an *AtariWriter Plus* document into another word processor or upload it via modem. Fortunately, Atari provided a solution: If you press CTRL-S at the main menu, a Save ASCII option pops up. It strips off the format header and saves the text in straight ASCII.

To convert an *AtariWriter* file for *AtariWriter Plus*, all you have to do is delete the print formatting line at the top of the document and reenter the codes on the Global Format screen. ASCII files created by other word processors are easily converted, too.

Adjustable Screen

Aside from the missing format line at the top, the writing/editing screen is

GET THE KNOW-HOW TO REPAIR EVERY COMPUTER ON THIS PAGE. AND MORE.

IBM is a Registered Trademark of International Business Machine Corporation.

Epson is a Registered Trademark of Epson America, Inc.

Apple and the Apple logo are Registered Trademarks of Apple Computer, Inc.

Compaq is a Registered Trademark of COMPAQ Computer Corporation.

©1984 AT&T Technologies, Inc.

Learn the Basics the NRI Way—and Earn Good Money Troubleshooting Any Brand of Computer

The biggest growth in jobs between now and 1995, according to Department of Labor estimates, will occur in the computer service and repair business, where demand for trained technicians will actually *double*.

You can cash in on this opportunity, once you've learned all the basics of computers the NRI way. NRI's practical combination of "reason-why" theory and "hands-on" building skills starts you with the fundamentals of electronics, then guides you through advanced electronic circuitry and on into computer electronics.

You Build—and Keep—a 16-bit Sanyo personal computer

The vital core of your training is the step-by-step building of the 16-bit Sanyo MBC-550 series computer. Once you've mastered the details of this IBM-PC compatible machine, you'll be qualified to service and repair virtually every major brand of computer, plus many popular peripheral and accessory devices.

With NRI training, you learn at your own convenience, in your own home. You set the pace—without classroom pressures, rigid night-school schedules, or wasted time. You build the Sanyo IBM compatible computer from the keyboard up, with your own personal NRI instructor and the complete NRI technical staff ready to answer your questions or give you guidance and special help whenever you need it.

Your NRI course includes installation and troubleshooting of the "intelligent" keyboard, power supply, and disk drive, plus you'll check out the 8088 microprocessor functions, using machine language. You'll also prepare the interfaces for future peripherals such as printers and joysticks.

100-Page Free Catalog Tells More

Send the coupon today for NRI's big 100-page color catalog on electronics training, which gives you all the facts about NRI courses in Microcomputers and other growing high-tech career fields. If the coupon is missing, write to NRI Schools, 3939 Wisconsin Ave., NW, Washington, D.C. 20016.

Your NRI course includes the Sanyo 16-bit IBM compatible computer with 128K RAM, monitor, double density/double sided disk drive, and "intelligent" keyboard; the NRI Discovery Lab®, teaching circuit design and operations; a Digital Multimeter; Bundled Spread Sheet and Word Processing Software worth over $1000 at retail—and more.

NRI SCHOOLS
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue, Washington, DC 20016
We'll give you tomorrow.

For Career courses approved under GI bill, ☐ check for details.

☑ **CHECK ONE FREE CATALOG ONLY**
- ☐ **Computer Electronics with Microcomputers**
- ☐ Data Communications
- ☐ Robotics & Industrial Controls
- ☐ Video Electronics Servicing
- ☐ Electronic Design Technology
- ☐ Digital Electronics
- ☐ Satellite Communications
- ☐ Communications Electronics
- ☐ Industrial Electronics
- ☐ Basic Electronics
- ☐ Telephone Servicing
- ☐ Small Engine Servicing
- ☐ Building Construction

Name (Please Print) ______ Age ______
Street ______
City/State/Zip ______

Accredited by the National Home Study Council

198-056

www.commodore.ca

almost identical to *AtariWriter's*. It defaults to a width of 38 columns and 20 lines. Below this area is a three-line command window that displays the tab stops, a pair of counters for the cursor position (column and line), system messages, and other information.

New additions to the command window are indicators for the typing mode, caps lock mode, and number of bytes free. The typing mode determines whether newly typed characters make room for themselves by pushing existing characters forward, or whether they simply replace existing characters. *AtariWriter* was limited to insert mode, which irked some people. *AtariWriter Plus* defaults to insert mode, but offers the strikeover mode as well. The caps lock indicator is only semi-useful, since you'll quickly discover if you've accidentally hit this key anyway. The free memory indicator is a little more practical, although it was only a keystroke away in the old *AtariWriter*.

By pressing OPTION-C, you can change the default 38-column screen to any width from 5 to 249 columns. Of course, no more than 40 columns can be displayed at a time, so the screen scrolls horizontally to bring additional ones into view. This makes it much easier to line up columns of numbers and the like. Since this setting can be altered at any time, you can type in 38- or 40-column mode if you find horizontal scrolling distracting, and then switch to a wider screen to check the alignment of your columns.

The column counter in the command window always displays the cursor position relative to the screen's full width, so it's easy to keep track of your location on a wide screen. However, the line counter doesn't do likewise—it stops at 20 when you reach the bottom of the writing/editing screen, no matter how far down you scroll through your document.

You can zip through your text a little faster in *AtariWriter Plus* because the SELECT and cursor-arrow keys let you move a whole word at a time left or right. Vertical movement is much quicker because the OPTION up-arrow and down-arrow keys now flip screens instantly rather than scrolling them slowly as in the old *AtariWriter*. And incidentally, the cursor itself has been changed to a blinking block instead of the blinking underline that was its precursor's cursor.

Memory In The Bank

If you're using a 48K or 64K machine, *AtariWriter Plus* leaves 12,645 bytes free for text (roughly 2,100 words). That's not much compared to other word processors for the Atari. Most of them, including *AtariWriter*, leave at least 20K free.

If you're using a 130XE, the news is a little better. The 130XE version on the flip side of the *AtariWriter Plus* program disk takes advantage of the extra RAM to give you a total of 47,616 bytes free (roughly 7,900 words). But there's a catch—the program usually doesn't see this memory as one, continuous "blank page." Instead, the memory is divided into three sections called banks. The technical reasons for this are beyond the scope of this review, but they're related to the memory-addressing limitations of eight-bit computers. Although it's possible to get around these limitations, it's a programmer's nightmare.

As a result, *AtariWriter Plus* gives you three banks of 15,872 bytes on a 130XE. When you save and load from disk, the program *does* treat the banks as one continuous block of memory: a long document that occupies two or three banks is saved and loaded as a single file. Almost all other functions, however, require you to deal with each bank separately. For instance, to switch from bank to bank, you must press START-B. To move the cursor to the top or bottom of the file, you can't simply press SELECT-T or SELECT-B as usual; these commands move the cursor only to the top or bottom of the current bank. You can move blocks of text from one bank to another, but you can't define a text block that crosses a bank boundary. Search and replace operations won't bridge the bank boundaries either, but the search and replace strings remain intact so you can continue the operation after switching banks.

Another limitation is that you can't merge a file across banks. In other words, if you've got a 10,000-byte file in memory, you can't merge it with another file that is 6,000 bytes long or more without exceeding the banks' 15,872-byte capacities. (If you can merge the two files outside of the program—perhaps with DOS—you can load the joined file as a single document, though.)

One interesting command (OPTION-F) redistributes your text equally among all three banks. For instance, if you fill up bank 1, continue writing in bank 2, and later decide to insert a paragraph in bank 1, you can press OPTION-F to free up some memory. Your document will be split across three banks—somewhat awkward, but at least you'll have room for your insert. When you load a long file that won't fit in a single bank, this redistribution happens automatically.

Other Enhancements

Like *AtariWriter*, *AtariWriter Plus* preserves deleted blocks of text in a failsafe buffer so you can restore them or paste them elsewhere in the document. But while *AtariWriter's* failsafe buffer has only enough memory for less than two screenfuls of text, *AtariWriter Plus* sets aside all remaining text memory for the buffer. That means the failsafe buffer can range in size from more than 12,000 bytes to 0 bytes, depending on how much you've written. A new command (START-E) lets you erase the buffer if it gets too full to let you continue writing.

The 130XE version, however, always sets aside about 8K for the failsafe buffer.

Defining text blocks is handled a little differently, too. Instead of marking the beginning and ending points of a block with CTRL-X, as you do in *AtariWriter*, you mark the beginning with OPTION-B. When you move the cursor, the characters you're defining as a block become highlighted in inverse video. Then you mark the end of the block with a different command that depends on which operation you want to perform. For example, to indicate the end of a text block you want to delete, you press OPTION-DELETE/BACK SPACE. A DELETE BLOCK Y/N? message asks you to confirm your choice.

AtariWriter Plus lets you perform several other operations on text blocks as well. OPTION-W counts the number of words in a block. (When you press OPTION-W without defining a block, the program counts all the words in your document—or in the current bank on a 130XE.) OPTION-A alphabetizes all the words in a block. OPTION-S saves the block on disk. And OPTION-E copies the block into the failsafe buffer so you can paste it elsewhere.

Another new feature is form printing. Let's say you need to print out a number of form letters with different names, addresses, or other information inserted at certain points. You can indicate those points by pressing OPTION-INSERT; an arrow appears on the screen. Later, the printer will pause at those points and let you type up to 35 characters.

So many other improvements have been added to *AtariWriter Plus* that we can't cover them all in detail. For instance, you can search forward and backward, search and replace control characters and specify wildcards in search strings, turn the alert beep on and off (except for keystrokes and error messages), print text in double columns for newsletters, and construct your own custom printer drivers with a BASIC program that's included. Furthermore, the 80-column (horizontal-scrolling)

www.commodore.ca

PRINTERS

EPSON

LX90	$234.95
HS80 Letterjet	318.95
JX80	448.95
FX185	454.95
LQ1500 (PAR)	949.95
FX85	331.95
LX80	210.95

STAR MICRONICS

SG10	$210.95
SG10C	234.95
SG15	368.95
SD10	321.95
SD15	440.95
SR10	468.95
SR15	581.95
Power type	299.95

CITIZEN

MSP10	$274.95
MSP15	438.95
MSP20	438.95
MSP25	548.95

PANASONIC

1091	$231.95
1092	368.95
1093	424.95
3151	424.95
3131	259.95

OKIDATA

Okimate 10	$170.95
Okimate 20	209.95
182	219.95
192	349.95
193	514.95
84	640.95

LEGEND

808/NLQ	$149.95
1080 NLQ	205.95
1380	259.95
13805	294.95

JUKI

6000	$189.95
6100	379.95
6300	769.95

TOSHIBA

P351	$1289.95
1340	599.95

DAISYWRITER

2000	$739.95

N.E.C.

2030	$699.95
2050	699.95
3530	1309.95
3550	1389.95

MODEMS

ANCHOR

Volkmodem 12	$179.95
Signalman Express	299.95
Lightning 2400	399.95
Westridge	59.95

DIGITAL DEVICES

300 Baud (Atari)	$98.95

HAYES

300	$137.95
300/1200	386.95
1200 B w/software	378.95
1200B	349.95

NOVATION

Smart Cat Plus (1200)	$309.95
Professional 2400	629.95

SUPRA

MPP1000E (Atari)	$54.95
MPP1064(C-64)	54.95

CAL-ABCO

Smart Team 1200	$199.95

C.D.I.

Mitey Mo (64)	$59.95

INTERFACES

SUPRA

MTP1150	$45.95

DIGITAL DEVICES

U Print/Port	$49.95
U Print/16K	69.95
U Print/64K	79.95

CARDCO

G-Wiz	$48.95
AT-1 (Atari)	48.95

TYMAC

Epson	$58.95
PAR (Centronics Std.)	54.95

DISKETTES

	5¼	3½
BONUS		
SS/DD	$ 9.50	—
DS/DD	13.50	—
NO LABEL (with Pen & Flip-n-file)		
SS/DD	$10.50	—
DS/DD	14.50	—
MAXELL		
MD1	$15.95	32.95
MD2	19.95	38.95
MEMOREX		
SS/DD	$12.50	29.95
DS/DD	16.50	34.95
FF30/20SS/DD	—	59.95
FF50/20SS/DD	36.95	—
FF50/20DS/DD	42.95	—

All Diskettes Carry a Lifetime Warranty

MONITORS

PRINCETON

Max 12E	$189.95
HX-9	459.95
HX-9E	519.95
HX-12	469.95
HX-12E	555.95
SR-12	589.95
SR-12E	649.95

NEC

1201 (Green)	$99.95
1205 (Amber)	99.95
1280 (TTL Green)	148.95
1285 (TTL Amber)	148.95
1460	269.95
1401 (R.G.B.)	659.95

ZENITH

ZVM 122/123	$ 74.95
ZVM 124 (TTL Amber)	149.95
ZVM 131 (RGB)	279.95
ZVM 135 (RGB/Color)	459.95

TEKNIKA

MJ-10	$178.95
MJ-22	254.95

AMDEK

300G	$116.95
300A	126.95
310A	144.95
Color 300	175.95
Color 500	339.95
Color 600	398.95
Color 700	468.95
Color 710	538.95
Color 722	579.95

BOARDS

AST

Six Pack Plus	$239.95
Advantage (AT)	399.95
Graph Pak	589.95
Preview	299.95
Mono Graph Plus	389.95
5251/11	799.95
5251/12	579.95

EVEREX

Color Card	$295.95
Magic Card	185.95

HERCULES

Color	$169.95
Graphics	289.95

PARADISE

Five Pack	$159.95
Mono	149.95
Modular Graphics	279.95
Multi Display	209.95

TECMAR

Graphics Master	$459.95
Captain 64	$189.95

DISK DRIVES

INDUS

GT (Atari)	$198.95
GT (Commodore)	218.95

TANDON

TM100-2	$109.95

TEAC

55B	$104.95

WESTERN REGION 1-800-351-3455 — **ORDER TOLL FREE** — **EASTERN REGION 1-800-351-3442**

In CA Call: 1-800-351-3422

"Where Prices are Born, Not Raised."

WHITE HOUSE COMPUTER

WESTERN REGION
11327 Trade Center Drive
Suite 335
Rancho Cordova, CA 95670
Customer Service: 916-635-3455

No deposit on C.O.D. orders. Free freight on all prepaid cash orders over $300 in the Continental U.S.A. APO and FPO orders add $5.00 per hundred. For Priority Mail add $10.00 per hundred. Free shipping for PA residents. PA residents add 6% sales tax. All defective products must have a prior RA number.

Hours: Monday thru Friday 9 a.m. - 6 p.m.

EASTERN REGION
P.O. Box 4025
Williamsport, PA 17701
Customer Service:
717-322-7700

MASTER CARD 4% VISA 4% AMERICAN EXPRESS 5%

The 1050 DUPLICATOR™ IS HERE...

THE 1050 DUPLICATOR™: The most powerful diskdrive copy system ever developed for the ATARI.

520 ST Duplicator Now available!

The only Copy System You will ever need!
What will it do?

►***The main purpose of the Duplicator is to copy disks!*** *You will be able to copy just about any disk! The copies you make will run on any Atari drive. The Duplicator need not be present to run your backup copies. The Duplicator is fully automatic. You need only insert source and destination disks. Custom formats will be read and in turn reproduced on the backup copy disk. Our device will reproduce any custom format or heavily copy guarded scheme, bad sectors, double sectors, 19 through 24 sector format will present no problem to the Duplicator.*

►***You will still have single density, density and one half, and double density.*** *When you have a Duplicator installed in a 1050 drive that drive will be turned into true double density. You will have twice the disk storage. Your drive will be compatible with other double density drives as The Rana Indus, Percom, etc.*

HARDWARE POWER

►***High speed read & write.*** *Your disk drive will read and load all of your software, saving wear and tear on your drive. The 1050 drive now reads one sector at a time. This is slow and inefficient. With the duplicator installed you will be able to read eighteen sectors in the time it takes standard, unenhanced drives to read one.*

►***Included with every Duplicator will be user friendly disk software.*** *A simple, menu driven program will allow you to copy all of your software. A Duplicator enhanced drive will be a SMART drive. We plan to write many new and exciting programs that can only be run on an enhanced drive, eg. sending a copy-guarded disk over the phone. Since the drive is now fully programmable, future upgrades can be made available to you on disks, should the need arise. No further hardware changes will ever be needed. The Duplicator comes with a full hardware and software guarantee.*

Fully Compatible with the XL & New XE Series.

Only $149.95

Plus $3.50 for shipping handling.
Add 7% outside U.S.A.
N.Y. State Residents add 7½% Sales Tax.
**Dealer inquires are welcome, call for quantity price quote.*

EASY 5 MINUTE INSTALLATION

NO HARM TO YOUR DRIVE OR INCOMPATIBILITY PROBLEMS CAN EVER ARISE AS A RESULT OF THE INSTALLATION OF OUR DUPLICATOR.

IMPORTANT: Only a hardware device like the DUPLICATOR can backup heavily copy-guarded disks. Don't be fooled by software programs that claim to do this.

DUPLICATING TECHNOLOGIES inc.
Formerly Gardner Computing

99 Jericho Tpke., Suite 302A Jericho N.Y. 11753 Order Business Hrs. (516) 333-5805, 5807, 5808 Order Eve's. and Weekends (516) 333-5950

TERMS: We accept American Express, Visa, MasterCard and C.O.D. orders. Foreign orders must be in U.S. dollars. All personal checks allow 14 days to clear. Most items shipped within 24 hours.

www.commodore.ca

print-preview feature has been improved: It now can show foreign-language characters if you have an XL or XE, and it even displays a preview of double-column printing.

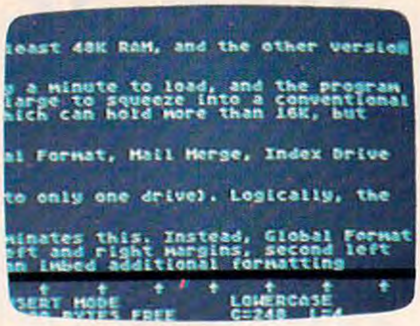
Editor On A Disk?

One of the most significant additions to *AtariWriter Plus* is *Atari Proofreader,* a spelling-checker program. Although it's integrated with *AtariWriter Plus,* it's not memory-resident. That means when you select Verify Spelling from the main menu, you have to insert the program disk and wait a half-minute or so as *Atari Proofreader* replaces *AtariWriter Plus* in memory. The same procedure is necessary when returning to the word processor from the spelling checker. Fortunately, your text file is preserved in memory during this exchange.

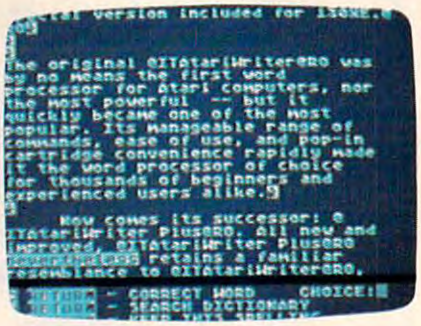
Once *Atari Proofreader* is up and running, you insert the dictionary disk and choose from several options (on a two-drive system, less disk-swapping is required). Highlight Errors checks the whole document and highlights in inverse video any words the *Proofreader* doesn't recognize; Correct Errors also highlights unrecognized words, but then pauses and lets you type in the correct spelling; Print Errors proofreads the document and dumps all unrecognized words on a printer; Dictionary Search lets you look up words on the dictionary disk by typing in your best guess as to how they're spelled; File Personal Dictionary saves on disk a short dictionary of your own special words; Add Personal Dictionary loads one of these special dictionaries from disk; Index Drive 1 or 2 calls up disk directories; Load File loads a document for proofing; Save File saves a proofed document; and Return To AtariWriter Plus exits the spelling checker back to the word processor.

Like all spelling checkers, *Atari Proofreader* lets you add your own list of special words to the dictionary. This keeps the checker from highlighting certain technical terms or proper names that you frequently use. Building a personal dictionary is easy. When proofing a file with the Correct Errors option, you can press a key that tells the checker to remember a word it didn't recognize. Later, you can save these words on disk as a personal dictionary. You can also build a dictionary by simply creating a new file with *AtariWriter Plus* and typing in a word list.

The personal dictionaries are fairly limited, however. *Atari Proofreader* remembers only the last 256 words it highlighted when checking a file. The maximum space available for a personal dictionary is 8,400 bytes on a 48K or 64K Atari—and, oddly, only 4,396



The writing/editing screen of Atari-Writer Plus *can now scroll horizontally as wide as 249 columns.*



Atari Proofreader, *another new feature of* AtariWriter Plus, *checks your spelling against a 36,000-word dictionary and highlights any mistakes.*

bytes on the 128K RAM 130XE. Since the average English word is about six letters long, that breaks down to about 1,200 words on a 48K/64K machine, or about 628 words on a 130XE. Of course, the kind of words you'll include in a personal dictionary will probably average longer than six letters, so these word counts may be misleading. To get around this limitation, though, you can create several personal dictionaries and proof a document in more than one pass.

However, it takes *Atari Proofreader* quite a while to check a large document—mainly because it must read the dictionary disk constantly when proofing a file (a good reason for backing up the disk). This 3,000-word review, written with *AtariWriter Plus* on a 130XE with a 1050 disk drive, took the *Proofreader* 15 minutes to check. Of course, even a good copy editor (the kind that doesn't come on a disk) would take at least that long.

Mail Merge

Another integrated program is *Atari Mail Merge.* Like *Atari Proofreader,* it must be loaded from disk and swapped in memory with *AtariWriter Plus.* It's a general-purpose filing program that lets you compile a list of names and addresses (or any other kind of information, for that matter) and merge them into form letters.

Mail Merge can handle a file up to 22,292 bytes long. This file can include as many as 255 records, and each record can contain up to 15 fields of 20 characters each. When creating a file, you can design your own record format or use the default format. You can then edit, append, and print these files with *Mail Merge.*

To use *Mail Merge* with *AtariWriter Plus,* you press OPTION-M at the point in your document where you want to insert information from *Mail Merge.* A heart character appears on the screen, and you follow it with a number that indicates which field you want printed (1 to 15). When you print the document, the information from that field is inserted into your text. This is a particularly useful feature for printing out form letters and address labels.

You can also define a subset of a *Mail Merge* file. For instance, you could send form letters to only ten people out of a list of 100.

Integrated features like *Mail Merge* and the *Proofreader* elevate *AtariWriter Plus* to the upper rank of word processors for Atari computers. It's still not the most powerful or full-featured word processor you can buy for your Atari. But like its popular predecessor, *AtariWriter Plus* strikes a good balance between versatility and ease of use.

AtariWriter Plus
Atari Corp.
1196 Borregas Avenue
Sunnyvale, CA 94088
$49.95

Borrowed Time

Selby Bateman, Features Editor

Requirements: Commodore 64 or 128; Apple II-series computer with at least 64K RAM; IBM PC with color/graphics adapter and at least 64K RAM; Enhanced Model IBM PCjr; Atari ST-series computer; Commodore Amiga; or an Apple Macintosh. Disk only; color monitor recommended.

Your name's Harlow...Sam Harlow, private eye.

You're sitting in your office with your feet propped up, trying to figure out how to pay the rent, when the phone rings. "Sam, they want you dead," says a voice at the other end of the line. The voice fades, and you get the funny feeling someone's watching you. Before you know it, people—all kinds of people—are doing more than watching. They're coming after you. And you're living on borrowed time.

www.commodore.ca

Your secretary, Iris Spencer, has taken a powder; Jerry the bartender freezes when you ask about Farnham; Hawkeye, the blind newspaper vendor, has a tip about Fred Mongo; Dave, who runs the corner hotdog stand, thinks you're trying to pin a hamburger rap on him. He keeps throwing hotdogs at you. And now someone's kidnapped your wife, Rita.

Life's tough in the big city for a down-in-the-heels gumshoe, and never tougher than in Activision's graphic-and-text adventure, *Borrowed Time*. But you don't have an opportunity to feel sorry for yourself. People keep breaking down your door, running after you in dark alleys, and unloading .38s in your direction. It's up to you, Sam. Find out who's sending all those thugs after you, and quickly. One more thing, Sam. You'd also better find out why.

Fun, Not Frustrating

In *Borrowed Time*, Activision has created a delightful game environment with the look and feel of those classic hard-boiled detective movies and novels. The game is also fun to work with, easier and faster than earlier graphics-and-text games (especially the ST and Amiga versions). Some computer adventures can be frustrating, limiting your options with complex puzzles to such an extent that just leaving a room can take hours of problem-solving. *Borrowed Time* offers a plot line and puzzles that are intriguing and challenging without demanding that you enroll in a code-breaking class.

The screen format and game movement are very well executed in *Borrowed Time*. The screen is divided into six sections: a graphics window showing scenes representing the action described in the text; a scrolling text window along the bottom third of the screen; an inventory window displaying what you're carrying; a compass showing north, south, east, and west; and two lists of words, verbs on the left and nouns on the right.

Using a joystick or mouse (depending on the computer), you can quickly select the direction you wish to travel, choose verb-noun commands from the lists, and even examine or drop what you're carrying by pressing the mouse or joystick button. Of course, you can still type in all the commands if you'd prefer, and the game's vocabulary is much more extensive than just the couple of dozen words listed on the screen at any one time. With the mouse or joystick, and especially with the fast disk access of computers like the ST, Amiga, and Macintosh, you can travel very rapidly from place to place with a minimum of typing. There is, naturally,

The Atari ST version of Borrowed Time *has the best graphics of all.*

much disk access. Depending on the computer, this can be frustratingly slow (Commodore 64 and 1541 disk drive) or amazingly swift (Atari ST).

Making a map of your travels through the city is almost mandatory. There are many different locations, and an engaging (and often dangerous) cast of characters. You can usually converse with the people you see, some of whom may not always tell you the truth. Just like any good private eye, you'll want to examine everything and watch the different screens for visual clues to help you solve the mystery or escape from tight spots.

Life Is Tough

As Sam Harlow, you're prone to meet with a lot of "accidents," resulting in the frequent untimely end of the game. But Activision has provided both a QUICKSAVE and a QUICKLOAD command that helps keep you in the action. If you hear someone pounding on your door, if a shadow suddenly looms behind you, or if there's the quiet click of a hammer being drawn back on a gun, you'd be well advised to use the QUICKSAVE command. Then, should something happen to you, use QUICKLOAD and you're back where you were just before your accident.

The designers of *Borrowed Time* obviously had a lot of fun putting the pieces together. There's a sense of humor in the text, and the visuals can be charming. Clothes hanging on a line ripple in the wind, a dog's tail wags and his tongue peeks out as he pants, the telephone receiver bounces in its cradle as it rings, and occasionally a character will glance at you from the corners of his eyes. As bullets whine over your head, your natural inclination may be to type in the command, DUCK!. All you'll get for that is the response, QUACK!.

In addition to a concise printed explanation of game rules and features, the program also contains a tutorial on the disk which helps first-time players get started. However, so intuitive is the feel of *Borrowed Time* that you can boot the disk and start to play without knowing anything about the game. The program and the graphics can vary slightly from computer to computer, depending on the differing technical capabilities of various machines. But game play appears to be quite similar throughout.

You can have a lot of fun with *Borrowed Time*. Just keep checking over your shoulder, keep moving, and expect the worst.

Borrowed Time
Activision, Inc.
2350 Bayshore Frontage Road
Mountain View, CA 94043
$29–$44 (depending on the version)

Europe Ablaze For Commodore And Apple

Neil Randall

Requirements: Commodore 64 (or 128 in 64 mode); or an Apple II-series computer with at least 64K RAM. Disk only.

Europe Ablaze is the third game from the Strategic Studies Group of Australia. Their first effort, *Reach for the Stars*, remains one of the most popular computer wargames ever released, and their second, *Carriers at War*, won the Charles Roberts Award at last year's Origins gaming convention for best computer game of the year. Add the fact that SSG's games are now being distributed by Electronic Arts, and we can see that SSG has quickly become one of the leading designers of computer wargames. And when you're at the top, people expect great things.

For those expecting excellence, *Europe Ablaze* will not disappoint. The game is a simulation of the World War II air war over Germany and England. The three scenarios included with the game cover the Battle of Britain, 1940 (the one with Churchill's famous speech); the summer 1943 U.S. and British bombing offensive; and the 1944 attempt to bomb Germany into submission.

Like *Carriers at War*, though, *Europe Ablaze* is not restricted to the scenarios provided. A detailed but remarkably easy scenario design kit allows you to alter the existing scenarios or add new ones of your own. One such scenario, which covers the Mediterranean bombing campaign of 1943, is

www.commodore.ca

included in the design book. With sufficient research, historians and air war buffs can work up their own.

Like *Carriers at War, Europe Ablaze* is entirely menu-driven. By simply asking you to type Y or N, the menus guide you through all the game's activities, from formatting a disk to ordering an interception. You can command your squadrons to perform reconnaisance, execute sweeps and raids, or, at the heart of the game, fly bombing missions to the target of your choice. And you don't just send them someplace; you tell them how to get there and what to do. You can target bombs against ports, against communications facilities, against industry, or against population. And you can fly straight to the target or divide the flight into two or three "legs" to confuse the enemy. In other words, you must make the same kinds of choices your historical counterparts had to make.

Two Different Roles

To this end, *Europe Ablaze* is actually two games in one. If you want to avoid the detailed mission and squadron planning, you can assume the role of Commander-in-Chief. In this role, you issue general orders to each of your Air Fleets, and the computer carries out your orders. If you want to take your Commander-in-Chief's orders and carry out the actual missions, you can select the role of Air Fleet Commander. This role is more time-consuming, but because you have more to do it is also more interesting.

You need not restrict yourself to one or the other, however. At the beginning of the game, the program asks you which commands are to be human-controlled and which computer-controlled. If you're playing the game with friends (up to 12), each can take a different command, and the realism multiplies as each commander demands missions and materials to suit his own objectives. *Europe Ablaze* impressively establishes the levels of command; as Commander-in-Chief, you will be helpless once you've issued your orders, and as Air Fleet Commander, you must often make the best of orders you don't agree with. That's the way the Air Force works.

As mentioned earlier, those expecting excellence should be very satisfied by *Europe Ablaze*. But those hoping for innovation may not be. *Europe Ablaze* extends the menu-driven system of *Carriers at War*, smoothing it out and allowing the player a bit more freedom. But sometimes the game plays too much like its forerunner.

Still, we're beginning to see more and more refinements of existing wargame systems, and as long as designers refine—and not simply copy—the result will be games of superior quality. *Europe Ablaze* is a fine simulation that uses a proven game system. We can't ask for much more.

Europe Ablaze
Strategic Studies Group
(Distributed by Electronic Arts)
2755 Campus Drive
San Mateo, CA 94403
$49.95

Ultima IV: Quest For The Avatar For Apple And 64

James V. Trunzo

Requirements: Apple II-series computer with at least 64K RAM (Mockingboard sound enhancer optional); or a Commodore 64 or 128 (in 64 mode). Disk only. The Apple version was reviewed.

Just when you thought it was safe to venture forth into the land of Britannia, along comes *Ultima IV: Quest for the Avatar*. The sequel to *Ultima III* certainly appears to live up to its advance notice. I say *appears* because I've spent only a dozen or so hours adventuring in the world of Britannia, and therefore can't claim to be all-knowledgeable about the surprises lurking in this game. However, considering the tremendous scope of *Ultima IV*, if I waited until completely finishing the game before reviewing it, *Ultima X* would probably be on the market by then. The world of *Ultima IV: Quest for the Avatar* is approximately 16 times larger than that of its predecessors.

But map size is hardly the only difference between *Ultima IV* and the other *Ultimas*. It is a tribute to the designers' programming skills and creative genius that *Ultima IV* quickly establishes its own identity while continuing the by-now familiar play format employed in *Ultima I* through *III*.

While *Quest for the Avatar* retains many of the standard features of the previous programs, it also has a heavy philosophical bent that comes very close to moralizing. At the least, it puts quite an emphasis on virtues that lead to a "good" life: honesty, valor, charity, etc. The player should never lose sight of this emphasis when making decisions throughout the game. Of course, philosophical contemplations are well and good, but they aren't tremendously useful against the aggressive monster-types that beset you during your quest.

Recipes For Magic

Combat and magic are still of major importance in *Ultima IV*. The procedure for combat is unchanged, but a new level of sophistication has been reached for the use of magic. No longer can you simply cast a spell. Instead, reagents (ingredients) must be purchased and properly mixed before using a spell—and woe on you if you're struggling to prepare your magic while five trolls are attacking your party. The Book of Spells lists the ingredients you need, but in many cases it's up to you to discover the proper portions of each ingredient. What does it take to make a fireball—two parts bloodmoss and one part sulfurous ash, or vice versa?

There's more, of course. At the risk of revealing too much, be aware that in *Ultima IV* you do not pick your party of adventurers; they pick you. If you're unworthy of help from a powerful Paladin, for example, you'll have to gain more experience before receiving his aid. Even the creation of your player character is handled in a unique and fascinating way. You no longer simply state your preference for character type and race. I won't tell you what happens instead because it would rob you of one of the initial enjoyments when playing *Ultima IV*. After discovering this for yourself, you'll be glad I was discreet.

The package is similar to previous versions of the *Ultima* series. It includes a cloth map, a Book of Magic, a History of Britannia, and two game disks. All the material is top quality, and the manuals are eminently readable. If you get hopelessly stuck, a hint book will be available containing maps and clues. This makes it possible for less industrious adventurers (like me) to have a chance of completing the game in their lifetimes.

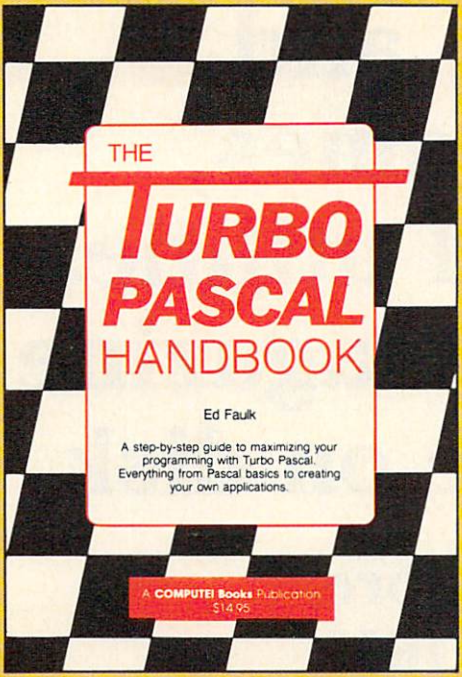
For those who have never experienced any of the *Ultima* games, note that it isn't necessary to have played the forerunners in order to enjoy *Ultima IV: Quest for the Avatar*. However, my bet is that once you play one, you'll want to play them all.

Ultima IV: Quest for the Avatar
Developed by:
Origin Systems, Inc.
Distributed by:
Electronic Arts
2755 Campus Drive
San Mateo, CA 94403
$64

www.commodore.ca

Explore Pascal with

THE TURBO PASCAL HANDBOOK from COMPUTE!.



The Turbo Pascal Handbook
Edward P. Faulk
With *The Turbo Pascal Handbook* and *Turbo Pascal* from Borland International, you'll be gently guided, step-by-step, until you're creating your own powerful applications in this impressive computer language.
$14.95 ISBN 0-87455-037-8

This information-packed book from COMPUTE! is an outstanding resource and programming guide. And it's written in COMPUTE!'s bestselling style so that even beginning programmers can quickly and easily understand all the applications.

Ask for *The Turbo Pascal Handbook* at your local computer store or bookstore. Or order directly from COMPUTE!. Call toll free 1-800-346-6767 (in NY 212-887-8525) or mail the attached coupon with your payment (plus $2.00 shipping and handling per book) to COMPUTE! Books, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

Note: You'll need *Turbo Pascal* in order to use this book. The software is not included with *The Turbo Pascal Handbook.*

Yes! Send me _____ copies of *The Turbo Pascal Handbook* at $14.95 each.
My payment is enclosed.

ALL ORDERS MUST BE PREPAID IN U.S. FUNDS

Subtotal ________
NC residents add 4.5% sales tax ________
Shipping and handling ________
($2.00 per book in U.S. and surface mail; $5.00 per book airmail.)
Total enclosed ________

☐ Payment enclosed (check or money order)
☐ Charge ☐ Visa ☐ MasterCard ☐ American Express
Account No. ________ Exp. Date ________ (Required)
Name ________
Address ________
City ________ State ________ Zip ________
Please allow 4–6 weeks for delivery.
36503711

COMPUTE!™ Publications, Inc. abc
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor, New York, NY 10019
Publishers of COMPUTE!, COMPUTE!'s Gazette, COMPUTE!'s Gazette Disk, COMPUTE! Books, and COMPUTE!'s

www.commodore.ca

All the exciting, entertaining, and educational games, applications, and utilities from COMPUTE! magazine are now available on disk

for your Commodore, Atari, Apple, or IBM personal computer.

The COMPUTE! Disk

A new *COMPUTE! Disk* is published every month, rotating among the four major machines covered by *COMPUTE!*: Commodore 64 and 128; Atari 400/800, XL, and XE; Apple II-series; and IBM PC, PCjr, and compatibles.

Every three months you can receive a disk with all the quality programs from the previous three issues of *COMPUTE!* that will run on your brand of computer.

Like the popular *COMPUTE!'s Gazette Disk*, the *COMPUTE! Disk* is ready-to-load and error-free. It saves you valuable hours of typing time and eliminates typing errors.

With a subscription, you will receive one disk every three months for a total of four disks a year—for only $39.95. That saves you $20 a year off the single-issue cost.

Or you can order individual issues of the *Disk* for $12.95 a disk plus $2.00 shipping and handling.

Remember to specify your type of computer when ordering the *COMPUTE! Disk*. You'll find more information about this month's *COMPUTE! Disk* in this issue. (Note: You'll need the corresponding issues of *COMPUTE!* magazine to use the *Disk* since the disk will have no documentation.)

For fastest service when ordering a subscription to the *COMPUTE! Disk*, call toll free 1-800-247-5470 (in Iowa 1-800-532-1272).

For more details or to order individual issues of the *COMPUTE! Disk*, call our Customer Service Department toll free at 1-800-346-6767 (in New York 212-887-8525).

Please allow 4–6 weeks after placing an order for your first disk to arrive.

COMPUTE! Publications, Inc. abc
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor, New York, NY 10019
Publishers of COMPUTE!, COMPUTE!'s Gazette, COMPUTE!'s Gazette Disk, COMPUTE! Books, and COMPUTE!'s Apple Applications

www.commodore.ca

Adding System Power To ST BASIC

Part 2

Kevin Mykytyn, Editorial Programmer

The VDISYS command lets you fill in the gaps in Atari ST BASIC by calling system routines to do jobs that would otherwise be impossible. Part 1 of this series explained the fundamentals of VDISYS and examined a general-purpose drawing routine. This part shows how VDISYS can perform two other important tasks—reading the screen position of the mouse pointer and sensing the status of the mouse buttons. An example program lets you create a custom shape for your ST's mouse pointer.

Have you ever tried to read the Atari ST's mouse controller from BASIC? If you have, you already know that BASIC lacks commands to read the mouse position or button status. Like certain other tasks, mouse reading can be done from BASIC only with the aid of VDISYS. Once you know how to read the mouse, you may also want to change the mouse pointer's appearance. This article explains how to do both.

Before you type in the example program below, here are some tips that make it easier to enter ST BASIC programs. First, although it may be obvious to some of you, it is far easier to enter a program from the Edit window than the Command window. (To move to the Edit window, type EDIT at the Command window's OK prompt or choose the Start Edit option from the Edit menu.) The Edit window's full-screen editor is much more convenient for entering program lines than the Command window's single-line interface. You can also run a program directly from the Edit window (type RUN or choose the Start option from the Run menu). When the program is finished, control returns to the Edit window, so you can immediately modify or add new lines to the program.

The Edit window has one feature that you may or may not appreciate. Until you press RETURN, the line you're working on appears in *ghost mode* (the letters look gray and fuzzy). The purpose of ghost mode is to show which lines you have changed. That's helpful to inexperienced programmers, but an annoyance in many cases, since ghosted letters are harder to read than normal ones. To disable ghost mode, enter this line in the Command window:

POKE SYSTAB+2,0

Another way to ease the task of program entry is to increase the speed of the cursor. This is done from the Control Panel. The second slider from the top (the one with a rabbit and a turtle) controls the cursor speed. To increase the speed, click on the slider and drag it to the left (toward the rabbit). To slow it down, drag the slider to the right. You can also turn the keyboard beeping sound off and on by clicking the C key icon in the Control Panel.

Redesigning The Pointer

Two more steps are required before typing in the pointer-editing program. First, set the computer to medium resolution (use the Set Preferences option). Second, turn off buffered graphics from BASIC's Run menu. If your ST has more than 512K of Random Access Memory (RAM) or the TOS operating system in Read Only Memory (ROM), the second step may or may not be necessary, but it can't hurt in any case.

Now enter Program 1 and save it to disk. It lets you change the mouse pointer from the familiar arrow shape to a custom design of your own. When you run the program, a grid appears on the left side of the screen, and the word DONE is shown on the right. To edit the pointer shape, move the mouse pointer into the grid, then click the button on any square you want to change. Clicking on a square toggles it on or off—if the square is on

www.commodore.ca

(dark) when you click, it is turned off (erased) and vice versa.

Once you're satisfied with the new pointer, move the mouse out of the grid and click on the word DONE. The program then asks for the location of the new pointer's *hot spot*—a single dot that the computer uses to tell exactly what the pointer is pointing at. On the normal mouse pointer, the hot spot is located at the very tip of the arrow. But you can place it anywhere within your custom pointer shape.

After you locate the hot spot, the new pointer appears on the screen. At this point, the program asks whether you want to save the pointer shape data to a disk file for later use. If so, press Y and enter a filename when prompted. If you press any other key, the program ends without saving the shape. Program 2 (see below) provides a method for reloading the shape data from the disk file and making the custom pointer appear in a BASIC program of your own.

Reading The ST's Mouse

If you're unfamiliar with the usage of VDISYS, CONTRL, and PTSIN, you may want to read Part 1 of this article (April COMPUTE!) before going any further. It explains the fundamentals of calling VDI routines from ST BASIC.

As mentioned earlier, ST BASIC has no commands to read the mouse or the state of the mouse buttons directly. Fortunately, there is a VDI routine (appropriately named Readmouse) which gives this information. Only three steps are needed to call this routine. Since Readmouse has an opcode of 124, we first execute POKE CONTRL,124 to tell the ST which VDI routine to call (line 90 in Program 1). This routine doesn't involve any vertices or attributes, so CONTRL+2 and CONTRL+6 are POKEd with zeros. Once that brief preparation is complete, the statement VDISYS(0) actually calls the routine.

As you may recall, Part 1 explained how to pass information from BASIC to a VDI drawing routine. When that routine was done with its work (drawing a graphic shape), we didn't care whether it passed any information back in the other direction. But many VDI routines pass significant information back to BASIC. Thus, calling a routine like Readmouse involves a two-way information transfer. You must supply certain data before calling the routine; and when it returns control to BASIC, the routine sends other information back to you.

Part 1 also explained how the parameter blocks named PTSIN and INTIN are used to pass data from BASIC to a VDI routine. These parameter blocks are paralleled by PTSOUT and INTOUT, which perform the same operations in reverse. Though they're considered reserved variables (which you can use only in certain, predefined ways), PTSOUT and INTOUT each point to a block of special storage locations in memory called a *parameter block*. Like PTSIN, PTSOUT points to a temporary holding area for information about x and y position coordinates. Like the INTIN parameter block, INTOUT defines the area where other information (attribute data, etc.) is passed.

Position And Button Status

To read the mouse pointer's screen location, call the Readmouse routine and PEEK the memory locations defined by PTSOUT and PTSOUT+2. In Program 1, this is done at line 110. The statement X=PEEK(PTSOUT) transfers the value stored in PTSOUT in the variable x, representing the mouse pointer's horizontal position. Similarly, Y=PEEK(PTSOUT+2) makes y equal to the mouse pointer's vertical position.

To read the status of the mouse buttons, call the Readmouse routine and PEEK the locations defined by INTOUT and INTOUT+2. INTOUT returns information about the left button, and INTOUT+2 tells you the status of the right button. If a button is pressed, the value in the corresponding location is 1; if it's not pressed, the value is 0. Program 1 reads both mouse buttons at line 120. When the left button is pressed, the variable LBUTTON is set to 1; when the right button is pressed, the variable RBUTTON is set to 1. Table 1 outlines the information you need to use Readmouse.

Customizing The Pointer

Though the ST's familiar arrow pointer may be suitable most of the time, occasionally you might want it to look like something else. In a drawing program, for instance, why not reshape the pointer as a pencil or a paintbrush? Once you know how to modify the pointer's appearance, you can make it look like a pointing hand, a musical note, a scientific symbol, or whatever else is needed to give your program that unique, customized look.

The VDI routine that redraws the mouse pointer is called Set Mouse Form, usually abbreviated as SMF. Because the SMF routine requires a lot of information, its setup procedure is fairly complex. The first step, as always, is to POKE the opcode for the VDI routine into CONTRL. Since the opcode for SMF is 111, Program 1 performs POKE CONTRL,111 at line 210. Next, you must POKE the number

Table 1: Readmouse Parameters

Input Parameters	
POKE CONTRL,124	(opcode)
POKE CONTRL+2,0	(number of vertices)
POKE CONTRL+6,0	(number of attributes)
Output Parameters	
PEEK(PTSOUT)	(horizontal mouse position)
PEEK(PTSOUT+2)	(vertical mouse position)
PEEK(INTOUT)	(1 = left button pressed)
PEEK(INTOUT+2)	(1 = right button pressed)

www.commodore.ca

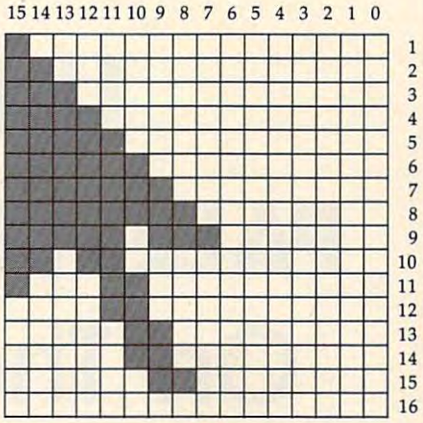
of vertices (0 in this case) and the number of attributes (37) into CONTRL+2 and CONTRL+6 (lines 210–220).

The mouse pointer can move anywhere on the screen, so there's no need to provide x and y coordinates for the shape as a whole. However, you must tell the system where, within that shape, it should put the hot spot. The hot spot's coordinates are defined relative to the upper-left corner of the new pointer shape. POKE the x coordinate value in INTIN and the y coordinate in INTIN+2. At the same time, you should also POKE a 1 into INTIN+4 (lines 220–230).

Who Was That Masked Mouse?

The mouse pointer you see on the screen is actually made of two separate parts—the pointer shape itself and a second shape called a *mask*. Both forms are the same size (16 pixels high and 16 pixels wide) and appear at the same place on the screen. Since the pointer and the mask can be different colors, this permits you to make a two-color mouse pointer. To create the illusion of solidity, for instance, you might draw the main body of the pointer in one color and add a darker shadow along its lower edges. To set the mask's color, POKE the desired color number into location INTIN+6. POKE the pointer's color into INTIN+8.

After you've defined the colors, you must supply shape information for both the pointer and the mask. Each shape requires 32 bytes (16 words) of data. The figure below illustrates how the 16 words of shape data go together to make up the entire shape.



Mouse Pointer Data

If you visualize the pointer shape within a 16 × 16 grid, the first 16-bit data word is in the top row of the grid, the second data word represents the second row, and so on. To pass the shape information to the SMF routine, you must first calculate the 16-bit values represented by the "on" bits within this grid. When that's done, the data for the mask is POKEd into locations INTIN+10, INTIN+12 ... INTIN+40. The pointer shape data is POKEd into locations INTIN+42 through INTIN+72. Don't be concerned if that sounds a bit confusing. Program 1 does all the calculations and POKEs for you automatically. For those who are interested, Table 2 outlines the information needed by the SMF routine.

Table 2: Set Mouse Form (SMF) Parameters

Input Parameters

POKE CONTRL,111	(opcode)
POKE CONTRL+2,0	(number of input vertices)
POKE CONTRL+6,37	(number of attributes)
POKE INTIN,X	(X=hot spot horizontal coordinate)
POKE INTIN+2,Y	(Y=hot spot vertical coordinate)
POKE INTIN+6,mask color	
POKE INTIN+8,pointer color	
POKE INTIN+10–INTIN+40,mask shape data	
POKE INTIN+42–INTIN+72,pointer shape data	

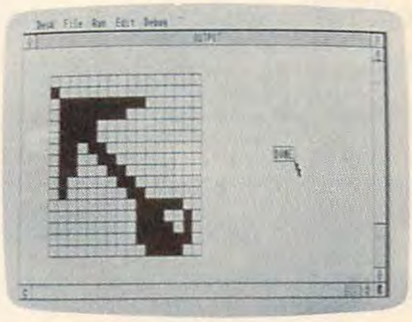
Saving Custom Pointers

Once you've created a custom pointer with Program 1, it appears on the screen and works just like the normal one. However, the pointer reverts to its usual shape as soon as you exit BASIC.

To help you incorporate custom pointers in your own BASIC programs, Program 1 lets you save all the pointer shape data in a disk file. Program 2 illustrates how to read the shape data from the disk file and recreate the custom pointer in another program.

The first two data items in the disk file are x and y coordinates for the pointer's hot spot. The next 16 data items are the 16 words (32 bytes) of shape data for the pointer and mask forms. After this data has been retrieved, it's simply a matter of performing the setup and calling the SMF routine just as we did in Program 1.

To incorporate this routine in your own program, replace the INPUT statement in line 10 with FN$="FILENAME" using your own filename in place of FILENAME. Of course, you could also convert the shape data into DATA statements contained in the program itself.



This drawing grid (created by Program 1) lets you create custom mouse pointers for use in your own ST BASIC programs.

Program 1: ST Mouse Pointer Editor

```
10 FULLW 2:CLEARW 2:COLOR 1,1,1,1
20 DIM AR(16,16),SHAPE(30):FOR A=
   1 TO 16:FOR B=1 TO 16:AR(A,B)=0:
   NEXT B,A
30 FOR A=50 TO 306 STEP 16:LINEF A,
   20,A,148:NEXT
40 FOR A=20 TO 148 STEP 8:LINEF 50,
   A,306,A:NEXT
50 GOTOXY 48,8:PRINT "DONE":Q=43
   0:R=72:LINEF Q,R,Q+36,R:LINEF Q+
   36,R,Q+36,R+10
60 LINEF Q+36,R+10,Q,R+10:LINEF Q,
```


www.commodore.ca

```
   R+10,Q,R
70 GOSUB READMOUSE:IF LBUTTON
   =TOGGLE THEN 70 ELSE TOGGLE
   =LBUTTON
80 IF LBUTTON THEN GOSUB FLIP:GO
   TO 70 ELSE GOTO 70
90 READMOUSE: POKE CONTRL,124
100 POKE CONTRL+2,0:POKE CONTR
   L+6,0:VDISYS (0)
110 X=PEEK(PTSOUT):Y=PEEK(PTSOU
   T+2)
120 LBUTTON=PEEK(INTOUT):RBUTT
   ON=PEEK(INTOUT+2)
130 RETURN
140 FLIP: GOSUB LOCATE
150 IF XP>24 AND XP<27 AND YP=
   8 THEN GOTO DEFINEIT
160 IF XP<1 OR XP>16 OR YP<1 OR YP
   >16 THEN RETURN
170 IF AR(XP,YP) THEN AR(XP,YP)=0:C
   OLOR 1,0:GOTO 190
180 AR(XP,YP)=1:COLOR 1,2
190 FILL XP*16+44,YP*8+18:RETURN
200 DEFINEIT: GOTOXY 43,12:PRINT "
   CHOOSE HOT SPOT":GOSUB HOT
   SPOT
210 POKE CONTRL,111:POKE CONTRL
   +2,0
220 POKE CONTRL+6,37:HX=XP:HY=
   YP
230 POKE INTIN,XP-1:POKE INTIN+2,
   YP-1:POKE INTIN+4,1
240 POKE INTIN+6,0:POKE INTIN+8,1
250 FOR A=10 TO 40 STEP 2:T=0
260 FOR B=16 TO 1 STEP-1:T=T-2^(1
   6-B)*(AR(B,A/2-4)=1):NEXT
270 POKE INTIN+A,T:POKE INTIN+A
   +32,T:SHAPE((a-10)/2)=t
280 NEXT:VDISYS (0):CLEARW 2:GOTO
   XY 32,0:PRINT "DEFINED"
290 PRINT "Do you want to save this sh
   ape?":A=INP(2):IF A<>121 THE
   N END
300 INPUT "Filename";FN$:OPEN "O",#
   1,FN$:PRINT #1,HX;HY
310 FOR A=0 TO 15:PRINT #1,SHAPE(
   A):NEXT:CLOSE #1:END
320 HOTSPOT: GOSUB READMOUSE:I
   F LBUTTON=TOGGLE THEN 32
   0 ELSE TOGGLE=LBUTTON
330 IF LBUTTON=0 THEN 320
340 GOSUB LOCATE:IF XP<1 OR XP>1
   6 OR YP<1 OR YP>16 THEN 340 EL
   SE RETURN
350 LOCATE: XP=INT((X-50)/16)+1:YP
   =INT((Y-40)/8)+1:RETURN
```

Program 2: Pointer Loader

```
10 DIM SHAPE(30):CLEARW 2:GOTOX
   Y 0,0:INPUT "Filename";FN$:OPEN "I
   ",#1,FN$
20 INPUT #1,HX,HY:FOR A=0 TO 15:IN
   PUT #1,SHAPE(A):NEXT
30 POKE CONTRL,111:POKE CONTRL
   +2,0
40 POKE CONTRL+6,37
50 POKE INTIN,HX-1:POKE INTIN+2,H
   Y-1:POKE INTIN+4,1
60 POKE INTIN+6,0:POKE INTIN+8,1
70 FOR A=10 TO 40 STEP 2:T=SHAPE(
   (A-10)/2)
80 POKE INTIN+A,T:POKE INTIN+A
   +32,T
90 NEXT:VDISYS (0):CLEARW 2:GOTOX
   Y 0,0:PRINT "Defined"
```
©

Using PALETTE USING On The PCjr

John And Jeff Klein

The IBM PCjr's PALETTE USING command lets you quickly change all screen attributes and colors in any graphics mode for a variety of effects. This article explains the details of PALETTE USING and demonstrates its usefulness with programming examples. An IBM PCjr with Cartridge BASIC is required.

Though Cartridge BASIC for the PCjr is very similar to BASICA for the IBM PC, the PCjr has extra graphics capabilities which the PC does not enjoy. One of these involves the PALETTE and PALETTE USING commands, which control color attributes. Before you can use PALETTE USING, you need to know how the simpler PALETTE statement works.

The PCjr offers 16 different colors, numbered from 0–15. An *attribute* is a number associated with a particular color. To explain what an attribute actually does, let's look briefly at the PCjr's color management scheme.

When you turn on the PCjr, there's a simple, one-to-one relationship between colors and attributes. Attribute number 1 corresponds to color number 1 (blue), attribute 2 corresponds to color 2 (green), and so on. Table 1 shows this initial arrangement.

Table 1: Default PCjr Attributes

Color Number	Attribute	Visual Color
0	0	black
1	1	blue
2	2	green
3	3	cyan
4	4	red
5	5	magenta
6	6	brown
7	7	white
8	8	gray
9	9	light blue
10	10	light green
11	11	light cyan
12	12	light red
13	13	light magenta
14	14	yellow
15	15	high-intensity white

The PALETTE command lets you change this arrangement by assigning a different color to any single attribute. Here's the general form of the PALETTE statement:

PALETTE *attribute, color*

Attribute tells the computer which attribute you're working with, and *color* indicates which

www.commodore.ca

Figure 1: PAL Integer Array

(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)	(15)	(16)
0	4	0	4	0	4	0	4	0	4	0	4	0	4	0	4

color you want to assign to that attribute. To take a simple example, say that you turn on the computer and draw some shapes in blue (color 1). This statement changes every blue object on the screen to red:

```
PALETTE 1,4
```

In this example, the 1 refers to attribute 1, and the 4 refers to color 4 (red). Before the statement executes, attribute 1 is assigned to color 1 (blue), the normal arrangement. After it executes, attribute 1 is reassigned so that it equals red. In an eyeblink, every blue object turns red. Likewise, the statement PALETTE 0,7 changes every shape that's initially black (color 0) to white (color 7). By performing a series of 16 PALETTE statements (FOR J=0 TO 15: PALETTE J,4: NEXT), you could change everything on the screen to red, regardless of its initial color.

In other words, PALETTE makes it possible to "rewire" the normal color scheme whenever you like. An object that starts out green can be changed to magenta; shapes that first appear in gray can be switched immediately to yellow, and so forth. If PALETTE were not available, such effects would be much more difficult and time-consuming; whenever you wanted a massive color change, you'd have to redraw every object of a given color in its new color.

After performing a number of PALETTE statements, you may want to restore the original color/attribute scheme. This can be done in two different ways, either by changing the screen mode, or by executing a PALETTE statement with no parameters (the keyword PALETTE followed by nothing).

USING A Shortcut

One disadvantage of PALETTE is that it affects only one attribute at a time. The more attributes you want to change, the more PALETTE statements you must perform.

Table 2: Attributes After PALETTE USING PAL(1)

Attribute	Array element linked with attribute	New color
0	PAL(1)	0
1	PAL(2)	4
2	PAL(3)	0
3	PAL(4)	4
–	–	–
–	–	–
15	PAL(16)	4

That's where PALETTE USING enters the picture. This special form of PALETTE can change all 16 attributes at once, assigning them to a set of 16 color values which are stored in an integer array. Here's the general format of the command:

PALETTE USING *arrayname(starting position)*

Arrayname identifies the array you want to use, and *starting position* tells the computer which array element belongs to attribute 0. The array can have any legal IBM array name, but it must be in existence before you execute PALETTE USING.

For example, let's say that you create an integer array named PAL with the contents shown in Figure 1 (array element 1 contains a zero value, element 2 contains the value 4, and so on).

Now assume that you execute the following statement:

```
PALETTE USING PAL(1)
```

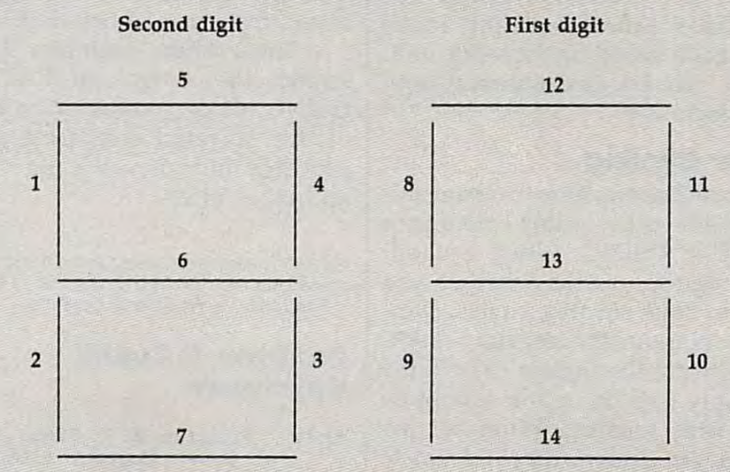
This single PALETTE USING command has the effect of reassigning all 16 attributes in one stroke. Table 2 shows the new color/attribute scheme which takes effect.

After a PALETTE USING, the values in the array replace the color numbers originally associated with each attribute. In this example, attribute 0 is still assigned to black (color 0), and attribute 4 is still assigned to red (4), but all the other odd attributes become black and all the even ones are changed to red. Since we specified a starting position of 1, the contents of array element 1 replace the color number for attribute 1; the other values follow in ascending numeric order.

In some cases, you may want to change some, but not all, of the 16 attributes with PALETTE USING. To retain the current color for a given attribute, make the corresponding array element equal to −1. For instance, if element PAL(2) equals −1 in the previous example, then attribute 1 retains its original color (blue), while the other 15 attributes are changed to black and red in even-odd order. Thus, the values you store in the integer array are limited to the range −1 to 15. The value −1 represents no change; values from 0–15 represent the colors shown in Table 1.

An interesting feature of PALETTE USING is its ability to start

Figure 2: Numeric Display Simulation



www.commodore.ca

anywhere in the array and begin changing attributes from that particular point (rather than always starting at the first array element). For instance, say that you have 100 different 16-color patterns stored in a single integer array (with a total of 1600 elements). To cycle through 100 different configurations, you need only execute a series of PALETTE USING statements, changing the starting position with each new command. The first statement could use a starting position of 1, the next a starting position of 17, and so forth.

Digital Countdown

Once you learn about PALETTE USING, many different effects come to mind. Program 1 demonstrates just one possibility, simulating the seven-segment numeric display seen on most electronic calculators and watches. Of course, calculators and watches create their displays with very different methods, but they still form the numerals 0–9 by turning various line segments on or off, just as we'll do here. Program 1 creates the line segments with the attributes shown in Figure 2 (note that attributes 0 and 15 are not used).

Line 20 creates the integer variable PAL, and line 50 READs the DATA values from lines 1000–1090 into the array. When that's done, the array contains all the patterns we'll need to form the digits 0–9. To display a new digit, we simply choose a different starting point for the next PALETTE USING command. As written, the program simply counts off seconds, but with a little additional programming, you can use it as a general routine to count any two-digit values. Lines 110–120 do the actual calculation based on the value of X, which can be any number from 0–99 inclusive.

Color Cycling

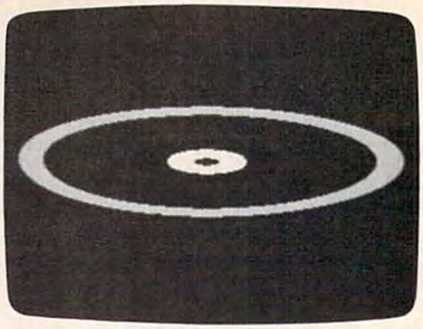
Program 2 shows how to create animated effects by cycling colors with PALETTE USING. When you run the program, it asks whether you want to draw squares, circles, ellipses, or random shapes, and whether you want the figures to be filled or empty (the fill option cannot be used with squares). When all the shapes are drawn, they'll seem to begin moving in a complex, tunnel-like effect. Though the figures appear to move in and out, the effect is entirely illusory. In fact, we're simply cycling through a pattern that includes the background color (black in this case). When a shape is colored the same as the background, it seems to disappear.

This type of animation is fairly simple to create, provided that you begin with a clear mental picture of the final result. For complicated pictures, you may want to sketch the various shapes on paper beforehand, or use "The Screen Machine," a graphics-design program published in the April 1984 issue of *COMPUTE!'s PC and PCjr* magazine. Since one color must be reserved to match the background, you're limited to 15 different colors for visible shapes. Changing the color associated with the background attribute also changes the border color, so it's often a good idea not to change attribute 0.

Program 3 employs similar techniques to create a colorful animated sign. In this case, only eight attributes are affected, leaving eight others free for additional effects.

These short examples barely scratch the surface of PALETTE USING. Once you learn the basics of this powerful command, you'll probably find yourself using it more and more often.

The IBM PCjr's PALETTE USING statement simplifies color control for graphic displays.



"Tunnel Vision" for the IBM PCjr creates a convincing 3-D effect with PALETTE USING commands.

For instructions on entering these listings, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

Program 1: Digital Countdown

```
PJ 1Ø ' Program #1 - Simulates t
      wo seven segment displays
IE 2Ø CLEAR,,,32768!:KEY OFF:SCR
      EEN 5:CLS:DEFINT P:DIM PAL
      (245):SOUND ON
PJ 3Ø C=Ø:X1=6Ø:GOSUB 2ØØØ
AB 4Ø C=1:X1=15Ø:GOSUB 2ØØØ
EO 5Ø FOR Z=Ø TO 245:READ PAL(Z)
      :NEXT
BB 6Ø X=Ø:PALETTE USING PAL(23Ø)
      :TIMER ON:ON TIMER(1) GOSU
      B 1ØØ
JO 7Ø GOTO 7Ø
PA 1ØØ X=X+1:IF X=6Ø THEN X=Ø
LC 11Ø Y=X-(INT(X/1Ø)*1Ø):PALETT
       E USING PAL(Y*23):SOUND 3
       ØØØ,.5,15,Ø
EM 12Ø Y=INT(X/1Ø):PALETTE USING
       PAL(Y*23+7):RETURN
JN 1ØØØ DATA -1,-1,-1,-1,-1,-1,-
        1,-1,6,6,6,6,6,Ø,6,-1,-1
        ,-1,-1,-1,-1,-1,-1
ME 1Ø1Ø DATA -1,-1,-1,-1,-1,-1,-
        1,-1,Ø,Ø,6,6,Ø,Ø,Ø,-1,-1
        ,-1,-1,-1,-1,-1,-1
OF 1Ø2Ø DATA -1,-1,-1,-1,-1,-1,-
        1,-1,Ø,6,Ø,6,6,6,6,-1,-1
        ,-1,-1,-1,-1,-1,-1
PE 1Ø3Ø DATA -1,-1,-1,-1,-1,-1,-
        1,-1,Ø,Ø,6,6,6,6,6,-1,-1
        ,-1,-1,-1,-1,-1,-1
LN 1Ø4Ø DATA -1,-1,-1,-1,-1,-1,-
        1,-1,6,Ø,6,6,Ø,6,Ø,-1,-1
        ,-1,-1,-1,-1,-1,-1
NG 1Ø5Ø DATA -1,-1,-1,-1,-1,-1,-
        1,-1,6,Ø,6,Ø,6,6,6,-1,-1
        ,-1,-1,-1,-1,-1,-1
LH 1Ø6Ø DATA -1,-1,-1,-1,-1,-1,-
        1,-1,6,6,6,Ø,6,6,6,-1,-1
        ,-1,-1,-1,-1,-1,-1
OI 1Ø7Ø DATA -1,-1,-1,-1,-1,-1,-
        1,-1,Ø,Ø,6,6,6,Ø,Ø,-1,-1
        ,-1,-1,-1,-1,-1,-1
LD 1Ø8Ø DATA -1,-1,-1,-1,-1,-1,-
        1,-1,6,6,6,6,6,6,6,-1,-1
        ,-1,-1,-1,-1,-1,-1
OI 1Ø9Ø DATA -1,-1,-1,-1,-1,-1,-
        1,-1,6,Ø,6,6,6,6,6,-1,-1
        ,-1,-1,-1,-1,-1,-1
BD 11ØØ DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
        ,Ø,Ø,Ø,Ø,Ø,Ø
PO 2ØØØ LINE (X1,3Ø)-(X1+2Ø,1ØØ)
        ,C*7+1,BF
FG 2Ø1Ø LINE (X1,1Ø1)-(X1+2Ø,17Ø
        ),C*7+2,BF
AG 2Ø2Ø LINE (X1+6Ø,1Ø1)-(X1+8Ø,
        17Ø),C*7+3,BF
EE 2Ø3Ø LINE (X1+6Ø,3Ø)-(X1+8Ø,1
        ØØ),C*7+4,BF
LE 2Ø4Ø Y=Ø:X=Ø:FOR Z=1 TO 16
BK 2Ø5Ø LINE (X1+X,3Ø+Y)-(X1+79-
        X,3Ø+Y),C*7+5
PB 2Ø6Ø LINE (X1+X,17Ø-Y)-(X1+79
        -X,17Ø-Y),C*7+7
```


www.commodore.ca

```
II 2Ø7Ø Y=Y+1:X=X+1.3:NEXT
BM 2Ø8Ø Y=Ø:X=Ø:FOR Z=1 TO 9
LM 2Ø9Ø LINE (X1+X,1ØØ+Y)-(X1+79-X,1ØØ-Y),C*7+6,BF
JB 21ØØ Y=Y+1:X=X+2.4:NEXT:RETURN
```

Program 2: Tunnel Vision

```
GN 1Ø ' Program #2 - Tunnel Animation
IB 2Ø CLEAR,,,32768!:KEY OFF:SCREEN 5:CLS:RANDOMIZE TIMER
CB 3Ø DEFINT P:DIM PAL(479):LOCATE 1Ø,11:PRINT "ONE MOMENT PLEASE"
KE 4Ø SPEED=3Ø
OI 5Ø FOR Z=Ø TO 479:READ PAL(Z):NEXT Z
IB 6Ø CLS:PALETTE:LOCATE 3,14:PRINT "MAIN MENU":LOCATE 8,14:PRINT "1) SQUARES":LOCATE 1Ø,14:PRINT "2) CIRCLE":LOCATE 12,14:PRINT "3) ELLIPSE":LOCATE 14,14:PRINT "4) RANDOM SHAPES":LOCATE 17,14:PRINT "CHOOSE ONE"
OG 7Ø A$=INPUT$(1):IF A$>"4" OR A$<"1" THEN 7Ø
OP 8Ø LOCATE 2Ø,14:PRINT "FILLED IN (Y/N) ?":AN$=INPUT$(1):IF AN$<>"N" AND AN$<>"n" AND AN$<>"Y" AND AN$<>"y" THEN 8Ø ELSE IF AN$="y" THEN AN$="Y" ELSE IF AN$="n" THEN AS$="N"
DC 9Ø CLS:ON VAL(A$) GOSUB 1ØØ,13Ø,16Ø,19Ø:GOSUB 27Ø:GOTO 6Ø
FJ 1ØØ ' *** SQUARES ***
CF 11Ø FOR Z=15Ø TO 1Ø STEP -1Ø:IF AN$="N" THEN LINE (16Ø-Z,1ØØ-Z/1.5)-(16Ø+Z,1ØØ+Z/1.5),Z/1Ø,B ELSE LINE (16Ø-Z,1ØØ-Z/1.5)-(16Ø+Z,1ØØ+Z/1.5),Z/1Ø,BF
NI 12Ø NEXT Z:LINE (157,98)-(163,1Ø2),Ø,BF:RETURN
AC 13Ø ' *** CIRCLES ***
FP 14Ø FOR Z=15Ø TO 1Ø STEP -1Ø:CIRCLE (16Ø,1ØØ),Z,Z/1Ø:IF AN$="Y" THEN PAINT STEP(Ø,Ø),Z/1Ø,Z/1Ø
GL 15Ø NEXT Z:CIRCLE (16Ø,1ØØ),2Ø:PAINT STEP(Ø,Ø),Ø,Ø:RETURN
MC 16Ø ' *** ELLIPSES ***
NG 17Ø FOR Z=15Ø TO 1Ø STEP -1Ø:CIRCLE (16Ø,1ØØ),Z,Z/1Ø,,,3/12:IF AN$="Y" THEN PAINT STEP(Ø,Ø),Z/1Ø,Z/1Ø
EJ 18Ø NEXT Z:RETURN
EI 19Ø ' *** RANDOM SHAPES ***
FK 2ØØ FOR Z=1 TO 15
EB 21Ø ON INT(RND*4+1) GOSUB 23Ø,24Ø,25Ø,26Ø
DO 22Ø NEXT Z:RETURN
QN 23Ø LINE (RND*32Ø,RND*2ØØ)-(RND*32Ø,RND*2ØØ),Z:RETURN
IL 24Ø CIRCLE (RND*32Ø,RND*2ØØ),RND*5Ø,Z:IF AN$="Y" THEN PAINT STEP(Ø,Ø),Z,Z:RETURN
PF 25Ø CIRCLE (RND*32Ø,RND*2ØØ),RND*5Ø,Z,,,RND*2:IF AN$="Y" THEN PAINT STEP(Ø,Ø),Z,Z:RETURN
NE 26Ø IF AN$="Y" THEN LINE (RND*32Ø,RND*2ØØ)-(RND*32Ø,RND*2ØØ),Z,BF:RETURN ELSE LINE (RND*32Ø,RND*2ØØ)-(RND*32Ø,RND*2ØØ),Z,B:RETURN
PF 27Ø ' *** MAIN ***
HH 28Ø FOR Z=1 TO 3Ø:FOR SLOW=1 TO SPEED:NEXT
JM 29Ø PALETTE USING PAL((Z-1)*16)
HA 3ØØ A$=INKEY$:IF A$<>"" THEN RETURN ELSE NEXT Z:GOTO 28Ø
MC 1ØØØ DATA -1,-1,-1, Ø,-1, 5,-1, Ø,-1,-1,-1,-1,-1,-1, 4,-1
QO 1Ø1Ø DATA -1,-1,-1,-1, 1, 2,14,-1,-1,-1,-1,-1,-1, 4,-1, Ø
FP 1Ø2Ø DATA -1,-1,-1, 1,-1, Ø,-1,14,-1,-1,-1,-1, 4,-1, Ø,-1
HJ 1Ø3Ø DATA -1,-1, 1,-1, Ø,-1, Ø,-1,14,-1,-1, 4,-1, Ø,-1,-1
PA 1Ø4Ø DATA -1, 1,-1, Ø,-1,-1,-1, Ø,-1,14, 4,-1, Ø,-1,-1,-1
PP 1Ø5Ø DATA -1,-1, Ø,-1,-1,-1,-1,-1, Ø, 6, 6, Ø,-1,-1,-1,-1
FL 1Ø6Ø DATA -1,-1, 1,-1,-1,-1,-1,-1, 4, 4,14,14,-1,-1,-1,-1
KI 1Ø7Ø DATA -1, Ø,-1, 1,-1,-1,-1, 4,-1, Ø, Ø,-1,14,-1,-1,-1
IN 1Ø8Ø DATA -1,-1, Ø,-1, 1,-1, 4,-1, Ø,-1,-1, Ø,-1,14,-1,-1
NP 1Ø9Ø DATA -1,-1,-1, Ø,-1, 5,-1, Ø,-1,-1,-1,-1, Ø,-1,14,-1
AF 11ØØ DATA -1,-1,-1,-1, 4,-1, 1,-1,-1,-1,-1,-1,-1, Ø,-1,14
ON 111Ø DATA -1,-1,-1, 4,-1, Ø,-1, 1,-1,-1,-1,-1,-1,-1, Ø,-1
MH 112Ø DATA -1,-1, 4,-1, Ø,-1, Ø,-1, 1,-1,-1,-1,-1,-1,14,-1
KL 113Ø DATA -1, 4,-1, Ø,-1,-1,-1, Ø,-1, 1,-1,-1,-1,14,-1, Ø
BF 114Ø DATA -1,-1, Ø,-1,-1,-1,-1,-1, Ø,-1, 1,-1,14,-1, Ø,-1
CK 115Ø DATA -1,-1, 4,-1,-1,-1,-1,-1,-1, Ø,-1, 2,-1, Ø,-1,-1
OK 116Ø DATA -1, Ø,-1, 4,-1,-1,-1,-1,-1,-1,14,-1, 1,-1,-1,-1
PA 117Ø DATA -1,-1, Ø,-1, 4,-1,-1,-1,-1,14,-1, Ø,-1, 1,-1,-1
AG 118Ø DATA -1,-1,-1, Ø,-1, 4,-1,-1,14,-1, Ø,-1, Ø,-1, 1,-1
IF 119Ø DATA -1,-1,-1,-1, Ø,-1, 4,14,-1, Ø,-1,-1,-1, Ø,-1, 1
II 12ØØ DATA -1,-1,-1,-1,-1, Ø, 6, 6, Ø,-1,-1,-1,-1,-1, Ø,-1
MB 121Ø DATA -1,-1,-1,-1,-1,14,14, 4, 4, Ø,-1,-1,-1,-1, 1,-1
JE 122Ø DATA -1,-1,-1,-1,14,-1, Ø, Ø,-1, 4, Ø,-1,-1, 1,-1, Ø
PE 123Ø DATA -1,-1,-1,14,-1, Ø,-1,-1, Ø,-1, 4,-1, 1,-1, Ø,-1
IO 124Ø DATA -1,-1,14,-1, Ø,-1,-1,-1,-1, Ø,-1, 5,-1, Ø,-1,-1
BI 125Ø DATA -1,14,-1, Ø,-1,-1,-1,-1,-1,-1, 1,-1, 4,-1,-1,-1
HA 126Ø DATA -1,-1, Ø,-1,-1,-1,-1,-1,-1, 1,-1, Ø,-1, 4,-1,-1
IM 127Ø DATA -1,-1,14,-1,-1,-1,-1,-1, 1,-1, Ø,-1, Ø,-1, 4,-1
OI 128Ø DATA -1, Ø,-1,14,-1,-1,-1, 1,-1, Ø,-1,-1,-1, Ø,-1, 4
CK 129Ø DATA -1,-1, Ø,-1,14,-1, 1,-1, Ø,-1,-1,-1,-1,-1, Ø,-1
```

Program 3: Animated Sign

```
IA 1Ø ' Program #3 - Animated Sign
BA 2Ø CLEAR,,,32768!:KEY OFF:SCREEN 5:CLS:DEFINT P:DIM P(95)
BH 3Ø C=1:FOR X=Ø TO 31Ø STEP 8.75:LINE(X,Ø)-(X+5,5),C,BF:LINE(X,195)-(X+5,2ØØ),4-C,BF:C=C+1:IF C=4 THEN C=1
OB 4Ø NEXT
JA 5Ø C=3:FOR Y=Ø TO 19Ø STEP 8.85ØØØ1: LINE(Ø,Y)-(5,Y+5),4-C,BF:LINE(3Ø6,Y)-(311,Y+5),C,BF:C=C+1:IF C=4 THEN C=1
OD 6Ø NEXT
HE 7Ø S$="BR6 BD1 H1 L4 G1 D1 F1 R4 F1 D1 G1 L4 H1 BU5 BR6;"
EN 8Ø I$="ND6;"
FC 9Ø G$="BD1 BR6 H1 L4 G1 D4 F1 R4 E1 U2 NL3 BU3;"
OG 1ØØ N$="ND6 F6 U6;"
OO 11Ø FOR Z=1 TO 5:FOR A=1 TO Z:Y=Z*3Ø+A-15:FOR B=1 TO Z:X=Z*2Ø+B+5Ø:S=Z*4:C=Z+3
PO 12Ø DRAW "BM=X;,=Y; C=C; S=S; XS$;BR2;XI$;BR2;XG$;BR2;XN$;"
KP 13Ø NEXT B,A,Z
CG 14Ø LOCATE 3,25:COLOR 13,Ø:PRINT "ALL THIS":LOCATE 5,22:PRINT "and still have":LOCATE 7,22:A$="8 colors open"
PD 15Ø FOR Z=1 TO 7:COLOR Z+8,Ø:PRINT MID$(A$,(Z-1)*2+1,2);:NEXT
IO 16Ø LOCATE 19,5:A$="TEXT":FOR Z=1 TO 4:COLOR 9-Z,Ø:PRINT MID$(A$,Z,1);:NEXT
MG 17Ø FOR Z=Ø TO 95:READ P(Z):NEXT Z
HO 18Ø FOR Z=Ø TO 5:PALETTE USING P(Z*16):NEXT:GOTO 18Ø
DG 1ØØØ DATA -1,-1, 4, Ø, 6, 2, 3, 4, 5,-1,-1,-1,-1,-1,-1,-1
HC 1Ø1Ø DATA -1, 4, Ø,-1, 5, 6, 2, 3, 4,-1,-1,-1,-1,-1,-1,-1
GD 1Ø2Ø DATA -1, Ø,-1, 4, 4, 5, 6, 2, 3,-1,-1,-1,-1,-1,-1,-1
EP 1Ø3Ø DATA -1,-1, 4, Ø, 3, 4, 5, 6, 2,-1,-1,-1,-1,-1,-1,-1
LI 1Ø4Ø DATA -1, 4, Ø,-1, 2, 3, 4, 5, 6,-1,-1,-1,-1,-1,-1,-1
LC 1Ø5Ø DATA -1, Ø,-1, 4,15,15,15,15,15,-1,-1,-1,-1,-1,-1,-1
```

©

www.commodore.ca

Applesoft List Enhancer

Steven Roth

Tired of program listings that whiz across the screen at unreadable speeds? This short utility for Apple II computers lets you scroll through a listing at your own pace, one screenful at a time. For all Apple II-series computers with DOS 3.3 and ProDOS.

LIST is one of the most frequently used commands in Applesoft BASIC, yet it has some inconvenient features. The program listing scrolls upward too fast to read, unless you repeatedly press CTRL-S. If you pass the program lines you want to see, you have to LIST the program again.

"Applesoft List Enhancer" is a machine language program that solves both of these problems. It divides the program listing into pages, and each page consists of one screenful of the listing. Instead of scrolling through the listing in only one direction, you now page through it, either forward or backward.

Type in and save the program below, then run it. This BASIC program automatically creates a machine language file on disk named ALE. (Don't use the name ALE for the BASIC program or you'll get a FILE TYPE MISMATCH error.) Once that's done, you don't need the BASIC program again except to create new copies of ALE.

Enter BRUN ALE to install and activate the List Enhancer. Now load any Applesoft BASIC program. To list it, type an ampersand (&) followed by an optional starting line number. If you don't enter any line number, the program is listed from the beginning.

The right and left arrow keys let you page forward or backward through the listing. Press the right arrow key to display the next screenful of program lines; the left arrow key displays the previous page. To exit listing mode so you can edit a line, press either the ESC key or CTRL-C.

Applesoft LIST Enhancer

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
1Ø 100 FOR I = 24576 TO 24847: R
       EAD A: POKE I,A:CK = CK +
        A: NEXT
1D 11Ø IF CK < > 33557 THEN PRIN
       T "ERROR IN DATA STATEMEN
       TS.": STOP
4Ø 12Ø PRINT CHR$ (4)"BSAVE ALE,
       A$6ØØØ,L$11Ø"
2D 13Ø DATA 169,32,141,245,3,169
       ,16,141
52 14Ø DATA 246,3,169,96,141,247
       ,3,96
D9 15Ø DATA 32,12,218,32,26,214,
       165,8Ø
71 16Ø DATA 141,32,97,165,81,141
       ,33,97
24 17Ø DATA 32,88,252,16Ø,Ø,132,
       6,2ØØ
66 18Ø DATA 132,3Ø,132,249,177,1
       55,24Ø,75
8A 19Ø DATA 2ØØ,177,155,17Ø,2ØØ,
       177,155,132
DØ 2ØØ DATA 9,133,8,134,7,32,36,
       237
1C 21Ø DATA 164,9,169,32,32,92,2
       19,165
EØ 22Ø DATA 36,2Ø1,33,144,15,23Ø
       ,3Ø,165
82 23Ø DATA 3Ø,2Ø1,23,24Ø,48,32,
       142,253
61 24Ø DATA 169,5,133,36,2ØØ,177
       ,155,2Ø8
5E 25Ø DATA 81,168,177,155,17Ø,2
       ØØ,177,155
5Ø 26Ø DATA 134,155,133,156,32,1
       42,253,23Ø
A1 27Ø DATA 3Ø,165,3Ø,2Ø1,23,24Ø
       ,14,16Ø
ØB 28Ø DATA 1,2Ø8,177,23Ø,6,23Ø,
       6,169
6Ø 29Ø DATA Ø,133,249,24Ø,17,23Ø
       ,6,23Ø
6B 3ØØ DATA 6,164,6,165,7,153,32
       ,97
45 31Ø DATA 2ØØ,165,8,153,32,97,
       32,12
DB 32Ø DATA 253,2Ø1,155,24Ø,18,2
       Ø1,131,24Ø
AØ 33Ø DATA 14,2Ø1,149,24Ø,1Ø1,2
       Ø1,136,24Ø
17 34Ø DATA 58,32,58,255,76,15Ø,
       96,76
A4 35Ø DATA 2Ø8,3,16,144,56,233,
       127,17Ø
CA 36Ø DATA 132,9,16Ø,2Ø8,132,15
       7,16Ø,2Ø7
7B 37Ø DATA 132,158,16Ø,255,2Ø2,
       24Ø,7,32
37 38Ø DATA 44,215,16,251,48,246
       ,169,32
B2 39Ø DATA 32,92,219,32,44,215,
       48,5
17 4ØØ DATA 32,92,219,2Ø8,246,32
       ,92,219
25 41Ø DATA 76,64,96,165,6,2Ø1,2
       ,24Ø
5F 42Ø DATA 192,56,233,4,133,6,1
       66,6
C7 43Ø DATA 189,32,97,133,8Ø,232
       ,189,32
D9 44Ø DATA 97,133,81,32,26,214,
       32,88
97 45Ø DATA 252,16Ø,1,132,3Ø,132
       ,249,76
CA 46Ø DATA 44,96,165,249,2Ø8,22
       4,24Ø,153
```

©

www.commodore.ca

Loading And Linking Commodore Programs

Part 3

Jim Butterfield, Associate Editor

This month's installment shows how one program can automatically load and link to any other program. The methods described here work on any eight-bit Commodore computer with either disk or tape.

There are three major ways of connecting Commodore programs together. *Chaining* allows several programs to perform a job, each program continuing the work that a previous program has done. *Load-linking* enables one program to activate another, with the new program starting fresh on a new task. *Overlaying* allows a main program to call in additional subroutines, data tables, or display information. This article discusses load-linking.

When a first program has finished a job, it may decide to load-link to a second program to do a succeeding task. The new program is not a continuation of the previous program—it's a brand new job and won't share any information from the earlier program unless it happens to use a common data file. Load-linking is harder to achieve on Commodore computers than chaining (see Part 2 of this article, April COMPUTE!). You must do a little extra work to load-link.

Why Load-Linking?

It's common, especially on non-Commodore machines, to have a disk start with a menu program which lets a user pick another program to run. The menu program has to load-link the chosen program, since the new program is not a continuation of the menu. Newcomers to Commodore computers often try to chain from the menu program to the other program. However, chaining requires the first program to be the larger of the two, and menu programs are almost always very short.

At other times, when a group of programs work with one or several data files, it's common for these programs to run independently of each other. Each may use the files in different ways, with data stored in various kinds of arrays for each job. Independent programs call for load-linking.

Sometimes, a series of modules are arranged to work together as a unit. There may be a main program, some machine language programs in varying locations, perhaps a high-resolution graphics screen, and possibly a special character set. The Commodore B128 (called the 700 series in Europe, and not to be confused with the Commodore 128) often requires a transfer sequence program in order to run machine language programs. All the various parts need to be arranged in the proper areas of memory. A special program called a *bootstrap* or *boot* brings everything in. When the bootstrap program has completed its work, often you want to erase it from memory after it sets the main program into operation. Load-linking can accomplish this job, too.

Two Methods

There are two major ways to load-link programs. One technique is used at the end of the first program, setting up the load of the second. It's a variation of the dynamic keyboard technique. The alternative method is used at the beginning of the second program, cleaning up some of the things that have happened during the loading process.

The dynamic keyboard method is probably the simplest. Essentially, it simulates a user typing on the keyboard, giving LOAD and RUN commands. When you type LOAD as a direct command, the new program comes in fresh. It scraps the old program and retains

www.commodore.ca

no variables or other data. That's what we want with a load-link, of course. RUN then sets the new program off to a fresh start.

The pointer cleanup method corrects conditions that result from program chaining. The start-of-variables pointer needs to be corrected so it is set directly behind the last byte loaded; then a CLR command erases any leftover variables and fixes up the other pointers. We'll try each method in turn.

Dynamic Keyboard

The dynamic keyboard technique is often described as "making the computer type on its own keyboard." To do this, a program stuffs characters into the computer's keyboard buffer and resets the buffer counter. When the computer discovers these characters in the buffer, it thinks they were actually typed on the keyboard and executes the commands they spell out.

Since the keyboard buffer normally holds only about nine or ten characters, we can't store a complete load sequence there. But we can use Commodore's screen-editing capability. We print a command on the computer's screen, move the cursor back onto the command line, then store one character (a carriage return, character 13) in the buffer. To perform LOAD followed by RUN, we print two command lines and store two RETURNs in the buffer. The effect is the same as if you typed in the LOAD and RUN commands by hand. The new program starts as a fresh, independent job—you'll have performed a successful load-link.

The keyboard buffer and its counter occupy various memory locations in different Commodore computers. You'll need to use the appropriate addresses for your machine. The following table will help:

	Key Counter	Start of Buffer
VIC-20, Commodore 64	198	631
Commodore 128 in 128 mode	208	842
Commodore 16, Plus/4	239	1319
PET/CBM with 4.0 BASIC	158	623
Original ROM PETs	525	527
B-128 (Model 700)	209	939

To execute two command lines with this method, we'll POKE a value of 2 into the counter and a value of 13 (RETURN) into the first two locations of the buffer. The following example uses Commodore 64 addresses.

Dynamic Keyboard Example

Let's write two simple programs and let the menu program select which one to run. Type in this simple square root program:

```
100 PRINT "TABLE OF SQUARE ROO
   TS"
110 FOR J=1 TO 20
120 PRINT J, SQR(J)
130 NEXT J
```

You can try running the program if you want. It's not very exciting, but it does work. Now save it with the filename SQUARE (don't substitute any other filename). Type NEW and enter this simple cube root program:

```
100 PRINT "TABLE OF CUBE ROOTS
   "
110 X=1/3
120 FOR I=1 TO 20
130 PRINT I, I↑X
140 NEXT I
```

Again, you might try running the program. Save it with name CUBE when you're satisfied that it works. Type NEW again. Now we'll write a simple loading program that uses the dynamic keyboard technique to perform load-linking. This program runs as listed on the VIC-20 and Commodore 64. For another type of machine, use the table above to change the POKE addresses in lines 280, 290, and 300.

```
100 DATA SQUARE,CUBE
110 READ A$(1),A$(2)
120 PRINT "WHICH ROOTS DO YOU
   {SPACE}WANT--"
130 FOR J=1 TO 2
140 PRINT A$(J)
150 NEXT J
160 INPUT "WHICH (1 OR 2)";N
170 IF N<1 OR N>2 GOTO 120
180 PRINT CHR$(147)
190 PRINT
200 PRINT
210 PRINT "LOAD";CHR$(34);A$(N
   );CHR$(34);",8"
220 PRINT
230 PRINT
240 PRINT
250 PRINT
260 PRINT "RUN"
270 PRINT CHR$(19)
280 POKE 198,2
290 POKE 631,13
300 POKE 632,13
310 END
```

In this case, the menu program is longer than the programs it loads. But that doesn't matter. This method works the same regardless of the length of the programs involved. You will see the LOAD and RUN commands appear on the screen.

Pointer Cleanup Method

The previous method puts extra commands in the first program to load the second. Now we'll look at another method that adds commands to the start of the second program. At this point, the new program has been chained—the old variables and their pointers still exist. To start out fresh, we must erase the old variables and make sure that new variables go into the proper memory area. What we're doing is changing a chain into a load-link.

Our task is to set the start-of-variables pointer to the correct address and then perform CLR to erase the old variables. On the B128 and Commodore 128 in 128 mode, there's a slight difference. Here, we don't need to set the start-of-variables, since variables are held in a different memory bank. Instead, we set a different pointer (called end-of-BASIC) which doesn't exist on the other machines.

Now that we know which pointer to change, we must decide what value to put there. The value will be the address of the last byte loaded, plus one. Immediately after a load, we can find this address still in a pointer. Here's a table to show the various locations:

	Pointer to be changed	Pointer where value is found
VIC-20, Commodore 64	45/46	174/175
Commodore 128 in 128 mode	4624/4625	174/175
Commodore 16, Plus/4	45/46	157/158
PET/CBM with 4.0 BASIC	42/43	201/202
Original ROM PETs	124/125	229/230
B-128 (Model 700)	47/48	150/151

www.commodore.ca

After the pointer is fixed, be sure to perform CLR to reset all the other variable pointers. Again, these steps must be taken at the very beginning of the new program.

Pointer Cleanup Example

Here's an example similar to the previous one which demonstrates the second method. This is the menu program:

```
100 DATA SQUARE1,CUBE1
110 READ A$(1),A$(2)
120 PRINT "WHICH ROOTS DO YOU
    {SPACE}WANT--"
130 FOR J=1 TO 2
140 PRINT A$(J)
150 NEXT J
160 INPUT "WHICH (1 OR 2)";N
170 IF N<1 OR N>2 GOTO 120
180 LOAD A$(N),8
```

Notice that this menu program is much shorter than the first example. We'll do the extra work when we write the programs to be loaded. Save the menu program, then enter NEW and type these lines:

```
70 POKE 45,PEEK(174)
80 POKE 46,PEEK(175)
90 CLR
100 PRINT "TABLE OF SQUARE ROO
    TS"
110 FOR J=1 TO 20
120 PRINT J, SQR(J)
130 NEXT J
```

This is similar but not identical to the first square root program. The difference is the three extra lines at the beginning. Don't try to run this program yet; instead, save it with the filename SQUARE1. Enter NEW a second time and enter this simple cube root program:

```
70 POKE 45,PEEK(174)
80 POKE 46,PEEK(175)
90 CLR
100 PRINT "TABLE OF CUBE ROOTS
    "
110 X=1/3
120 FOR I=1 TO 20
130 PRINT I, I↑X
140 NEXT I
```

Save this program with the filename CUBE1. If you have a computer other than the Commodore 64 or VIC-20, remember to change the POKE and PEEK values in lines 70 and 80 of both these programs according to the table above.

Now load the menu program and run it. You've seen two different ways to perform load-linking. A program can get another program off to a clean start by using either of these techniques.

Better Branching In Applesoft

Mark Russinovich

Are you ready to update the Applesoft BASIC on your Apple II-series computer? This handy utility adds extra flexibility by letting you branch to line numbers computed by variables or even complex expressions. For both DOS 3.3 and ProDOS.

Though it's been used to write a tremendous number of programs, Applesoft BASIC has some significant shortcomings compared to more recent versions of BASIC. One of these is the inability to use a variable or BASIC expression as the object of a GOTO, GOSUB, or RESTORE command. Applesoft BASIC requires you to use a line number as the destination of a GOTO, GOSUB, or RESTORE.

There are two disadvantages to this. First, line numbers contain no clue to the purpose of the branch: GOSUB DELAY makes the purpose of a subroutine more obvious to everyone than GOSUB 1000. Second, branching statements that are limited to constants can't be modified while a program is executing. Unlike line numbers, variables can change as a program runs, letting you modify the destination of a command just by changing the value of the variable. For example, you could use GOSUB CHOICE*1000 to branch to subroutines at lines 1000, 2000, or 3000 depending on whether the variable CHOICE equals 1, 2, or 3.

Improved GOTO And GOSUB

"Enhancer" lets you substitute variables and even complex expressions as the object of GOTO, GOSUB, and RESTORE in Applesoft BASIC. To use it, first enter Program 1 and be sure to save a copy. Program 1 is a BASIC program that creates the machine language routine for Enhancer on disk, using the filename APPLE.ENHANCER. (Be careful to use some name *other than* APPLE.ENHANCER for Program 1 itself; otherwise, you'll get a FILE TYPE MISMATCH error when you run it.)

After you've created the APPLE.ENHANCER file, you won't need Program 1 again, except to make additional copies of the machine language. To load and activate the utility, add this line to the beginning of any BASIC program:

```
10 PRINT CHR$(4)"BRUN APPLE.
   ENHANCER"
```

Make sure the Enhancer machine language file is on the disk in the current drive. As soon as your

www.commodore.ca

program executes this line, it can use the features of APPLE.ENHANCER. You can still use the normal form of GOTO and GOSUB (GOTO 100, GOSUB 20005, or whatever), but in addition, you can also use this format:

& GOTO ***expression***
& GOSUB ***expression***

Note the ampersand (&) symbol that precedes the GOTO or GOSUB. In place of *expression*, you can substitute any legal Applesoft variable name or expression. Here's a short example to show how the new commands work:

```
AB 1Ø BEGIN = 2Ø
7B 2Ø INPUT "CHOOSE 1 OR 2 ";CHO
      ICE
B5 3Ø IF CHOICE < 1 OR CHOICE >
      2 THEN & GOTO BEGIN
6D 4Ø & GOSUB CHOICE * 1ØØØ
FA 5Ø PRINT "DONE."
F1 6Ø END
D1 1ØØØ PRINT "FIRST ROUTINE"
D5 1Ø1Ø RETURN
2B 2ØØØ PRINT "SECOND ROUTINE"
D6 2Ø1Ø RETURN
```

Notice how the improved GOSUB serves as a substitute for ON-GOSUB. Depending on what value CHOICE has, the program branches to line 1000 or 2000. Likewise, the improved GOTO command can replace an ON-GOTO command.

RESTORE To A Destination

The RESTORE statement in Applesoft normally stands by itself; it simply resets the READ pointer to the beginning of the program. If you wish to READ a particular piece of data midway through the list, you first READ past every preceding DATA item in the program. With APPLE.ENHANCER in place, you can use this more convenient command:

& RESTORE ***expression***

Again, you must include the ampersand before the command; replace *expression* with any legal Applesoft expression or variable. Now you can access any DATA line in the program instantly.

For instance, say that a program has many DATA statements containing information for three different graphics screens. Ordinarily, you'd have to READ through all the DATA for screens 1 and 2 before reaching the DATA for screen 3—a process that takes time and increases the possibility of errors. The improved RESTORE command allows you to zero in on the precise DATA line you want, without time-consuming delays.

Program 2 demonstrates all the features of these new commands. Without these features, the program would be considerably longer and less flexible.

For instructions on entering these listings, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

Program 1: APPLE.ENHANCER Filemaker

```
ED 1Ø FOR I = 768 TO 887: READ A
      : POKE I,A:CK = CK + A: NE
      XT
7A 2Ø IF CK < > 14482 THEN PRINT
       "ERROR IN DATA STATEMENTS
      .": STOP
E1 3Ø PRINT CHR$ (4)"BSAVE APPLE
      .ENHANCER,A$3ØØ,L$77"
EE 4Ø DATA 169,76,141,245,3,169,
      16,141
AB 5Ø DATA 246,3,169,3,141,247,3
      ,96
E9 6Ø DATA 16Ø,Ø,177,184,217,115
      ,3,24Ø
88 7Ø DATA 11,2ØØ,192,3,24Ø,3,76
      ,2Ø
C2 8Ø DATA 3,32,2Ø1,222,14Ø,118,
      3,23Ø
BA 9Ø DATA 184,2Ø8,2,23Ø,185,32,
      1Ø3,221
95 1ØØ DATA 172,118,3,192,1,24Ø,
       1Ø,192
75 11Ø DATA 2,24Ø,35,32,82,231,7
       6,65
A4 12Ø DATA 217,169,3,32,214,211
       ,165,185
AA 13Ø DATA 72,165,184,72,165,11
       8,72,165
BF 14Ø DATA 117,72,169,176,72,32
       ,82,231
63 15Ø DATA 32,65,217,76,21Ø,215
       ,32,82
96 16Ø DATA 231,32,26,214,56,165
       ,155,233
43 17Ø DATA 1,164,156,176,1,136,
       133,125
A5 18Ø DATA 132,126,96,171,176,1
       74,Ø,Ø
```

Program 2: Better Branching Demo

```
ED 1Ø PRINT CHR$ (4)"BRUN APPLE.
      ENHANCER"
53 2Ø TEXT : HOME
EF 3Ø & RESTORE 1ØØ
CD 4Ø FOR I = 1 TO 3: READ A: &
      GOSUB 5Ø + A * 1Ø: NEXT I
F8 5Ø END
FC 6Ø PRINT "THIS IS STATEMENT 2
      ": RETURN
7E 7Ø PRINT "THIS IS STATEMENT 3
      ": RETURN
7E 8Ø PRINT "THIS IS STATEMENT 1
      ": RETURN
F8 9Ø DATA THIS IS UNWANTED DATA
55 1ØØ DATA 3,1,2
```
©

Copies of articles from this publication are now available from the UMI Article Clearinghouse.

For more information about the Clearinghouse, please fill out and mail back the coupon below.

UMI Article Clearinghouse

Yes! I would like to know more about UMI Article Clearinghouse. I am interested in electronic ordering through the following system(s):

☐ DIALOG/Dialorder ☐ ITT Dialcom
☐ OnTyme ☐ OCLC ILL Subsystem
☐ Other (please specify)____________
☐ I am interested in sending my order by mail.
☐ Please send me your current catalog and user instructions for the system(s) I checked above.

Name____________
Title____________
Institution/Company____________
____
Department____________
Address____________
City________State______Zip______
Phone (_____)____________

Mail to: University Microfilms International
300 North Zeeb Road, Box 91 Ann Arbor, MI 48106

www.commodore.ca

Random Numbers
In Machine Language For Commodore 64

Neil Boyle

Here are two different methods for obtaining random numbers from machine language on the Commodore 64. Even if you're not a machine language programmer, you may find the explanation of the RND function useful in BASIC programming.

Sooner or later, nearly every programmer needs to generate random numbers. They're useful for introducing an element of uncertainty in games, as well as many other applications. Random numbers are easily available in BASIC with the RND function, but it's not so easy to generate these numbers in machine language. Two different methods are available on the Commodore 64: The first involves the BASIC RND function, and the second uses the SID (Sound Interface Device) chip.

BASIC's RND Function

The BASIC routine which performs RND is at memory location $E097 (decimal 57495) in the Commodore 64's Read Only Memory (ROM). This function generates values of different types, based on a *seed* number. Random numbers generated by the RND function are not truly random; rather, they are part of a very long, repeating number series (hence, the term *pseudo-random* numbers). However, they are usable as random numbers in most programs.

There are three different ways to use RND in BASIC. If you supply a positive number inside the parentheses—for instance, RND(1)—the first seed value used is the one copied from ROM to locations $8B–$8F when you turn the computer on. Each subsequent seed value is generated by scrambling the previous one. The RND function looks at the sign of the number in parentheses to see whether it's positive, but does not use the number itself. Thus, since RND(1234) and RND(1) both use positive values, both expressions produce exactly the same result.

What this means in practice is that RND with a positive number produces exactly the same number sequence each time you turn on the computer. To demonstrate, enter SYS 64738 to reset the computer, then enter the following line in direct mode (without a line number):

```
FOR J=1 TO 5:PRINT RND(1):NEXT
```

The computer prints these numbers:

```
.185564016
.0468986348
.827743801
.554749226
.897233831
```

Now reset the computer and enter the same line, substituting a different positive number in the parentheses. As you'll see, positive values generate the same results whether you use RND(1) or RND (65536). Since the reset stores the same values in the seed locations, and RND's scrambling method is predictable, the same numbers show up every time.

If you supply a zero inside the parentheses, RND gets values from the computer's internal timer A and the time-of-day clock on CIA chip #1 to use as the seed. Since the clock values are constantly changing, this would seem to be a way to generate more truly random numbers. However, because the time-of-day clock operates with binary coded decimal numbers, certain values never appear in the seed. Thus, the randomness of RND(0) is questionable. To illustrate this, enter and run the following one-line program:

```
10 POKE 1024 + (RND(0) * 1000), 160:
  GOTO 10
```

The many unfilled character spaces on the screen represent values that are never generated by RND(0).

When you supply a negative number in the parentheses, RND uses the value itself as the seed. For instance, enter the following line in direct mode:

```
X=RND(-654321):PRINT RND(1)
```

The result is always .333675369. If you substitute a different negative number, RND generates a different yet predictable result. This can be useful when testing a program: To generate the same series of numbers each time, set the seed with a negative number, then use RND with a positive number. In cases where you want a wholly unpredictable number series, the best method is to start with

www.commodore.ca

X=RND(−TI) to begin with an unknown seed value, then use positive RND arguments thereafter. Since the seed value depends on how many sixtieths of a second have elapsed since you turned on the computer (or reset the software clock with TI$), the results are unpredictable enough to satisfy most ordinary needs.

Calling RND From ML

To call the RND function from within an ML program, execute JSR $E09A. The value held in the accumulator register when you call the routine determines what RND does. If the accumulator holds a negative byte value ($80–$FF), then the value in floating point accumulator 1 (FAC1) is used as the seed. If the accumulator holds zero, RND uses values from timer A and the time-of-day clock, just as in BASIC. If the accumulator value is positive ($00–$7F), then the seed value in locations $8B–$8F is used.

Thus, your choices for ML programming are essentially the same as for BASIC, except that the system variable TI is not available as an argument. An alternative to using TI is to load the byte values from the software clock ($A0–A2) directly into the seed addresses ($8B–8F), thus giving a fairly random seed. If you'd rather not bother with loading the accumulator, you can perform JSR $E0BE to go directly to the routine which uses the stored seed value.

As you probably know from using RND in BASIC, the function always returns a floating point number between 0 and 1. In machine language, it is usually more convenient to use a single byte value in the range 0–255. Unfortunately, some of the randomness of the BASIC floating point number comes from scrambling the bytes in FAC1; this is lost when single bytes are used. One alternative is to convert the floating point number to an integer and use one or more bytes of the integer. But this is somewhat awkward. Nonrepeating random numbers involving all possible single byte values seem to be produced in locations $63 and $64 of FAC1 after a call to $E09A with the accumulator set appropriately. The values found in locations $62 and $65 do not include all the values from 0–255 and therefore should not be used.

SID's Random Number Generator

The 64's SID chip also has the ability to generate random values and is very easy to use. All you need to do is select the noise waveform for the SID's voice 3 oscillator and set voice 3's frequency to some nonzero value (the higher the frequency, the faster the random numbers are generated). It is not necessary to gate (turn on) the voice. Once this is done, random values appear in location $D41B. The parameters need only be set once, as shown in this example. (You'll need a machine language assembler to enter this and the following example; the semicolons and the comments which follow them are optional.)

```
LDA #$FF    ; maximum frequency value
STA $D40E   ; voice 3 frequency low byte
STA $D40F   ; voice 3 frequency high byte
LDA #$80    ; noise waveform, gate bit off
STA $D412   ; voice 3 control register
RTS
```

Once this code is executed, the SID chip continuously produces random byte values, which you can retrieve with a statement like LDA $D41B. This method works only on the Commodore 64 and 128, since only those two computers have a SID chip. Values obtained from the SID chip do seem to be random: Each of the values in the range 0–255 occurs at about the same frequency, and the series does not repeat in the first 34,000 values.

In many cases, you'll want to obtain random numbers only within a certain range. In BASIC, this is done with a statement like RND (1)*(U−L)+L. Here the variables U and L stand for the upper and lower limits, respectively, of the number range we want. This statement produces integer numbers within that range, excluding the first and last values in the range (for instance, if U=30 and L=1, then 30 and 1 do not appear in the series). To include the first and last values, use a statement like INT (RND(1)*(U−L+1)+L.

In machine language, we're usually interested in single byte values: integers in the range 0–255. Should larger values be needed, you can always generate two or more bytes and combine them. Getting numbers within a certain range is simply a matter of generating numbers until you find one that falls in the range you want. For instance, the following routine generates numbers within the range $10–$40 (before performing this routine, you must set up the SID chip as shown above):

```
RAND LDA $D41B ; get random value
               ;   from 0–255
     CMP #$31  ; compare to
               ;   U−L+1
               ; U−L+1 = $31 =
               ;   $40−$10+$01
     BCS RAND  ; branch if value >
               ;   U−L+1
     ADC #$10  ; add L
     RTS
```

This routine generates random numbers until one is found that falls between 0 and the difference between the high and low values. Then the low value is added to the result to give a value between the low and high limits, inclusive. If the range is very small, many numbers may have to be generated before you find one that's suitable.

You can decrease the delay by ANDing the random number with a value that removes the unwanted higher bits. For instance, if the difference between the low and high limits is $0A, then AND the random value with $0F to remove the four high bits, then test whether it falls within the range. One useful special case deserves mention. To generate an integer in the range −1 to 1 inclusive, use the BASIC statement INT(RND(1)*3)−1. To get a value between $01 and $FF in machine language, AND the random value with $03 and use a routine like the one shown above, where L=$FF and U−L+1=$03.

The method you use to generate random numbers depends on your needs. In machine language, if repeatable values aren't needed, it's simplest to use the SID chip. Should repeatable values be required (for testing a routine, etc.), set the chosen seed in the seed locations $8B–$8F, call the ROM routine at $E0BE, then use the value found in locations $63 or $64 for the random number. Due to its questionable randomness, the timer/clock method is not recommended.©

www.commodore.ca

Amiga Puzzle

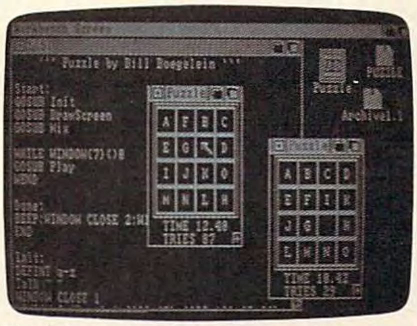
Bill Boegelein

The following programs provide you with an entertaining puzzle game that pops up as a small window on the Workbench screen. But more importantly, they demonstrate some interesting and powerful programming techniques in Amiga BASIC.

A popular game that used to keep kids occupied for hours in the back of the station wagon was known as the "slide puzzle." This was simply a plastic frame with 15 numbered or lettered tiles arranged in a 4 × 4 pattern with one square left vacant. The object was to slide one tile at a time into the vacant slot in an attempt to restore the puzzle to its proper numeric or alphabetic order. "Amiga Puzzle" is the Amiga BASIC equivalent of the slide puzzle.

Amiga Puzzle works on any Amiga with Amiga Microsoft BASIC (not ABasiC—Amiga BASIC is available as a free upgrade if your computer was shipped with ABasiC). To get started, run Amiga BASIC, enter Program 1 below, and save a copy on disk. Be sure to set Preferences for a 60-column screen before running Amiga Puzzle. When you run the program, it pops open as a small window on the Workbench screen and scrambles the tiles. When the program announces "Ready" using the Amiga's built-in speech capability, you can begin.

To move a tile into the vacant slot, point to the tile with the mouse and press the left mouse button. If you try to cheat by moving a tile diagonally, the program will scold you. The gadget in the lower-right corner of the window, normally reserved for window resizing, has been replaced with a plus sign in Amiga Puzzle. Clicking on this gadget rescrambles the puzzle and starts a new game. The other gadgets (front/back, the move bar, and close window) are all active as usual.



This photo shows two copies of "Amiga Puzzle" running simultaneously, an example of the Amiga's multitasking capabilities.

As a final touch, the Puzzle window displays the elapsed time and number of moves since the start of the game.

How It Works

Amiga Puzzle was adapted from a similar puzzle game available on the Macintosh. Those of you familiar with Microsoft BASIC 2.0 on the Macintosh will immediately see many similarities in Amiga BASIC. In fact, many Macintosh BASIC programs can be converted to Amiga BASIC with little effort. Both languages support windows, pull-down menus, labels, many types of graphics commands, and other features.

Amiga Puzzle is divided into seven subroutines named Done, Init, DrawScreen, Mix, Play, CheckCheat, and More. There are also two small subprograms, Talk and Position. Here are descriptions of what these subroutines do:

Init: initializes variables, user-defined functions, and loads the puzzle's characters A–O into the two-dimensional array *c()*.

DrawScreen: displays the puzzle's squares and characters.

Mix: mixes only adjacent squares and keeps track of the blank square's position in variables *blankX* and *blankY.*

Play: the main subroutine where the current mouse position is compared to coordinates of each square stored in the three-dimensional array *rat.*

CheckCheat: makes sure the attempted move is a legal one (adjacent squares only, no diagonal moves).

More: checks if the mouse is clicked on the plus-sign gadget in Amiga Puzzle's lower-right corner. If so, the program starts a new game by jumping back to Start.

Done: ends the program and returns control to BASIC.

Special Features

Most of the program is standard Microsoft BASIC. The only lines that merit special attention are the user-defined functions in the Init subroutine. One function operates the puzzle's timer, and the other determines if the puzzle has been solved.

A quick way to time any event is to define a function that subtracts the current time from the initial starting time. This is done with the function FNlaps. The current time is obtained from TIMER and the

www.commodore.ca

initial time stored in the variable *starttime*.

The other defined function, FNwin, determines if the puzzle has been solved by comparing the letter in each tile—stored in the array *c()*—with the characters A–O. Each letter in the correct position returns a value of −1 (true). So the puzzle has been solved when all 15 letters are sorted, returning a value of −15.

A very powerful feature of Amiga BASIC is its subprogram capability (see "Requester Windows in Amiga BASIC," COMPUTE!, March 1986). In effect, subprograms let you add new commands to the language. The word TALK is not a command in Amiga BASIC, but has been added to Amiga Puzzle as a subprogram. It lets you execute both a SAY command and the text-to-speech TRANSLATE$ function by simply typing TALK followed by the desired string. Similarly, POSITION is a subprogram that expects parameters of the X and Y coordinates as the location to PRINT the desired string.

Program 2, "MakeIcon," lets you create a special icon for Amiga Puzzle. This is optional, of course, since Amiga BASIC automatically assigns a standard icon when you save the program. MakeIcon recreates the special icon I designed with the Amiga's Icon Editor (a new feature that's included with the free upgrade to version 1.1 of the operating system). The icon data is converted from the hexadecimal values in the DATA statements into single-byte values and stored in the file *Puzzle.info* (all Amiga icon files end with *.info*). In addition to the icon's shape and color, the special data also contains information that makes Amiga BASIC the program's *default tool*. That means when the icon is clicked on the Workbench screen, Amiga BASIC is automatically loaded and run prior to loading and running the game.

Multitasking With BASIC

Since multitasking is built into the Amiga's operating system as a standard feature, no special programming techniques are required to write a BASIC program that's capable of running simultaneously with other tasks. Feel free to move Amiga Puzzle over or under other windows running different programs without causing interference. If your computer has at least 512K RAM, you can even click on the Amiga Puzzle icon a second time and run two of the games at once.

Program 1: Amiga Puzzle

```
Start:←
GOSUB Init←
GOSUB DrawScreen←
GOSUB Mix←
WHILE WINDOW(7)<>0←
GOSUB Play←
WEND←
Done:←
BEEP:WINDOW CLOSE 2:WINDOW 1←
END←
Init:←
DEFINT a-z←
Talk " " ←
WINDOW 1,"Puzzle",(230,45)-(230+13
8,45+96),30←
tries=0:RANDOMIZE TIMER←
FOR y=0 TO 3←
FOR x=0 TO 3←
c(x,y)=x+y*4+ASC("A") ' Load charac
ters←
NEXT x←
NEXT y←
blankX=x-1:blankY=y-1←
c(blankX,blankY)=ASC("  ")←
DEF FNlaps!=((TIMER-starttime!)\60)
+(((TIMER-starttime!) MOD 60)/100)←
DEF FNa=(c(0,0)=65)+(c(1,0)=66)+(
c(2,0)=67)+(c(3,0)=68)←
DEF FNb=(c(0,1)=69)+(c(1,1)=70)+(
c(2,1)=71)+(c(3,1)=72)←
DEF FNc=(c(0,2)=73)+(c(1,2)=74)+(c
(2,2)=75)+(c(3,2)=76)←
DEF FNd=(c(0,3)=77)+(c(1,3)=78)+(
c(2,3)=79)+(c(3,3)=80)←
DEF FNwin=(FNa+FNb+FNc+FNd
) 'Won if = -15←
RETURN←
DrawScreen:←
FOR y=0 TO 3←
FOR x=0 TO 3←
Position (x+1)*3,(y+1)*2,CHR$(c(x,y
)) ' Print characters←
x1=x*30+10:y1=y*18+3 ' Draw boxes
←
LINE (x1,y1)-(x1+30,y1+18),1,b←
LINE (x1-1,y1-1)-(x1+30+1,y1+18+1)
,1,b←
LINE (10-5,3-3)-(4*30+10+5,4*18+3+
3),1,b←
moreX=128:moreY=89←
LINE (moreX,moreY)-(moreX+10,more
Y+10),1,bf ←
Position 14,11,"+" ' New game gadget←
rat(x,y,0)=x1:rat(x,y,1)=y1←
NEXT x←
NEXT y←
Position 3,10,"TIME 0.00" ←
Position 3,11,"TRIES 0"←
RETURN←
Mix:←
x=blankX:y=blankY←
FOR mixing=333 TO 1 STEP -1 ←
IF (mixing AND 1)=0 THEN←
x=INT(RND*4):y=blankY ' Even←
ELSE ←
y=INT(RND*4):x=blankX ' Odd←
END IF←
GOSUB CheckCheat←
NEXT mixing←
Talk "Ready."←
starttime!=TIMER←
RETURN←
Play:←
LOCATE 10,8:PRINT USING "##.##";FNl
aps!;←
WHILE MOUSE(0)<>0←
mouseX=MOUSE(3):mouseY=MOUSE(
4)←
FOR y=0 TO 3←
FOR x=0 TO 3←
IF (mouseX>rat(x,y,0) AND mouseX<r
at(x,y,0)+30) AND (mouseY>rat(x,y,1
) AND mouseY<rat(x,y,1)+18) THE
N GOSUB CheckCheat:RETURN←
NEXT x←
NEXT y←
GOSUB More←
WEND←
RETURN←
CheckCheat:←
IF (ABS(x-blankX)>1 OR ABS(y-blankY)
>1) OR ((x<>blankX AND y<>blankY)
) THEN←
IF mixing=0 THEN Talk "Cheater." 'Ch
eating←
ELSE 'Not cheating←
SWAP c(x,y),c(blankX,blankY)←
Position (x+1)*3,(y+1)*2,CHR$(c(x,y))
←
SWAP x,blankX:SWAP y,blankY←
Position (x+1)*3,(y+1)*2,CHR$(c(x,y))
←
END IF←
IF mixing=0 THEN←
tries=tries+1←
LOCATE 11,8:PRINT tries;←
WHILE MOUSE(0)<>0:WEND←
IF FNwin=-15 THEN Talk "We have a wi
nner.":GOTO More←
END IF←
RETURN←
More:←
WHILE MOUSE(0)<>0 OR FNwin=-15 'A
nother game?←
mouseX=MOUSE(3):mouseY=MOUSE(
4)←
IF MOUSE(0)=0 AND (mouseX>more
X AND mouseX<moreX+10) AND (mou
seY>moreY AND mouseY<moreY+10
) THEN GOTO Start←
IF WINDOW(7)=0 THEN Done←
WEND←
RETURN←
SUB Talk(a$) STATIC←
SAY TRANSLATE$(a$)←
END SUB←
SUB Position(x,y,a$) STATIC←
LOCATE y,x:PRINT a$;←
END SUB←
```

Program 2: MakeIcon

```
WIDTH 60←
PRINT:PRINT SPC(20);"Creating Puzzl
e Icon":PRINT ←
DEFINT a-z←
CLOSE←
OPEN "Puzzle.info" FOR OUTPUT AS 2←
FOR ii=1 TO 98←
READ long$←
```


www.commodore.ca

```
FOR i=1 TO 8 STEP 2←
byte$=MID$(long$,i,2)←
PRINT byte$;←
HX byte$←
tmp=VAL(byte$)←
checksum=checksum+tmp←
PRINT #2,CHR$(tmp);←
NEXT i←
NEXT ii←
CLOSE←
KILL"Puzzle.info.info"'Delete icon's ic
on←
IF checksum<>12063 THEN BEEP:PRI
NT:PRINT "Error in data statements."←
END←
SUB HX(a$) STATIC 'STR$ hex to dec (m
ax=FFFF)←
a$=UCASE$(a$):n=0:ans=0 ←
FOR i=1 TO LEN(a$)←
c=ASC(MID$(a$,i,1))←
n=VAL(CHR$(c))←
IF c>ASC("9") THEN n=(VAL(CHR$(
c-17)))+10←
ans=ans*16+n←
NEXT i←
a$=STR$(ans)←
END SUB←
' Puzzle Icon data←
DATA E3100001,00000000,00800062,0
0280017 ←
DATA 00040003,00010001,B8C00000,0
0000000 ←
DATA 00000000,00000000,00000000,0
0000000 ←
DATA 044C0001,B8F00000,00000000,0
07C0000←
DATA 00560000,00000000,00000000,1
0000000←
DATA 00000028,00170002,00021600,0
3000000←
DATA 00000000,00000000,3FFFFFFF,F
C002000←
DATA 00000400,27FFFFFF,E4002404,0
8102400←
DATA 24040810,24002404,08102400,2
7FFFFFF←
DATA E4002418,7E0F2400,243C3319,A
400243C←
DATA 33302400,27663E30,E400247E,3
3302400←
DATA 24C33319,A40024C3,7E0F2400,2
7FFFFFF←
DATA E4002404,08102400,24040810,2
4002404←
DATA 08102400,27FFFFFF,E4002000,0
0000400←
DATA 3FFFFFFF,FC000000,00000000,0
0000000←
DATA 00000000,00000000,00000000,0
0000000←
DATA 00000000,00000000,00000000,0
0000000←
DATA 00000000,00000000,00000000,0
0000000←
DATA 00000000,00000000,00000000,0
0000000←
DATA 00000000,00000000,00000000,0
0000000←
DATA 00000000,00000000,00000000,0
0000000←
DATA 00000000,00000000,00000000,0
0000000←
DATA 00000000,00000000,00000000,0
0000000←
DATA 00000000,00000000,000C3A41,6
D696761←
DATA 42415349,43000000←
```

©

DEBUT
Atari BASIC Debugging Tool

Gary W. Swanberg

Here's a powerful debugging utility that lets Atari BASIC programmers analyze and manipulate variables, search for BASIC keywords, produce printed documentation of important program information, and call up disk directories without leaving BASIC. It works on all Atari 400/800, XL, and XE computers with at least 40K RAM, a disk drive, and Atari DOS 2.0 or 2.5.

"DEBUT," a utility for debugging Atari BASIC programs, was inspired by the "Atari Wedge" (COMPUTE!, December 1982). It uses the Wedge technique to provide BASIC with direct-mode commands for listing and locating variables, dumping variable values, and replacing one variable with another. It also searches for BASIC keywords and includes a DIR command for listing the disk directory. Output is easily redirected from the screen to the printer, disk drive, or any other output device. And because DEBUT is written in machine language, it remains completely transparent—BASIC programs can be loaded, modified, tested, and saved without interference.

Preparing DEBUT

The program following this article creates DEBUT as an AUTORUN.SYS file that runs with two of the three revisions of Atari BASIC (revisions B and C). If PRINT PEEK(43234) returns 162, then you have revision A, in which case you must substitute the following lines for the corresponding lines in Program 1:

```
200 DATA 255,255,0,31,157,31,175,164,
    41,168
202 DATA 76,98,164,76,144,164,76,92,181,
    76
204 DATA 12,181,76,53,181,76,159,186,76,
    137
206 DATA 171,76,188,175,76,155,171,68,69,
    66
```

(Revision A is the original BASIC cartridge made for the 400/800 and 1200XL computers. Revision B is the BASIC built into the 600XL and 800XL. Revision C is built into the 65XE and 130XE and also is available as a cartridge from Atari.)

Type in and save the BASIC program, then insert the disk on which you want to create DEBUT's AUTORUN.SYS file and type RUN. Make sure an AUTORUN.SYS file—such as Atari *SpeedScript* or *SpeedCalc*—doesn't already exist on the disk or it will be

www.commodore.ca

replaced. After the AUTORUN.SYS file is created, you won't need to run the BASIC program again, except to create copies of DEBUT on other disks. To start DEBUT, turn the computer off, make sure a disk with the DEBUT AUTORUN.SYS file is in the drive, then turn the computer on. DEBUT loads automatically and announces its presence with the message DEBUT. This message appears whenever you press SYSTEM RESET, reminding you that DEBUT is active.

DEBUT protects itself by resetting MEMLO (the pointer to the bottom of free memory, locations 743 and 744) and by trapping the DOS command. As in the Atari Wedge, typing KILL gets rid of the utility and enables DOS.

Note: DEBUT works with Atari DOS 2.0 and 2.5, but may not work with other types of DOS, especially after a SYSTEM RESET. Be sure that you have one of these versions before typing in the program.

Using DEBUT

Once DEBUT is up and running, you have four new commands at your disposal for debugging and documenting BASIC programs—plus the DIR command. You type the command and press RETURN, just as you would with any direct-mode BASIC statement. Type DIR, for example, to list the disk directory, or use the form DIR D:*filespec* to list specific files. The ? and * wildcard characters are allowed, and drive numbers (D2:, D3:, etc.) can be specified as well. (D: defaults to drive 1.)

Now you're ready to load one of your BASIC programs and try the four main commands:

FIND	(Find a variable or BASIC keyword.)
REP	(Replace one variable with another.)
XREF	(List variables and cross-reference line numbers.)
VIEW	(View the values of all variables.)

These commands act on your entire BASIC program unless you specify a range of line numbers using the form *starting line,last line.* You may specify either or both. When specifying only the last number, precede it with a comma, or DEBUT thinks it's a starting number. The line range parameter is optional, so it follows the other arguments to the command, as we'll explain below.

All commands can also use the *output switch*—a slash appended to the command followed by the name of the device to which you want to redirect output (P for Printer, D for Disk, C for Cassette, etc.). This sends the command's output to the device rather than to the screen. To send the disk directory listing to the printer, for example, type DIR/P. This option uses channel 1 for output, closing the channel when it's done. Files created on disk or cassette are ASCII files which can be viewed with most Atari word processors or ENTERed into memory with BASIC.

We'll see more examples as we take a detailed look at each command.

The FIND Command

With FIND, you can quickly identify any lines in your program that contain a certain variable or BASIC keyword. Here is the general format:

FIND *[/output device] target [starting line,last line]*

where */output device* is the optional switch for redirecting output from the screen; *target* is the variable name or BASIC keyword you want to find; and *starting line,last line* is the optional line number range for the search.

FIND lists every line in which the target appears. The target can be a statement name (keywords that begin a statement, such as PRINT, PLOT, and POKE), a function name (such as RND, COS, and PEEK), or a variable name. Include the right parenthesis as part of subscripted variable names, and the dollar sign as part of string variable names. (Remember: X, *X(*, and *X$* are three distinct variables.)

FIND does not locate keywords that are neither statements nor functions—THEN and TO, for instance. GOSUB and GOTO are missed when they appear as part of the ON-GOSUB or ON-GOTO statements. Also, FIND won't locate any operators, such as OR, AND, +, and /, to name a few.

FIND *does* locate ERROR lines, however; use FIND * for this purpose.

Directing output to the disk drive (FIND/D:*filename*) presents an interesting possibility—the file thus created can later be retrieved using BASIC's ENTER command, allowing you to sift lines out of one program to create another.

Perhaps you've discovered that keywords can sometimes be used as variable names in Atari BASIC. FIND assumes that if it's spelled like a keyword, it *is* a keyword. You can force FIND to search for variables only by preceding the variable name with the greater-than symbol (>). Thus, FIND LEN comes up with all occurrences of the LEN function, while FIND >LEN lists occurrences of a variable called LEN. This option is needed only if your target variable could be confused with a keyword. (Be careful when choosing variable names in Atari BASIC. LET NOTE=66 creates a legitimate variable, but if you try to access its value—for example, SOUND 0,NOTE,10,10—BASIC treats it as NOT E, while FIND, without the > option, would go off looking for the keyword NOTE.)

When you specify a line number range with the FIND command, only those lines within that range are listed. Here are some examples:

FIND X(

(List all lines in the program containing the *array variable* X.)

FIND TEMP 100

(List all lines containing the variable name TEMP, beginning at line 100.)

FIND/P CHR$,100

(List all lines through line 100 containing the function CHR$ and send the listing to the printer.)

FIND * 100,200

(List all lines from 100 through 200 that contain ERRORs.)

FIND/D:NEWFILE.ASC PRINT,20000

(Create a disk file named NEWFILE.ASC to hold the listing of all lines through line 20000 that contain the keyword PRINT.)

The XREF Command

As you write an Atari BASIC program, newly declared variables are added to an area of memory called the *variable name table.* XREF lists the contents of this table and cross-

www.commodore.ca

references the lines in which each variable appears. Here is the general format:

XREF *[/output device] [starting line,last line]*

where */output device* is the optional switch for redirecting output from the screen; and *starting line,last line* is the optional line range parameter.

Often, XREF finds variables in the variable name table that don't actually appear in your program. You can recognize these unused variables because XREF lists them with no line number references. Unused variables can happen when you delete all occurrences of a certain variable from a program, or when you misspell a variable or command and BASIC thinks you're creating a new variable. These unused variable names clutter up the variable name table, which might become a factor in a long program since the table is limited to 128 names.

XREF can help you eliminate this deadwood. If a large number of unused variables show up, you can clear them out by LISTing the program to disk, typing NEW, and then re-ENTERing the program. (LIST and ENTER reconstruct the variable name table, whereas SAVE and LOAD preserve it.)

When you specify a line number range with XREF (for instance, XREF 1000,2000), only those variables and line references within that range are listed. At the foot of the listing, both the count of listed variables and the total number of variables in the name table are displayed.

XREF/P creates a hardcopy of the variable listing—useful documentation. XREF also makes it easy to spot misspelled variable names. No longer must you trace through a recalcitrant program line by line only to find that your mysterious bug is due to the slip of a finger. And if you're as poor a typist as I am, you'll be making frequent use of the REP command described below.

Here are a few examples:

XREF

(List and cross-reference all variables in the program.)

XREF 2000,2999

(List and cross-reference all variables in the subroutine in lines 2000 through 2999.)

XREF ,10000

(List and cross-reference all variables up to line 10000.)

XREF/D:VARLIST

(List and cross-reference all variables in the program, and send the list to the disk file VARLIST.)

The REP Command

Now it's easy to replace those cryptic variable names with new names of crystal clarity—or, depending on your motives, vice versa. REP acts in a flash, replacing old variable names with new variable names. It's handy for making corrections and tightening up code—but be careful. With or without such a command, the more ambitious your program, the wiser it is to save backup copies and document changes as you make them.

Here is the general format:

REP *[/output device] oldvar newvar [starting line,last line]*

where */output device* is the optional switch for redirecting output from the screen; *oldvar* is the old variable name; *newvar* is the new variable name (separated by a space, not a comma, from *oldvar*); and *starting line,last line* is the optional line number range. (The only output produced by the output switch is a message telling you how many variables were changed.)

REP does not let you replace keywords or mismatched variable types (for instance, you can't change A$ to A). Also, REP *requires that the new variable already exists in the variable name table.* To add a new variable to the table, simply use it in any BASIC statement, even in direct mode. (If you want to create a new variable SUM, for instance, you can just type SUM=0 and press RETURN.)

If you don't include the optional line number range, REP works on the entire program. If you specify a range of lines, it replaces the old variable name only within those lines. Watch out for this, since it may cause a conflict elsewhere in your program.

Here are some examples:

REP CB CHECKBALANCE

(Replace all occurrences of the variable CB with the new variable name CHECKBALANCE.)

REP CLIENTNAME$ CN$

(Replace all occurrences of the variable CLIENTNAME$ with the new variable name CN$.)

REP I(INDEX(

(Replace all occurrences of the array variable I() with the array variable INDEX().)

REP SUM TOTAL 3000,3999

(Replace all occurrences of the variable SUM with the new variable name TOTAL within the subroutine in lines 3000 to 3999.)

Variables are stored as one-byte tokens in Atari BASIC. REP looks up the old and new variable names in the name table to determine their tokens, then replaces every old token (within range) with the new one. The old name remains in the name table (type XREF to see for yourself) and keeps the value it had at the time of the change.

The VIEW Command

BASIC's PRINT statement lets you examine variable values in direct mode as you test and debug a program, but it can be cumbersome when you're juggling lots of variables, especially subscripted variables. Of course, there's an easier way.

DEBUT's VIEW command lists all variables and their values, including strings and arrays, without requiring you to name each one. Type VIEW to quickly display the values, or use the optional output switch to generate hardcopy that you can analyze with the proverbial fine-toothed comb.

Here is the format of the command:

VIEW *[/output device] [starting line,last line]*

where */output device* is the optional switch for redirecting output from the screen; and *starting line,last line* is the optional range of line numbers.

Here is the format of the VIEW command's output:

VARIABLE=*value*
STRING$ *current length,maximum length*
"string contents"
ARRAY(*rows,cols*
10 20 30 40
2 4 6 8
3 5 7 11

Scalar variable values are displayed in a pretty straightforward

www.commodore.ca

manner—the variable name is followed by its value. Strings are listed with additional information—the current and maximum length. String contents are printed in quotes below the name.

For subscripted variables, VIEW lists the row and column dimensions corresponding to the indexes you assign with the DIM statement, plus one. (Because indexing starts at zero—ARRAY(0,0) is a valid array element—the actual size of each dimension is one greater than the value you give.) Values are listed by column and row; sample values are shown in the example above. A singly subscripted variable is really a one-column array; its values are listed vertically. If a string or array has not been dimensioned, only the name prints. Unused variables aren't listed at all.

The output switch and range option work with this command as they do with all others. For example:

VIEW/P

(Prints a hardcopy of all the variables and their values.)

VIEW/P 200,300

(Prints a hardcopy of the variables and values in lines 200 through 300.)

Additional Hints

The DEBUT commands are an aid to debugging, but they won't do the whole job for you. Use them along with other techniques (such as tracing the program's execution with temporary PRINT statements) to close the gap between what you're telling BASIC to do, and what you *want* it to do.

Since the optional disk output produced by these commands is compatible with any word processor that handles ASCII text, you can easily produce program documentation with DEBUT. For example, you could save the XREF listing to disk, and use a word processor to add explanations of each variable. Every minute spent documenting a program pays off threefold should you ever have to go back and modify it.

The DEBUT command formats are tolerant of extra spaces, and command words can be abbreviated—just type the first letter followed by a period. In fact, the period alone is all you need for DIR. The output switch is unaffected by abbreviations—DIR/P, D./P, and ./P all do the same thing. In addition, the FIND command accepts abbreviated keywords as its target. For example, FIND FOR, FIND F., and F.F. all seek out the keyword FOR. Sound confusing? Experiment. You'll find the shortcuts which are most useful to you.

Notes For ML Programmers

It would be nice if DEBUT had a renumbering command, program trace capabilities, or even a few more DOS commands. Machine language programmers can add such new commands by patching entries into DEBUT's command table, which reserves 25 bytes for just this purpose.

A command table entry consists of your machine language routine's address *minus one*, followed by the command name. The address is stored in high byte/low byte order. The command name is in ASCII code exactly as you'd type it, except that the rightmost byte must have the most significant bit on (inverse video). Add entries starting at location 8149 (hex $1FD5), but don't use more than the 25 bytes or you'll overwrite DEBUT. Your code can be appended to the DEBUT AUTORUN.SYS file by using the DOS copy with append feature.

DEBUT does some preprocessing before your routine gets control. As a result, you'll be able to abbreviate your commands, with DEBUT abbreviations taking precedence. The output switch works as well, opening channel 1 to the output device and storing the channel number at location 181 ($B5) before branching to your routine. If the command line requires further parsing, your routine will find it at location 1408 ($580); the offset to the first argument (if any) will be in location 242 ($F2). Your machine language routine should end with an RTS instruction if you wish it to return through DEBUT to BASIC.

DEBUT calls on several routines in Atari BASIC and accesses various BASIC tables whose addresses have changed in the later releases, which accounts for the slightly different version of the program for BASIC revision A. DEBUT commands will *not* trigger the notorious lockup bug of revision A and (especially) revision B. For those with an interest in the inner workings of Atari BASIC, I highly recommend *The Atari BASIC Source Book* (COMPUTE! Books), as well as the aforementioned "Atari Wedge" article by Charles Brannon (reprinted in *COMPUTE!'s Third Book of Atari* as "The Wedge: Adding Commands to Atari BASIC").

DEBUT: Atari BASIC Debugging Tool

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
CF 10 ? "INSERT DISK - PRESS
      RETURN WHEN READY"
HG 20 IF PEEK(764)<>12 THEN
      20
PC 30 OPEN #1,8,0,"D:AUTORUN
      .SYS"
CD 40 ? "CREATING DEBUT AUTO
      RUN.SYS FILE"
KB 50 FOR I=1 TO 1304:READ B
      YTE:PUT #1,BYTE:NEXT I
EN 60 ? "AUTORUN.SYS CREATED
      ":END
BM 70 DATA 255,255,0,31,157,
      31,159,164,36,168
CJ 80 DATA 76,84,164,76,130,
      164,76,142,181,76
FK 90 DATA 62,181,76,103,181
      ,76,153,186,76,129
GD 100 DATA 171,76,167,175,7
       6,147,171,68,69,66
BK 110 DATA 85,84,155,53,46,
       48,155,165,12,141
NK 120 DATA 86,31,165,13,141
       ,87,31,169,0,141
DL 130 DATA 241,31,169,85,13
       3,12,169,31,133,13
GO 140 DATA 160,31,169,31,32
       ,3,36,32,91,31
AM 150 DATA 169,46,141,231,2
       ,169,36,141,232,2
IB 160 DATA 96,32,88,31,76,5
       1,31,160,0,185
PL 170 DATA 26,3,201,69,240,
       8,200,200,200,192
EL 180 DATA 34,208,242,96,20
       0,169,158,153,26,3
OG 190 DATA 169,31,153,27,3,
       162,15,189,0,228
NF 200 DATA 157,158,31,202,1
       6,247,169,241,141,162
AG 210 DATA 31,169,31,141,16
       3,31,24,173,4,228
MG 220 DATA 105,1,141,243,31
       ,173,5,228,105,0
DI 230 DATA 141,244,31,96,17
       4,31,215,31,32,108
PB 240 DATA 68,73,210,32,178
       ,86,73,69,215,33
NB 250 DATA 95,88,82,69,198,
       33,212,70,73,78
MG 260 DATA 196,34,23,82,69,
       208,34,93,75,73
BG 270 DATA 76,204,32,76,68,
       79,211,0,0,0
NL 280 DATA 242,31,45,36,32,
       73,242,8,201,155
```


www.commodore.ca

```
BE 290  DATA 240,5,238,241,31
        ,40,96,152,72,138
HH 300  DATA 72,165,194,201,9
        3,208,92,174,241,31
EH 310  DATA 240,87,169,155,1
        57,128,5,162,0,142
DI 320  DATA 241,31,134,242,3
        2,64,35,169,31,160
NF 330  DATA 174,162,2,32,4,3
        1,176,61,134,242
LM 340  DATA 189,128,5,73,47,
        72,208,8,32,215
AN 350  DATA 35,48,22,32,47,3
        5,160,255,132,204
LN 360  DATA 140,254,2,200,13
        2,203,132,153,132,154
MN 370  DATA 132,245,32,91,32
        ,104,208,3,32,10
OL 380  DATA 36,169,0,141,254
        ,2,76,83,160,177
FF 390  DATA 149,72,200,177,1
        49,72,76,64,35,104
KK 400  DATA 170,104,168,169,
        155,40,96,72,162,112
NM 410  DATA 32,12,36,160,173
        ,169,32,32,38,36
CN 420  DATA 104,201,155,240,
        3,32,28,36,162,112
BE 430  DATA 169,6,32,227,35,
        162,112,169,5,160
EE 440  DATA 128,157,73,3,133
        ,150,132,149,32,38
CA 450  DATA 36,169,5,32,14,3
        6,48,8,32,16
DG 460  DATA 31,32,222,34,208
        ,225,162,112,76,12
LL 470  DATA 36,68,58,42,46,4
        2,155,32,243,34
OD 480  DATA 32,195,35,240,10
        6,32,38,35,32,92
LB 490  DATA 35,176,89,32,213
        ,34,32,16,31,32
NF 500  DATA 107,34,240,70,14
        4,76,32,28,31,32
GJ 510  DATA 182,221,32,217,3
        4,162,226,32,241,35
BB 520  DATA 169,44,32,19,31,
        162,228,32,241,35
AJ 530  DATA 38,210,144,60,32
        ,213,34,169,34,32
DK 540  DATA 219,34,162,226,3
        2,25,31,240,17,160
NC 550  DATA 0,177,224,32,19,
        31,32,222,34,240
ED 560  DATA 30,32,118,34,208
        ,232,169,34,32,219
IF 570  DATA 34,76,28,33,169,
        61,32,19,31,32
DE 580  DATA 252,35,32,222,34
        ,240,4,230,203,208
HF 590  DATA 145,76,213,34,32
        ,40,221,165,228,133
NB 600  DATA 234,165,229,133,
        235,32,213,34,162,232
PM 610  DATA 32,25,31,240,223
        ,162,234,32,25,31
MN 620  DATA 240,231,166,224,
        164,225,32,137,221,32
DH 630  DATA 252,35,32,217,34
        ,32,222,34,240,207
GJ 640  DATA 160,6,32,118,34,
        136,208,250,240,221
BA 650  DATA 208,2,133,245,32
        ,243,34,32,195,35
LM 660  DATA 240,59,32,38,35,
        32,92,35,176,35
EH 670  DATA 32,197,34,230,20
        5,198,205,208,3,32
EF 680  DATA 209,34,162,237,3
        2,241,35,32,217,34
HM 690  DATA 230,175,165,175,
        201,6,208,245,32,81
BE 700  DATA 35,144,228,176,7
        ,165,245,240,3,32
GM 710  DATA 197,34,32,222,34
        ,240,37,230,203,208
AK 720  DATA 192,32,213,34,32
        ,213,34,32,239,35
AM 730  DATA 169,33,160,201,3
        2,3,36,165,203,73
FA 740  DATA 128,168,169,0,32
        ,245,35,169,33,160
IJ 750  DATA 205,76,3,36,96,3
        2,79,70,160,32
PO 760  DATA 76,73,83,84,69,6
        8,155,240,64,230
HA 770  DATA 242,201,62,240,3
        0,170,169,55,224,42
BH 780  DATA 240,7,198,242,32
        ,131,34,176,5,133
KO 790  DATA 204,76,2,34,32,1
        41,34,176,6,105
NJ 800  DATA 61,133,203,16,7,
        32,173,34,176,23
AN 810  DATA 133,203,32,246,3
        4,32,92,35,176,13
GF 820  DATA 32,10,31,32,222,
        34,240,5,32,81
FC 830  DATA 35,144,243,96,24
        0,49,32,173,34,176
DN 840  DATA 44,133,203,32,10
        9,34,133,245,32,64
EN 850  DATA 35,32,173,34,176
        ,29,133,246,32,109
BL 860  DATA 34,197,245,208,2
        0,32,246,34,32,92
HN 870  DATA 35,176,12,165,24
        6,145,138,32,122,34
FA 880  DATA 32,170,35,144,24
        4,32,239,35,169,34
GJ 890  DATA 160,85,76,3,36,3
        2,67,72,65,78
CB 900  DATA 71,69,68,155,173
        ,86,31,133,12,173
EF 910  DATA 87,31,133,13,76,
        116,228,165,203,41
EC 920  DATA 127,32,22,31,165
        ,210,74,96,162,224
KF 930  DATA 208,2,162,153,24
        6,0,208,2,246,1
HF 940  DATA 96,173,1,31,172,
        0,31,162,2,208
AM 950  DATA 8,173,3,31,172,2
        ,31,162,0,32
NO 960  DATA 4,31,176,12,189,
        128,5,201,32,240
BI 970  DATA 6,201,155,240,2,
        56,96,134,242,165
EH 980  DATA 175,24,96,165,13
        1,164,130,162,0,32
LE 990  DATA 4,31,176,238,32,
        154,34,144,5,32
NL 1000  DATA 7,31,144,246,9,
         128,96,32,122,34
PM 1010  DATA 32,213,34,32,21
         3,34,32,16,31,169
EF 1020  DATA 5,133,205,169,1
         55,208,2,169,32,76
BI 1030  DATA 19,31,164,17,20
         8,2,198,17,152,96
DA 1040  DATA 32,0,216,176,7,
         32,210,217,164,212
AN 1050  DATA 165,213,96,56,1
         02,203,169,127,160,2
         55
GB 1060  DATA 32,34,35,200,13
         2,162,132,163,32,64
DD 1070  DATA 35,201,44,240,1
         6,32,230,34,176,24
JL 1080  DATA 132,162,133,163
         ,32,64,35,201,44,208
NG 1090  DATA 13,230,242,32,2
         30,34,176,6,48,4
GM 1100  DATA 133,165,132,164
         ,165,136,133,138,165
         ,137
NO 1110  DATA 133,139,96,166,
         242,202,232,189,128,
         5
LO 1120  DATA 201,32,240,4,20
         1,155,208,244,134,24
         2
GO 1130  DATA 166,242,189,128
         ,5,201,32,208,3,232
AK 1140  DATA 208,246,134,242
         ,201,155,96,24,165,2
         07
JC 1150  DATA 101,138,133,138
         ,144,2,230,139,160,0
EI 1160  DATA 177,138,133,237
         ,200,177,138,133,238
         ,56
CB 1170  DATA 165,164,229,237
         ,165,165,229,238,176
         ,2
BM 1180  DATA 56,96,200,177,1
         38,133,207,165,237,2
         29
HI 1190  DATA 162,165,238,229
         ,163,144,206,200,177
         ,138
DA 1200  DATA 133,156,200,177
         ,138,197,204,240,34,
         201
MO 1210  DATA 2,144,190,201,5
         5,240,186,200,177,13
         8
JA 1220  DATA 197,203,240,19,
         201,14,240,17,201,15
PN 1230  DATA 240,17,201,22,2
         40,167,200,196,156,2
         08
DJ 1240  DATA 233,240,211,24,
         96,169,6,208,3,200
DC 1250  DATA 177,138,24,200,
         132,170,101,170,168,
         208
DF 1260  DATA 213,165,203,41,
         127,133,175,165,131,
         164
PO 1270  DATA 130,162,0,32,13
         ,31,160,0,177,149
NI 1280  DATA 96,162,1,134,18
         1,32,10,36,32,26
MH 1290  DATA 36,169,8,157,74
         ,3,169,0,157,75
AE 1300  DATA 3,169,3,208,31,
         162,153,181,1,180
IE 1310  DATA 0,133,213,132,2
         12,32,170,217,32,230
DN 1320  DATA 216,165,244,164
         ,243,133,150,132,149
         ,76
NM 1330  DATA 16,31,162,16,16
         9,12,157,66,3,32
HE 1340  DATA 86,228,152,16,2
         ,132,207,96,230,242
HE 1350  DATA 165,242,24,105,
         128,168,169,0,105,5
LO 1360  DATA 157,69,3,152,15
         7,68,3,96,224,2
PN 1370  DATA 225,2,41,31
```

©

COMPUTE!
TOLL FREE
Subscription
Order Line
1-800-247-5470
In IA 1-800-532-1272

www.commodore.ca

MODified Shapes For IBM

Paul W. Carlson

Ever wonder how to use the MOD operator in IBM BASIC? Here's a short explanation of MOD operations that create some stunning graphics displays as well. The programs run on any IBM PC with color/graphics adapter and BASICA or PCjr with Cartridge BASIC.

Though you may never have seen it used, IBM BASIC's MOD operator is a powerful tool for performing certain arithmetic operations. Simply put, MOD gives the integer (whole number) remainder of an integer division. For example, 17 MOD 3 = 2 because 17 divided by 3 equals 5 with a remainder of 2. Likewise, 30 MOD 5 = 0 since 5 divides into 30 evenly with a remainder of zero.

To see some applications of MOD arithmetic, type in, save, and run Programs 1–4 below. Each generates a different geometric display (see photos), using MOD in various ways to simplify the graphics calculations.

Although some dialects of BASIC don't include a MOD operator, the INT function can be used for the same purpose. In Microsoft BASIC, the expression K-INT(K/J)*J gives exactly the same result as the IBM BASIC expression K MOD J. If your computer's BASIC doesn't have MOD but does have a line-drawing command (the Commodore 128 and Atari fall into this category), you may find it interesting to convert the programs to run on your machine.

What Does MOD Do?

One of the most common uses of MOD is to test whether a value is odd or even. The expression X MOD 2 yields a 1 if X is odd, or 0 if X is even. Program 2 uses MOD for this purpose in line 20, branching to line 40 only when the variables I and J are both odd or even.

Another common use of MOD is to make a variable equal to a repeating sequence of consecutive integers. Consider these statements:

```
10 C=0
20 C=C MOD 3+1
30 PRINT C
40 GOTO 20
```

In this case the variable C cycles repeatedly through the series 1-2-3. Program 1 uses MOD in this manner to select the right color for each side of the rotating triangles.

MOD is also useful when arrays must be treated as circular rather than linear. For example, say you have a numeric array X composed of three array elements, and that elements X(1), X(2), and X(3) contain the X coordinates for the vertices (corners) of a triangle. In this case, if X(*m*) is the X coordinate for the beginning of a side, then the expression X(*m* MOD 3+1) gives the X coordinate for the end of that side. This sort of expression appears in Programs 1, 2, and 3, which compute the variable NJ with MOD. The result becomes an index into the arrays containing the vertex coordinates whenever the program needs to know the values for the end of a side.

How The Programs Work

After running the example programs, you may want to know how to produce similar displays of your own. The following explanation applies to Programs 1, 2, and 3, which have been made as similar as possible for comparison purposes.

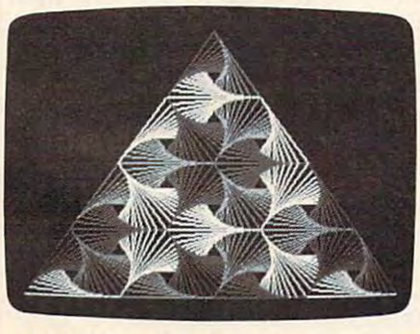
The variable SU selects the spot on the side of a figure where the figure's corner will land after it is rotated and redrawn. If you're not sure what the last sentence means, try changing SU to a slightly different value and rerunning the program: You'll see exactly what it does. The variables I and J represent the row and column of the figure. The arrays X and Y contain the relative vertex coordinates for the current rotation. The arrays XD and YD contain relative vertex coordinates for the next rotation. N is the current rotation, M is the current side of a polygon, and NJ performs the function described above.

Here is a line-by-line description of the various sections in each program:

www.commodore.ca

Lines 10–50	Loop through J columns and I rows. Skip over some combinations of I and J (Programs 1 and 3). Change values in the Y array to plot some polygons upside-down (Programs 1 and 2).
Line 60	Loop through N rotations. Compute the absolute coordinates of the first vertex.
Line 70	Loop through M sides. Compute the absolute coordinates for the next vertex.
Line 80	Plot the side. Make the absolute coordinates of the beginning of the next side equal to the absolute end coordinates of the side just plotted.
Line 90	Compute the relative coordinates of the vertex that fall on this side in the next rotation.
Line 100	Move the values from the array of relative coordinates for the next rotation into the array of relative coordinates for the current rotation.

Program 4 works somewhat differently from the other three programs. After saving and running this program, remove the statement C=1 from line 50 and run it again. This time the colors take on a spiral pattern. Can you explain why? (Hint: Ten areas are painted with three colors each time through the FOR A=10 TO 360 loop, and 10 MOD 3 = 1.)

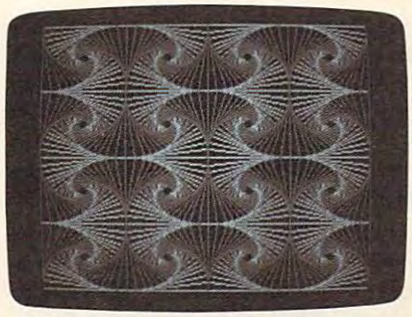


"MODified Shapes For IBM" shows how to easily generate complex graphics displays like this.

For instructions on entering these listings, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

Program 1

```
DK 10 SU=.1:RU=1-SU:KEY OFF:SCRE
      EN 1:COLOR 0,1:II=1:C=1
FO 20 FOR J=0 TO 3:II=-II:JJ=1:F
      OR I=0 TO 6:JJ=-JJ:IF I<J
      OR I>6-J THEN 110
EB 30 IF J<2 OR I>2 THEN C=C MOD
      3+1
FP 40 IF J=3 THEN C=C MOD 3+1
GI 50 X(1)=0:X(2)=39:X(3)=78:Y(1
      )=0:Y(3)=0:IF II=JJ THEN Y
      (2)=48 ELSE Y(2)=-48
EO 60 FOR N=1 TO 11:X1=3+X(3)+I*
      39:Y1=175-Y(3)-J*48+II*JJ*
      24
DL 70 FOR M=1 TO 3:X2=3+X(M)+I*3
      9:Y2=175-Y(M)-J*48+II*JJ*2
      4:C=C MOD 3+1
FD 80 LINE(X1,Y1)-(X2,Y2),C:X1=X
      2:Y1=Y2:NJ=M MOD 3+1
FK 90 XD(M)=RU*X(M)+SU*X(NJ):YD(
      M)=RU*Y(M)+SU*Y(NJ):NEXT M
KB 100 FOR P=1 TO 3:X(P)=XD(P):Y
       (P)=YD(P):NEXT P,N
CF 110 NEXT I,J
PP 120 IF INKEY$="" THEN 120 ELS
       E CLS:SCREEN 0:WIDTH 80:K
       EY ON:END
```



The MOD operator simplifies the calculations needed to produce this display.

Program 2

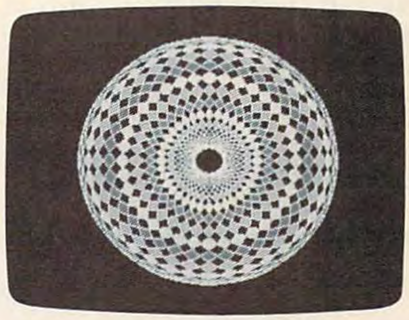
```
LG 10 SU=.12:RU=1-SU:KEY OFF:CLS
      :SCREEN 1:COLOR 0,1
MI 20 FOR I=0 TO 3:FOR J=0 TO 3:
      IF I MOD 2=J MOD 2 THEN 40
ME 30 Y(1)=49:Y(2)=0:Y(3)=0:Y(4)
      =49:GOTO 50
IE 40 Y(1)=0:Y(2)=49:Y(3)=49:Y(4
      )=0
HH 50 X(1)=20:X(2)=20:X(3)=89:X(
      4)=89
FO 60 FOR N=0 TO 18:X1=X(4)+I*69
      :Y1=Y(4)+J*49
PL 70 FOR M=1 TO 4:X2=X(M)+I*69:
      Y2=Y(M)+J*49
BK 80 LINE(X1,Y1)-(X2,Y2),M MOD
      2+1:X1=X2:Y1=Y2:NJ=M MOD 4
      +1
FK 90 XD(M)=RU*X(M)+SU*X(NJ):YD(
      M)=RU*Y(M)+SU*Y(NJ):NEXT M
EO 100 FOR P=1 TO 4:X(P)=XD(P):Y
       (P)=YD(P):NEXT P,N,J,I
EP 110 IF INKEY$="" THEN 110 ELS
       E CLS:SCREEN 0:WIDTH 80:E
       ND
```

Program 3

```
PN 10 SU=.2:RU=1-SU:KEY OFF:CLS:
      SCREEN 1:COLOR 0,1
HF 20 FOR J=0 TO 2:FOR I=0 TO 2:
      IF J=0 AND I<>1 THEN 110
ML 30 IF I=1 THEN E=31 ELSE E=0
```

This screen is produced with only a dozen program lines.

```
DA 40 X(1)=0:X(2)=25:X(3)=75:X(4
      )=100:X(5)=75:X(6)=25
KB 50 Y(1)=31:Y(2)=0:Y(3)=0:Y(4)
      =31:Y(5)=62:Y(6)=62
CG 60 FOR N=0 TO 20:X1=35+X(6)+I
      *75:Y1=223-Y(6)-J*62-E
AC 70 FOR M=1 TO 6:X2=35+X(M)+I*
      75:Y2=223-Y(M)-J*62-E
JG 80 LINE(X1,Y1)-(X2,Y2),M MOD
      3+1:X1=X2:Y1=Y2:NJ=M MOD 6
      +1
FK 90 XD(M)=RU*X(M)+SU*X(NJ):YD(
      M)=RU*Y(M)+SU*Y(NJ):NEXT M
NB 100 FOR P=1 TO 6:X(P)=XD(P):Y
       (P)=YD(P):NEXT P,N
CF 110 NEXT I,J
PP 120 IF INKEY$="" THEN 120 ELS
       E CLS:SCREEN 0:WIDTH 80:K
       EY ON:END
```



A multicolored screen display with help from MOD.

Program 4

```
MK 10 KEY OFF:CLS:SCREEN 1:COLOR
      0,0:FOR K=0 TO 9:READ F(K
      ):NEXT
DL 20 FOR A=10 TO 360 STEP 10
NG 30 X=160+48*COS(A*.017453):Y=
      100+40*SIN(A*.017453)
QM 40 CIRCLE(X,Y),60:NEXT
OI 50 FOR A=10 TO 360 STEP 10:C=
      1:FOR K=0 TO 9:C=C MOD 3+1
FD 60 X=160+1.2*F(K)*COS(A*.0174
      53):Y=100+F(K)*SIN(A*.0174
      53)
PJ 70 PAINT(X,Y),C,3:NEXT K,A
NI 80 IF INKEY$="" THEN 80 ELSE
      CLS:SCREEN 0:WIDTH 80:KEY
      ON:END
KI 90 DATA 26,34,41,51,61,68,78,
      84,87,89
```

©

www.commodore.ca

Custom Characters For Atari *SpeedScript*

Charles Brannon, Program Editor

Atari SpeedScript*'s large-sized character set with true descenders is designed for maximum readability. Since it is built into* SpeedScript*, though, it's difficult to change. Now, with the accompanying program, you can add your own custom font to* SpeedScript*—even special characters for foreign languages. To help you design your own sets, last month we published "Atari FontMaker," a versatile character editor written completely in machine language. This month's program automatically creates a new copy of* SpeedScript *with your custom set installed. It works on any Atari 400/800, XL, or XE computer with at least 24K RAM and a disk drive.*

When you're writing, you probably don't think about the technical miracles that make word processing possible. The characters on the screen seem as fixed as those on paper, but they're actually displayed from a section of memory. If you can change this memory, you can change the shape, form, and order of the characters.

Normally, the Atari's character set is permanently and unalterably stored in Read Only Memory (ROM). But you can tell the video chip to look elsewhere for the character patterns, so it's possible to substitute your own set. Atari *SpeedScript* uses just such a technique, coupled with a special Atari character mode, to display large, well-formed characters.

One way to design your own characters is to sketch them on grid paper, convert the resulting binary patterns into numbers, and then substitute these numbers in the section of *SpeedScript* that modifies the character set.

Fortunately, there's an easier way. You can use a character set editor (or *font* editor) to "draw" the desired characters with a joystick. The font editor converts the patterns into numbers for you and saves the new set on disk for later use. There are many font editors available for the Atari, but relatively few of them let you design characters for the unusual ANTIC 3 mode that *SpeedScript* uses. This mode displays a taller character grid so you can design characters with true *descenders* (the protruding tails on letters such as j and y).

Last month's COMPUTE! introduced "Atari FontMaker," a full-featured font editor written completely in machine language. FontMaker lets you design custom fonts for all Atari character modes, including ANTIC mode 3. The FontMaker article included some instructions for creating *SpeedScript* ANTIC 3 characters, and also some tips for designing more readable characters. After designing a new character set with FontMaker, you can use Program 1, "*SpeedScript* Characterizer," to automatically create a new copy of *SpeedScript* with your custom font. The custom characters appear each time you load the modified copy of *SpeedScript*.

Making A *SpeedScript* Font

If you want to convert a normal character set for use with *SpeedScript* (perhaps one that you've designed with another font editor), start by loading the set into FontMaker. Press G to switch to the ANTIC 3 mode. (The G command cycles through the graphics modes, and the first mode after the default GRAPHICS 0 mode is ANTIC 3.) You'll recognize this mode because the normal characters appear split, with their lower portions shifted to the top of the character grid.

Use FontMaker's *roll up* command (CTRL-cursor up) to move each character up by one line. Roll up automatically wraps the lowercase descenders. If you want to make two-line descenders such as those found in the *SpeedScript* font, roll the lowercase letters up by two notches, erasing the last line of any characters that intrude into the descender zone, such as the lowercase h. Then *shift up* (SHIFT-cursor up) all the uppercase characters by two lines, and add an extra line of definition to each character so they have the same baseline as the lowercase characters. As you work, refer to the "quick brown fox" sentence displayed by FontMaker to see how your characters match up. When you're done, save your new font on disk with the S (Save) command.

One additional preparatory step is required if your copy of *SpeedScript* came from the May 1985

www.commodore.ca

COMPUTE! DISK. A version of *SpeedScript* typed in and saved with MLX (or from the companion disk for the book *SpeedScript: The Word Processor for Atari Computers*) appears on disk as one continuous block and requires no special preparation. The COMPUTE! DISK version is stored in an alternative format—as a collection of linked segments. (For a discussion of the differences between these formats, see Bill Wilkinson's "Insight: Atari" column in the April 1986 issue.) This causes a problem because Program 1 expects to load *SpeedScript* as a continuous file.

The solution is to use Program 2, a corrected version of the file unification program from the April "Insight" column, to convert the linked segment file into a continuous block file. If you're unsure about the origin of your copy of *SpeedScript*, it won't hurt to process it with the file unifier program anyway. If the file was already a continuous block, Program 2 just produces a duplicate copy. (One way to tell the difference: In DOS 2.0 format the desired continuous block version of *SpeedScript* is 67 blocks long, while the linked segment version requiring conversion is 66 blocks long.)

Now you're ready to merge your redefined character set with *SpeedScript*. Type in and save the *SpeedScript* Characterizer (Program 1) below. After you run it, insert the disk containing your copy of *SpeedScript* (the unified copy produced by Program 2 if you have the COMPUTE! DISK version) and type in the filename you use for *SpeedScript* (AUTORUN.SYS, for example). Characterizer loads *SpeedScript* into a memory buffer.

Next, insert the disk containing your character set and enter its filename. After this character set is loaded, insert the disk that will store the new copy of *SpeedScript*. Type in the filename you'd like to call it. (For safety's sake, be very careful not to replace your existing copy of *SpeedScript*. If you have to name *SpeedScript* as AUTORUN.SYS to run it with Atari DOS 2.0 or 2.5, use another disk or rename the original copy of *SpeedScript* something else before you try to use Program 1.)

Finally, boot up the new *SpeedScript* and behold your custom character set. Try typing all the characters to make sure they are well-formed. If you made a mistake, you can always edit the character set and reinstall it.

Keep in mind that *SpeedScript* uses the heart character to pad out the end of each line with spaces. If you don't want to see a screenful of hearts, make sure you've blanked out the heart character in your set. Also, *SpeedScript* normally uses the left arrow (cursor-left) for the return mark at the end of each paragraph, so you may want to change this to some personalized symbol. If you don't want to see return marks, blank out this character, or perhaps change it to a single tiny dot so you can easily delete what would otherwise be an invisible character.

Program 1: *SpeedScript* Characterizer

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
JK 10  OPEN #3,4,0,"K:"
LG 15  DIM CIO$(7):CIO$="hhh*
       LV":CIO$(4,4)=CHR$(170
       ):CIO$(7)=CHR$(228)
AL 100 DIM SF$(14),T$(40),SS
       $(10000),CS$(14)
CJ 110 GRAPHICS 17:POKE 82,0
       :SETCOLOR 4,0,10:SETC
       OLOR 2,0,2:SETCOLOR 0
       ,3,4:SETCOLOR 1,5,14
LJ 120 ? #6;"SPEEDSCRIPT 3.0
       ":? #6;"CHARACTERIZER
       "
IK 130 POKE 87,0:? "ENTER SP
       EEDSCRIPT{3 SPACES}FI
       LENAME:"
AN 140 INPUT SF$:IF SF$="" T
       HEN RUN
FM 150 IF LEN(SF$)>2 THEN IF
        SF$(2,2)=":" OR SF$(
       3,3)=":" THEN 170
CP 160 T$=SF$:SF$="D:":SF$(3
       )=T$
EA 170 TRAP 180:OPEN #2,4,0,
       SF$:TRAP 40000:GOTO 1
       90
OE 180 ? "error trying to op
       en":? SF$:? "PRESS RE
       TURN":GET #3,A:RUN
IC 190 LET READ=1:X=32:SADR=
       ADR(SS$):MAXLEN=9999:
       GOSUB 970:SS$(TRUELEN
       )=CHR$(0):CLOSE #2
LC 200 ? CHR$(125);"ENTER NA
       ME OF":? "CHARACTER S
       ET."
KC 210 INPUT CS$:IF CS$="" T
       HEN 200
EP 220 IF LEN(CS$)>2 THEN IF
        CS$(2,2)=":" OR CS$(
       3,3)=":" THEN 240
CE 230 T$=CS$:CS$="D:":CS$(3
       )=T$
DH 240 TRAP 250:OPEN #2,4,0,
       CS$:TRAP 40000:GOTO 2
       60
NP 250 ? CHR$(125);"error tr
       ying to open":? CS$:?
        "PRESS RETURN":GET #
       3,A:CLOSE #2:GOTO 200
PF 260 LET READ=1:X=32:MAXLE
       N=1024:SADR=ADR(SS$)+
       262:GOSUB 970:CLOSE #
       2
PN 270 ? CHR$(125);
JO 280 ? "ENTER NEW NAME":?
       "FOR SPEEDSCRIPT."
LA 290 INPUT SF$:IF SF$="" T
       HEN 200
FG 300 IF LEN(SF$)>2 THEN IF
        SF$(2,2)=":" OR SF$(
       3,3)=":" THEN 320
CM 310 T$=SF$:SF$="D:":SF$(3
       )=T$
DG 320 TRAP 335:OPEN #2,8,0,
       SF$:TRAP 40000
BF 330 LET READ=0:X=32:MAXLE
       N=LEN(SS$):SADR=ADR(S
       S$):GOSUB 970:GOTO 34
       0
AL 335 IF PEEK(195)<>1 THEN
       ? CHR$(125);"CAN'T WR
       ITE FILE":CLOSE #2:GO
       TO 280
FP 340 POKE 82,2:GRAPHICS 0:
       ? "Good luck!"
GP 350 END
IN 899 GOTO 899
MB 970 REM CIO Load/Save Fil
       e#2 opened  READ=0 fo
       r write, READ=1 for r
       ead
AM 980 REM File#2,$20
EF 990 ICCOM=834:ICBADR=836:
       ICBLEN=840:ICSTAT=835
AL 1000 H=INT(SADR/256):L=SA
        DR-H*256:POKE ICBADR
        +X,L:POKE ICBADR+X+1
        ,H
FC 1010 H=INT(MAXLEN/256):L=
        MAXLEN-H*256:POKE IC
        BLEN+X,L:POKE ICBLEN
        +X+1,H
MD 1020 POKE ICCOM+X,11-4*RE
        AD:A=USR(ADR(CIO$),X
        )
BK 1025 TRUELEN=PEEK(ICBLEN+
        X)+256*PEEK(ICBLEN+X
        +1)+1
BG 1030 POKE 195,PEEK(ICSTAT
        ):RETURN
```

Program 2: File Unifier

Program by Bill Wilkinson

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
EE 1000 REM ** BINARY FILE U
        NIFIER **
DF 1010 BUFSIZE=FRE(0)-300
AH 1020 DIM BUF$(BUFSIZE)
IF 1030 DIM FILEOLD$(40),FIL
        ENEW$(40)
FJ 1040 FILL=0
NO 1050 PRINT "I need two fi
        le names: An existin
        g"
EA 1060 PRINT "  object file
         and a new file whic
        h"
EE 1070 PRINT "  will get th
        e 'unified' object c
        ode."
FG 1080 PRINT
AA 1090 PRINT "Existing file
        ? ";
```


www.commodore.ca

```
CL 1100 INPUT #16,FILEOLD$
NI 1110 PRINT "{5 SPACES}New file? ";
DI 1120 INPUT #16,FILENEW$
FG 1130 OPEN #1,4,0,FILEOLD$
NP 1140 GET #1,HDR1:GET #1,HDR2
PD 1150 IF HDR1=255 AND HDR2=255 THEN 1180
PF 1160 PRINT :PRINT "Existing file: invalid format"
KA 1170 END
GL 1180 OPEN #2,8,0,FILENEW$
BI 1190 PUT #2,HDR1:PUT #2,HDR2
IN 1200 GET #1,SEGLOW:GET #1,SEGHIGH
BC 1205 REM ** Process a new origin
AE 1210 BUFPTR=0
OI 1220 BUF$=CHR$(0):BUF$(BUFSIZE)=CHR$(0)
GL 1230 BUF$(2)=BUF$:REM zap buffer
MF 1240 PUT #2,SEGLOW:PUT #2,SEGHIGH
DE 1250 SEGSTART=SEGLOW+256*SEGHIGH
DI 1255 REM ** Process a segment
ID 1260 GET #1,ENDLOW:GET #1,ENDHIGH
GP 1270 SEGEND=ENDLOW+256*ENDHIGH
HC 1280 SEGLEN=SEGEND-SEGSTART+1
HI 1290 FOR PTR=1 TO SEGLEN
JL 1300 GET #1,BYTE:BUF$(BUFPTR+PTR)=CHR$(BYTE)
PK 1310 NEXT PTR
ML 1320 TRAP 1490
JB 1330 GET #1,SEGLOW:GET #1,SEGHIGH
PD 1340 TRAP 40000
KB 1350 IF SEGLOW=255 AND SEGHIGH=255 THEN GET #1,SEGLOW:GET #1,SEGHIGH
DG 1360 SEGSTART=SEGLOW+256*SEGHIGH
IO 1370 GAP=SEGSTART-SEGEND-1
HD 1380 IF GAP>FILL OR GAP<0 THEN 1420
JM 1390 BUFPTR=BUFPTR+SEGLEN+GAP
DL 1400 IF BUFPTR+256>BUFSIZE THEN 1420
MI 1410 GOTO 1260
BA 1420 GOSUB 1440
MF 1430 GOTO 1210
BF 1435 REM ** Dump buffer
LH 1440 PUT #2,ENDLOW:PUT #2,ENDHIGH
OJ 1450 FOR PTR=1 TO LEN(BUF$)
LF 1460 PUT #2,ASC(BUF$(PTR))
AB 1470 NEXT PTR
KN 1480 RETURN
LA 1485 REM ** Error encountered
EA 1490 IF PEEK(195)=136 THEN 1520
NL 1500 PRINT "Unexpected error, number ";PEEK(195)
JO 1510 END
JA 1515 REM ** End of file
CM 1520 GOSUB 1440:REM write out buffer
KA 1530 END
```

©

64 Autobooter

Terry Roper

Now you can throw away your list of start-up SYS addresses for different machine language programs. Using a unique loading method, this clever utility lets you transform a BASIC or machine language program into an autobooting *program that runs as soon as it loads—no matter where it normally loads into memory. In addition, the autobooting program can survive many system crashes that would destroy a conventionally loaded program. A disk drive is required.*

"64 Autobooter" can add autobooting—the ability to run automatically after loading—to many different BASIC and machine language programs. While you may be familiar with other autoboot utilities, the most common techniques have significant drawbacks: Either they require a separate boot file to load and run the main program, or they increase the size of your program significantly. But Autobooter programs don't suffer from either problem. The autobooting file is all in one piece and is only 257 bytes longer than the original program. Even better, Autobooter programs can survive unscathed in situations that would erase or scramble programs loaded the normal way.

Creating Autoboot Files

Type in 64 Autobooter and save a copy before you run it. Using the program is just a matter of responding to a few prompts. First, you type a filename for the program you wish to convert. This file may be any program file on disk, but it may not exceed 7,935 bytes in length (more on this later). As a rough guide, 7,935 bytes take up about 31 blocks on the disk directory.

After you supply the filename, the disk drive makes two passes through the program file. The first pass counts every byte of the file to make sure that it's within the 7,935-byte limit and calculates the load address for the new Autobooter version. The second pass stores the file in memory. When the drive stops, you are asked whether the program is BASIC or machine language (ML). If the program is all BASIC (or it begins with a BASIC line like SYS 2061), type B to choose BASIC. If it is an ML program that loads at some address *other than* the start of BASIC, enter the normal SYS address when prompted.

Finally, you'll be prompted to enter a filename for the new autobooting file. At this point, the program writes the autobooting file back to disk, using the last filename you specified. The new file consists of a copy of the original file plus 257 bytes (about one disk block) of machine language. This one-block addition is the only penalty you pay for the convenience of using Autobooter files.

When this process is complete, the new file is ready to use. An autobooting file is a self-sufficient, autobooting version of the original, but with some unusual differences. The most unusual fact is that all Autobooter files share exactly the same ending address, even though nearly all of them have different load addresses. And loading ML files with the Autobooter method does not disturb the computer's start-of-variables pointer at locations 45–46 ($2D–2E). So you can load an ML program with the assurance that any BASIC program previously in memory will still run and save correctly.

An Impressive Demonstration

Here's a short demonstration that shows how useful 64 Autobooter can be. Choose a BASIC program that you would like to convert to Autobooter form. It can be any program, as long as the original file contains fewer than 7,935 bytes.

www.commodore.ca

Follow the procedure for converting the program to Autobooter format, then perform these steps:

1. Type NEW.

2. Type LOAD "FILENAME",8,1 (replace FILENAME with the filename of your autobooting file). The program should load and run automatically; if it doesn't, reload 64 Autobooter and check for typing errors.

3. When you're satisfied that your program is running as usual, stop the program by pressing RUN/STOP (or RUN/STOP–RESTORE if necessary).

4. Now it's time to get tough. Type NEW, then LIST to confirm that the program is gone. Type POKE 1,0 and press RETURN. The program should be running once again, even though that POKE is one that normally locks up the computer.

5. Press RUN/STOP to stop the program, then load any other BASIC program into memory. This ensures that the first program is completely overwritten by new data.

6. Type POKE 1,0 and press RETURN. Immediately, the autobooted program reappears in memory and begins to run. You should be able to VERIFY at this point with normal results.

What's going on here? For one thing, you've seen how durable these files can be. Even after you destroy the program in its normal location, you can resurrect it instantly with POKE 1,0. Another advantage is that you no longer need to memorize different SYS addresses for every machine language program. Regardless of where an Autobooter program normally loads, whether it's BASIC or ML, you can restart it with POKE 1,0.

It's not hard to see how this would come in handy. For instance, suppose you're using a BASIC program. Suddenly, the need arises to run a program that occupies the same memory area. If the original program is in Autobooter format, this presents no problem at all. Load the second program and run it as usual. When you're finished, type POKE 1,0 to restore the autobooted program.

A second advantage has to do with loading ML files. Under ordinary circumstances, loading an ML program into high memory after loading a BASIC program can cause an OUT OF MEMORY error when you try to run the BASIC program. If the ML program is an Autobooter file, you'll avoid this error.

Wraparound Addressing

By now you might be wondering how these unusual features are possible. We mentioned earlier that Autobooter files may not exceed 7,935 bytes in length. This is necessary for two reasons. The 257-byte ML program attached to your original file must always load into memory beginning at location $FF00 (65280). The rest of your original file comes into memory directly below the ML code in the area from $E000–$FEFF (57344–65279). As you may know, loading a program into this zone causes it to be stored in the RAM underlying the Kernal ROM chip.

Why is it so crucial to load the last 257 bytes at $FF00? If you're paying close attention, you may have noticed that 257 plus $FF00 equals more than 65535, the highest address that the 64's microprocessor can refer to. After you load the byte that goes into location $FFFF, there are still two bytes to load. Since it's impossible to use a higher address, the computer wraps around to the bottom of its addressing range and loads the last two bytes into addresses $00 and $01.

Loading a file in this wraparound fashion has two important consequences. It allows you to change the contents of the 64's on-chip I/O port (location 1) as well as the vector to its main IRQ handler ($FFFE–$FFFF). Let's look at location 1 first. The chief function of location 1 is to control whether the 64 "sees" ROM and I/O chips or the underlying RAM at memory locations above $A000 (40960). Under normal circumstances, the computer sees BASIC ROM at locations $A000–BFFF, I/O chips at locations $D000–DFFF, and the Kernal ROM at locations $E000–FFFF. When you load an Autobooter file into memory, it loads the value $35 into location 1, switching out the ROM in locations $E000–$FFFF and exposing the underlying RAM.

Once the ROM is switched out, the microprocessor can see the rest of the Autobooter file that was just loaded. Among other things, this file loaded new information into locations $FFFE–$FFFF. This pair of locations contains a *vector*, or two-byte address, which points to the 64's main IRQ/BRK handler routine in ROM. Sixty times every second, and whenever the computer encounters a BRK instruction, the computer jumps to the address pointed to by this vector. Autobooter files replace the normal vector with a new one that points to the new ML that we just loaded into underlying RAM.

The rest of the ML program moves the autobooted program to its normal load address, switches ROM back in (by putting a value of $37 in location 1), and runs the program as usual. (Incidentally, the Commodore 128 normally has a value of $77 in location 1 while it's in 64 mode. This has no effect on Autobooter files, but does prevent certain commercial programs from working properly. Pressing CAPS LOCK restores the normal value of $37, which fixes the problem in at least some of the cases.)

Mysterious VERIFY Errors

Although the computer normally resets the computer's start-of-variables pointer after every load, 64 Autobooter does this job on its own. If the program is ML which doesn't load at the start of BASIC, the start-of-variables pointer is not changed. This lets you load ML programs without disturbing any BASIC program in memory. If the Autobooter file is BASIC, the pointer is set to the end of the program text as usual.

Some Autobooter files may give you mysterious VERIFY errors. Don't panic—it's not uncommon for ML programs to alter themselves once run. Naturally, a file that's been modified isn't the same as the original file on disk. If a BASIC program was saved from an abnormal location it won't VERIFY either; this is due to the relinking process that occurs after every BASIC program is loaded. A good method for initial testing of ML files is to create them as usual but enter 42100 instead of the normal SYS address. This returns you to BASIC

www.commodore.ca

without starting your program; the file should VERIFY correctly.

Here's a subtle bug to look out for. Suppose you have the following program:

100 A0 = 828: POKE AO, 0

This defines the variable A0, but not the variable AO (with the letter O instead of a zero). Since undefined numeric variables always equal zero, the POKE is directed to location 0, the data direction register for location $01. If you run this program and then try to start an autobooting program with POKE 1,0, it fails because the new bit pattern at location 0 prevents any new value from being stored in location 1. Hit RUN/STOP–RESTORE to fix both locations, then try again.

As a matter of fact, it doesn't hurt to hit RUN/STOP–RESTORE before activating any autobooting program. Many ML programs change the IRQ vector at locations $0314–$0315. If you try to activate an Autobooter file under these conditions, the computer will probably crash as soon as the interrupt is reenabled.

Finally, while some cartridges don't interfere with Autobooter files, others cause erratic behavior. Cartridges can reconfigure the computer in many different ways, so you'll have to find out by trial and error whether a particular Autobooter file works with a given cartridge.

64 Autobooter

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
BC 100 POKE 55,Ø:POKE 56,48:CL
       R
HR 110 PRINT:PRINT "AUTO BOOTM
       AKER":IFPEEK(24591)=164
       THEN15Ø
GP 120 PRINT:PRINT"JUST A MOME
       NT..."
DE 130 FOR J=12288 TO 12744:RE
       AD D:S=S+D:POKE J,D:NEX
       T
JE 140 IF S<>51414 THEN PRINT:
       PRINT"MISTYPED DATA":EN
       D
GF 150 OPEN 15,8,15:SYS 12288
GX 160 PRINT:PRINT"FILE FOR AU
       TO BOOTMAKER";
SJ 170 GOSUB37Ø:PRINT:PRINT"ME
       ASURING YOUR FILE"
AJ 180 OPEN 5,8,5,"Ø:"+N$+",P,
       R":GOSUB35Ø:IF E1$<>"ØØ
       " THEN16Ø
CR 190 SYS 24834:CLOSE 5:B=PEE
       K(24578)+256*PEEK(24579
       ):PRINT B;"BYTES"
AG 200 IF B>7935 THEN PRINT CH
       R$(18);STR$(B-7935)"
       {2 SPACES}BYTE OVERFLOW
       ":GOTO16Ø
CR 210 PRINT:PRINT"READING FIL
       E INTO MEMORY"
PF 220 OPEN 5,8,5,"Ø:"+N$+",P,
       R":SYS24834:CLOSE 5
BH 230 FOR J=Ø TO 31:POKE24592
       +J,PEEK(24112+J):POKE24
       624+J,PEEK(22Ø32+J)
HH 240 POKE 22Ø32+J,Ø:NEXT
CE 250 PRINT:PRINT"BASIC OR ML
        (B/M)";:GOSUB37Ø
QR 260 IF N$="B" THEN POKE 245
       82,1:POKE 24583,8:GOTO3
       ØØ
EH 270 POKE 24587,1:PRINT:PRIN
       T" JUMP TO ADDRESS";:GO
       SUB37Ø
JG 280 H%=(VAL(N$)-1)/256:L%=(
       VAL(N$)-1)-H%*256
FR 290 POKE 24588,L%:POKE 2458
       9,H%
BF 300 PRINT:PRINT"PREPARING T
       O WRITE NEW FILE"
XK 310 PRINT"INSERT ANOTHER DI
       SK IF DESIRED"
SA 320 PRINT:PRINT"NEW FILENAM
       E";:GOSUB37Ø:PRINT#15,"
       IØ"
PE 330 OPEN 5,8,5,"Ø:"+N$+",P,
       W":GOSUB35Ø:IF E1$<>"ØØ
       "THEN3ØØ
JF 340 SYS 2495Ø:CLOSE5:CLOSE1
       5:END
PP 350 INPUT#15,E1$,E2$,E3$,E4
       $:IF E1$="ØØ"THENRETURN
MP 360 PRINT CHR$(18);E1$","E2
       $","E3$","E4$:CLOSE5:RE
       TURN
MC 370 T=POS(T)
FF 380 FORJ=ØTO25:WAIT 197,64:
       NEXT:POKE 198,Ø:PRINT T
       AB(T)
AJ 390 N$="":INPUT N$:IF N$=""
       THENPRINT:PRINT:PRINT:G
       OTO38Ø
AS 400 RETURN
KS 410 DATA 162,Ø,189,18,48,15
       7,Ø,96
BG 420 DATA 189,18,49,157,Ø,97
       ,232,2Ø8
XK 430 DATA 241,96,Ø,Ø,Ø,Ø,Ø,Ø
CK 440 DATA Ø,Ø,4,93,255,Ø,227
       ,167
KB 450 DATA 115,164,Ø,Ø,Ø,Ø,Ø,
       Ø
RM 460 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
KK 470 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
FJ 480 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
BX 490 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
JJ 500 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
EX 510 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
AS 520 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
DX 530 DATA Ø,Ø,162,25Ø,154,16
       2,7,189
KH 540 DATA 8,255,72,2Ø2,16,24
       9,64,162
MG 550 DATA 47,224,32,176,12,1
       89,16,255
KJ 560 DATA 157,48,253,189,48,
       255,157,16
MM 570 DATA 245,189,196,255,15
       7,Ø,2,2Ø2
KQ 580 DATA 16,231,173,11,255,
       2Ø8,14,173
CC 590 DATA 2,255,1Ø5,1,133,45
       ,173,3
HG 600 DATA 255,1Ø5,8,133,46,1
       74,4,255
XG 610 DATA 172,5,255,134,251,
       132,252,174
KD 620 DATA 6,255,172,7,255,13
       4,253,132
GM 630 DATA 254,16Ø,Ø,174,3,25
       5,24Ø,14
PX 640 DATA 177,251,145,253,2Ø
       Ø,2Ø8,249,23Ø
SQ 650 DATA 252,23Ø,254,2Ø2,2Ø
       8,242,174,2
EE 660 DATA 255,24Ø,8,177,251,
       145,253,2ØØ
CF 670 DATA 2Ø2,2Ø8,248,76,Ø,2
       ,169,55
CM 680 DATA 133,1,88,169,Ø,32,
       189,255
JG 690 DATA 169,15,168,162,8,3
       2,186,255
CQ 700 DATA 32,192,255,169,15,
       32,195,255
CG 710 DATA 1Ø4,2Ø8,11,32,51,1
       65,162,44
DA 720 DATA 16Ø,2,134,122,132,
       123,169,13
PC 730 DATA 76,21Ø,255,138,Ø,Ø
       ,Ø,Ø
RH 740 DATA Ø,Ø,Ø,Ø,92,255,92,
       255
KR 750 DATA 8Ø,255,47,53,173,1
       78,97,73
XG 760 DATA 255,141,178,97,174
       ,179,97,172
PC 770 DATA 18Ø,97,134,251,132
       ,252,162,5
PB 780 DATA 32,198,255,32,228,
       255,141,6
ME 790 DATA 96,32,228,255,141,
       7,96,16Ø
GF 800 DATA Ø,32,228,255,44,17
       8,97,48
JK 810 DATA 7,145,251,44,178,9
       7,16,22
XM 820 DATA 238,2,96,2Ø8,3,238
       ,3,96
QA 830 DATA 56,173,179,97,233,
       1,141,179
PR 840 DATA 97,176,3,2Ø6,18Ø,9
       7,32,183
EF 850 DATA 255,41,64,2Ø8,8,2Ø
       Ø,2Ø8,2Ø9
PB 860 DATA 23Ø,252,76,37,97,1
       73,178,97
RQ 870 DATA 24Ø,19,56,173,181,
       97,237,2
DM 880 DATA 96,141,4,96,173,18
       2,97,237
DF 890 DATA 3,96,141,5,96,76,2
       Ø4,255
JQ 900 DATA 56,169,Ø,237,2,96,
       133,251
HD 910 DATA 169,96,237,3,96,13
       3,252,162
BA 920 DATA 5,32,2Ø1,255,173,4
       ,96,32
KA 930 DATA 21Ø,255,173,5,96,3
       2,21Ø,255
MG 940 DATA 16Ø,Ø,177,251,32,2
       1Ø,255,23Ø
GM 950 DATA 251,2Ø8,2,23Ø,252,
       165,252,2Ø1
QG 960 DATA 97,2Ø8,239,165,251
       ,2Ø1,2,2Ø8
SX 970 DATA 233,76,2Ø4,255,Ø,Ø
       ,96,Ø
AP 980 DATA 255
```

©

www.commodore.ca

Upgrading The Apple ESCape Key

Robert Jacques Beck

With this short routine, you can reprogram the Apple's ESCape key (or any other key) to perform a variety of special functions within Applesoft BASIC. It works with any Apple II-series computer using Applesoft BASIC and DOS 3.2, DOS 3.3, or ProDOS.

Have you ever wished you could detect special escape codes in Applesoft BASIC? For instance, say that you have an Applesoft program that uses INPUT to ask you for a disk filename. In many cases, it would be handy to view the disk catalog before entering the filename. But that's impossible under normal circumstances—you can't type a CATALOG command while the computer's waiting for a response to INPUT.

"ESCape Key Upgrade," listed below, lets you add a special escape key function to display the catalog automatically whenever you press ESCape. With a little extra programming, you can use the same routine to give special meaning to any other key or keypress sequence on your Apple II computer.

Hit ESC For A CATALOG

Type in and save Program 1, then run it for a demonstration. The first ten lines install and activate a short machine language routine; line 110 performs an ordinary INPUT command. When the INPUT prompt appears, press ESC. The computer immediately displays a disk catalog, just as if you'd stepped outside program mode for a moment and typed CATALOG in immediate mode. Of course, you're not limited to a catalog display; by substituting different commands after the THEN statement in line 120, you can perform other commands, branch to a subroutine, or do whatever else you like.

The ESC key is ideally suited for this purpose, since it's located near the upper-left corner of the keyboard and is rarely used except for editing. Program 1 checks ESC after an INPUT statement, but this technique works just as well after any GET statement. If you want a program to respond to ESC more quickly, you needn't wait for an INPUT or GET. Try the technique known as *polling* (periodically testing) the keyboard. For instance, your program might call the following subroutine at regular intervals:

```
900 IF PEEK (-16384) < 128 TH
    EN RETURN: REM RETURN IF
    NO KEY WAS PRESSED
910 POKE -16368,0
920 IF PEEK (-16384) = 27 THE
    N PRINT CHR$(4) "CATALOG"
930 RETURN
```

The first line of this routine checks whether a key has been pressed since the last keyboard check was made. Only if the answer is yes does the routine go on to test for the ESC key specifically (this prevents the computer from reacting twice to the same keypress). With this routine in place, you can check ESC whenever you like by inserting GOSUB 900 at the appropriate spot in a program.

Checking Other Keys

Testing for keys other than ESC is quite easy. Take a look at line 60 of Program 1. The second value in the DATA statement represents CHR$(27), the ASCII value of the ESC key. To test for a different key, replace the 27 in line 60 with the ASCII code of the desired key, then make the same substitution in line 120 as well. For example, to test for the letter A instead of ESC, substitute a 65 for 27 in lines 60 and 120. If you rerun the program and press A at the INPUT prompt, the computer displays a disk catalog exactly as before.

What about reprogramming more than one key at a time? Let's check for three special key combinations: The ESC key will display the catalog for drive 1, CTRL-A will catalog drive 2, and CTRL-C will home the cursor. The first step is to add these lines to Program 1 (note that line 120 is replaced with a new line):

```
58 DATA 201,1,240,10 : REM CO
   NTROL A
59 DATA 201,3,240,6 :REM CONT
   ROL C
120 IF PEEK (-16384) = 27 THE
    N PRINT CHR$(4) "CATALOG,
    D1": GOTO 110
130 IF PEEK (-16384) = 1 THEN
     PRINT CHR$(4) "CATALOG,D
    2": GOTO 110
140 IF PEEK (-16384) = 3 THEN
     HOME: GOTO 110
```

The second number in each DATA statement represents the ASCII code for a particular keypress. The last number in each DATA statement must equal the last num-

www.commodore.ca

ber in the following line plus 4; since line 59 ends with the value 6, line 58 must end with the value 10 (6+4), and so forth. Lines 130 and 140 add the IF statements needed to test for CTRL-A and CTRL-C. Following this pattern, you can add your own key definitions.

Reprogrammed keys behave just like the RETURN key, causing INPUT to respond immediately. However, there are some minor restrictions. First, in a multiple INPUT command such as INPUT A$,B$,C$, the computer detects reprogrammed keys only after the last entry (C$ in this case). Second, if the INPUT command expects numeric input, you must always enter some number before typing the special key.

One disadvantage of checking for ESC is that normal ESC editing functions are disabled. The normal keyscan can be restored by pressing CTRL-RESET, but it's better form for your program to leave the computer in its standard state when it ends. To accomplish this, add the appropriate line below to your program at a point just before the end:

For DOS 3.2 or 3.3:
POKE 56,27:POKE 57,253:CALL 1002

For ProDOS:
PRINT CHR$(4);"IN#0"

A Short Diversion

Though you can use this routine without knowing how it works, machine language programmers may appreciate a closer examination. In simple terms, it diverts the computer from its normal keyscan routine to a new one that we install in RAM. To understand the details of this process, you'll need to know something about how the Apple's keyboard is normally scanned.

The Apple reads its keyboard with a routine known as RDCHAR (located at $FD35). RDCHAR in turn calls another routine known as RDKEY which retrieves an ASCII code from the keyboard and stores it in the microprocessor's accumulator. Then the computer adds 128 to the key code and stores the resulting value in the *keyboard data byte* (location $C000, which is expressed in decimal integer form as −16384). You can subtract 128 from the keyboard data byte with a GET, an INPUT, or by PEEKs or POKEs to the keyboard *status* byte; Apple calls this location ($C010 or decimal −16386) the keyboard *strobe*. After RDKEY is finished, RDCHAR checks to see if the key pressed was the ESC key. If so, it branches to a special routine. The ESC keypress itself is always ignored except as a signal indicating that the next keypress should be processed specially—as an escape code rather than a normal keypress.

Instead of obtaining the keycode on its own, RDKEY jumps to yet another routine, KEYIN, which actually does the work. KEYIN ordinarily begins at location $FD1B. This final bit of code is not called as a subroutine; instead, the computer performs an indirect jump instruction, passing control to the address stored in zero page locations 56–57 ($38–$39). Apple documentation refers to this pair of locations as the *keyboard input switch*. The important points to remember are that the keyboard data byte (−16384) can be PEEKed from BASIC, and that you can divert the computer to a different keyscan routine by changing two bytes in zero page.

Replacing the ROM version of KEYIN with a customized version is straightforward. Lines 10–70 of Program 1 POKE the new routine into memory beginning at address 768. Line 80 checks a memory location in the ProDOS global page, which begins at address 48640. If this location contains the value 76, ProDOS is active, so the standard input routine is changed using the command IN#A768, which tells ProDOS to get its input from the routine beginning at address 768. (That routine begins with a byte of 216, the CLD opcode, to let ProDOS know that this is a valid machine language routine.)

Lines 90 and 100 perform the input redirection for DOS 3.2 and 3.3. First, the high and low bytes of the routine's starting address ($300, or decimal 768) are POKEd into locations 56 and 57. (Since this routine is relocatable, you could put it any other place that isn't used by your main program.) Then the CALL 1002 in line 100 tells DOS to use the new KEYIN routine in place of the usual code. If you're interested, Program 2 contains the source code for this routine.

The first instruction in Program 2 is necessary for ProDOS. The next eight lines simply mimic the normal KEYIN routine. These instructions continuously generate a random number by incrementing locations $4E–$4F until a key is pressed. When you press a key, the computer moves the keycode found in the keyboard data byte into the accumulator, then clears the keyboard status byte. At that point, the normal version of KEYIN would return to RDCHAR, which would check the keycode to see if it is ESC. The custom KEYIN routine does the check right away. If ESC was pressed, we replace the accumulator's contents with the ASCII code for a carriage return (with the high bit set).

Program 1: ESCape Key Upgrade

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
AC 10 J = 768
2F 20 READ A: IF A < > 256 THEN
      POKE J,A:J = J + 1: GOTO 2
      0
31 30 DATA 216,230,78,208,2,230,
      79,44,0
1E 40 DATA 192,16,245,145,40,173
      ,0,192
8E 50 DATA 44,16,192,72,41,127
A7 60 DATA 201,27,240,2 : REM ES
      CAPE KEY
66 70 DATA 104,96,104,169,141,96
      ,256
EA 80 IF PEEK (190 * 256) = 76 T
      HEN PRINT CHR$ (4);"IN#A76
      8": GOTO 110
3D 90 POKE 56,0: POKE 57,3
92 100 CALL 1002
2E 110 INPUT "ENTER CHOICE ";A$
CC 120 IF PEEK ( - 16384) = 27 T
       HEN PRINT CHR$ (4)"CATALO
       G": GOTO 110
```

Program 2: New KEYIN Routine

This source code listing is for illustrative purposes only. It requires an assembler to be typed in.

```
        CLD           ; Valid ML routine.
START   INC $4E
        BNE NEXT
        INC $4F
NEXT    BIT $C000
        BPL START
        STA ($28),Y
        LDA $C000
        BIT $C010
        PHA           ; Save key code
        AND #$7F      ; Set high bit to zero
        CMP #$1B      ; Is it ESC?
        BEQ RETURN; If yes, branch.
        PLA           ; Not ESC, so replace
        RTS           ; original code, exit.
RETURN  PLA           ; Tidy up stack.
        LDA #$8D      ; Carriage return.
        RTS
```

©

www.commodore.ca

INSIGHT: ST

Bill Wilkinson

Exploring The ST

Hi—welcome to "Insight: ST," COMPUTE!'s new monthly column for the Atari ST-series computers. Over the coming months, we're going to help you become more familiar with the ins and outs of the Atari ST, its operating system, GEM, and ST BASIC. The ST series is the most powerful line of computers ever released by Atari—one of the most powerful in the industry, in fact—so there's a great deal to learn and explore.

Before getting started, I want to reassure those of you who still own and use the eight-bit Atari 400/800, XL, and XE computers. Veteran readers will recognize that I've been writing the "Insight: Atari" column on these machines for the past five years in COMPUTE!. This new column *does not* mean that we're dropping Insight: Atari. In fact, I plan to continue writing Insight: Atari in addition to Insight: ST for the next few months. Eventually I'll turn over Insight: ST to someone more specialized in ST BASIC. Don't be surprised, though, to find occasional ST tidbits in Insight: Atari as well. Both columns will be of continuing interest to all Atari enthusiasts.

The ST In Perspective

Just what *is* an Atari ST computer? Even if you already own an ST, I may have some surprising answers for you.

From a hardware viewpoint, the ST is most commonly compared to the Apple Macintosh and the Commodore Amiga. Indeed, it shares characteristics with both. All three use a Motorola 68000-series microprocessor, 3½-inch floppy disk drives, bit-mapped screen display, a range of peripheral interfaces, and more—generally the things we've come to expect from today's advanced personal computers. Both the ST and the Amiga have one advantage over the Macintosh: color graphics (though admittedly only the Amiga uses a sophisticated graphics processor chip to display true sprites).

Even the user interfaces of the three machines are similar: All have a system of icons, multiple screen windows, and a mouse controller to visually display and manipulate the contents of disks and perform general "maintenance" chores.

Finally, as long as we're making comparisons, we should be fair and mention that the Macintosh has much, much more software available for it than either of its competitors. But that situation is changing rapidly, even as I write.

What makes the ST stand out from other computers? Well, the Atari marketing department has a whole series of answers, but let me tell you the ones which impress me. First and foremost is the built-in hard disk port. It's capable of transferring data to or from a hard disk (or a network or external RAM disk or whatever) at a rate of up to *1,300,000 bytes per second*. That means you could, in theory, fill the 512K RAM memory of a 520ST in under half a second.

The Allure Of Speed

Theory is nice, but what does this mean in practice? Well, for me (or any other programmer) it means that after writing source code with a text editor, I can save the source file to disk, exit the editor, load a compiler, compile the source code I just saved, link the resulting object code with both system and GEM libraries, and maybe (if I swallow fast) finish eating a bite or two of a sandwich. Elapsed time: between 10 and 15 seconds, depending on the size of the program. On an IBM PC with a hard disk, it would take four to six times as long. And on the Macintosh, most external hard disks aren't much faster than the ST's floppies.

Thanks to its fast processor and amazing hard disk speed, for sheer computing power there is probably no "home" computer available which can touch the ST. Exception: If you're doing heavy work with floating-point math (for example, scientific or engineering computing), an IBM PC with 8087 floating-point chip will win hands down. (Are you listening, Atari?)

The only other hardware features which are distinctively Atari are the built-in MIDI (Musical Instrument Digital Interface) ports, the cartridge slot, and the absolutely beautiful *black-and-white* display. Now MIDI will be of interest to musicians, and the cartridge port may have some interesting future applications (perhaps a way to get that fast floating-point chip?), but the surprise here is the 640 × 400 monochrome display. Why do I rave about this on a machine with advanced color graphics?

Although I enjoy color displays, I will probably never create one. I'm not particularly artistic and I don't write games. But I *do* write programs. Which means I appreciate an easy to read, rock-steady display. Atari went to the trouble to equip the ST with a completely separate monochrome video port, and its quality is nothing short of amazing. And, besides, it costs $200 less than a color system. (But be forewarned: Many games only run on a color monitor. Poor software design, in my opinion, but that's how it is.)

Next month, we'll begin turning the Atari ST inside out and exploring the intricacies of TOS, its multilevel operating system. ©

www.commodore.ca

AmigaView

Charles Brannon

New Amiga Software

Welcome to the premiere of COMPUTE!'s Amiga column. For the last two years, I've been writing "Horizons," a Commodore column in COMPUTE!'s GAZETTE, and I'm very excited to be writing a new column for the Amiga. While other publications concentrate primarily on Amiga reviews and using your Amiga with commercial software, this column will have a strong bent toward programming techniques, tricks, and tips.

Nonprogrammers will find information on using the operating system (both the Workbench and AmigaDOS) to its fullest. We'll also pass along late-breaking Amiga developments.

Amiga Steals The Show

Although Commodore didn't exhibit at the recent Comdex or Winter Consumer Electronics Show, the Amiga did make a big appearance in early February at The Commodore Show II in San Francisco. This show was hosted by the West Coast Commodore Association, one of the largest Commodore user groups. Thousands of enthusiastic Commodore owners turned out to look at the latest offerings of software and hardware companies. In fact, the show was so popular that the fire marshal finally had to lock the doors to prevent the crowd from exceeding the building's occupancy limit.

About half the show was dedicated to the Amiga, and the Amiga was clearly the show-stealer. Commodore sponsored a large booth which was lent to developers for product demonstrations. This was the first time many Commodore owners got a chance to see the Amiga working with actual software instead of the same old demo programs, so there was quite a bit of interest. And more than a few people at the show were Amiga owners themselves, scouring the floor for everything they could find.

They didn't go home empty-handed. Lots of companies were displaying their latest Amiga wares. (Keep in mind that I'm writing this column in mid-February, so check with your dealer on the availability of promised titles.)

Activision wowed showgoers with *The Music Studio,* a complete music composition system. Using Commodore's MIDI (Musical Instrument Digital Interface) adapter, the Amiga and *The Music Studio* were playing a complex nine-part harmony on a keyboard synthesizer. Besides MIDI compatibility, other features of *The Music Studio* are a sound editor/synthesizer, standard music notation, beginner's music notation, a sound library, up to 15 instruments per song, full editing capability, a staff that scrolls during play, and the ability to print out lyrics along with the sheet music. Many of these options are accessible from pull-down menus. The suggested retail price is $59.95.

From C To Shining C

Aztec C, by Manx Software (Eaton Town, New Jersey), is a new alternative to *Lattice C* which supposedly compiles faster and generates smaller, faster-running code than the Lattice compiler. With its built-in libraries, *Aztec C* is compatible with all Amiga function calls. A new translation utility lets you translate existing Lattice source code to *Aztec C. Aztec C* is available in several levels, from a simple bare-bones C system for beginners who want to learn C to the full-power Commercial System designed for developers writing commercial software.

A couple of companies were taking orders for their memory expansion boards. Skyles Electric Works (Mountain View, California) sells a 256K module that plugs into the front memory port for $149.95. Comspec Communications (Toronto, Ontario) sells a 256K board for $130 and offers two megabytes (2,048K) for $1,196. StarPoint Software (Yreka, California) sells a 256K board for $120, and it even comes with the schematics.

Two companies are introducing touch tablets for the Amiga. These flat pads let you draw with a pen to transmit screen coordinates to the Amiga. You can use the pad as a substitute for the mouse with graphics software for more natural freehand drawing and effortless tracing. Kurta Corp. (Phoenix, Arizona) sells a full line of digitizer pads, from the $375 *Penmouse+* to a drafting-table size professional system. Software is included to interface the tablet with the Amiga operating system. Anakin Research (Rexdale, Ontario) has a high-resolution tablet called the *EASYL* with software that works in all Amiga screen modes. In 640 × 400, it's a lot like drawing with a pen on a white piece of paper. It also works with Electronic Art's *Deluxe Paint.* It sells for around $500.

Micro-Systems Software (Boca Raton, Florida) is selling *Analyze!,* a spreadsheet ($99.95), *Online!,* an already-popular telecommunications program ($69.95), and *BBS-PC,* a bulletin board system that lets anyone with an auto-answer modem set up their own BBS ($99.95). All three packages are available now.

I've seen a list of over 100 Amiga products that are promised by the first quarter of 1986. Even more projects are under development and "soon to come." As more machines sell, and as more products jell, the Amiga will escape this temporary shortage of software and expansion hardware. Every machine has had to run this gauntlet, but the Amiga may emerge as a particularly strong runner.

www.commodore.ca

Arlan R. Levitan

Online Etiquette

Manners are "in" this year. Since I can't turn on a television or radio without running into a self-styled expert on genteel behavior, I figure I might as well get into the act with my patented "Proper Pointers for Bulletin Board Paparazzi."

When logging onto a BBS, remember that you are a guest in the system operator's (sysop's) electronic house. Sysops have the right to lay whatever ground rules they see fit for their personal domains. If you have any problems with the house rules, feel free to register a mild-mannered and polite complaint with the management, but be prepared to find another game in town more to your liking if the house does not relent.

Give your real name when asked to log in. Using the name of heavy metal groups, five-letter expletives, or such hackneyed titles as "Hacker," "Cracker," or "Whacker" are considered passé and a harbinger of the imminent demise of the rest of your brain cells. If the bulletin board you are accessing encourages the use of *noms-de-plume,* stick with an obscure or bizarre moniker. My favorites are the names of ex-postal ministers of Liechtenstein.

Don't be a data glutton. Downloading every new file that shows up on a BBS may keep you off the streets, but ties up many boards an inordinate amount of time. Logging off and back on with a different name after your time limit has expired is tantamount to hogging the shower until all of the hot water is gone. Show your discrimination and taste by carefully examining the descriptions of files that are available for downloading, and choose only those that are of real interest or utility to you.

Anyone Need A Ginsu?

If there is a message section on the BBS, make an effort to read the latest messages and participate in the flow of conversation. The content of messages you leave should be consistent with any statement of direction that the sysop has established. The right of free speech notwithstanding, leaving a message offering to trade a set of Ginsu knives for a Veg-A-Matic may be considered a breach of protocol on a board dedicated to discussions of artificial intelligence and the search for UFOs.

Try to instill some degree of content into every message you leave. Politeness is a virtue, but responding to every bit of assistance with the single word "Thanks" wastes pointer space within the BBS indexes and often leaves other folks wondering just what you were thanking someone for when the original message rolls off the message base.

Leaving public messages that give away solutions to particularly difficult problems in the latest adventure game is somewhat less sporting than standing in front of the audience at a movie theater and announcing the conclusion before the film begins. If you must request help with such programs, solicit assistance in the form of a phone call or letter.

Obscenity is not only "out," it's boring. Ask anyone who's read the unexpurgated Watergate transcripts. Questioning the lineage of the sysop or attempting to crash the board because you don't like a particular policy or rule is a waste of your time and as welcome as an IRS audit.

Sibling Rivalry

Perhaps the most difficult thing for many people is abstaining from the various "My computer is bigger/better/faster than yours" message threads. These discussions usually exhibit all the charm of a dozen or so bull elks ramming their heads together during the rutting season.

The most recent examples are the dozens of Amiga versus Atari ST shouting matches that have been jamming both public bulletin boards and commercial information services all over the country. For the most part, the opposing sides are made up of loyal Atari and Commodore followers who are anxious to defend the honor of their long-time corporate allegiances—holdovers from the Atari 800 versus Commodore 64 debates. What makes all of these arguments and insults somewhat ludicrous is that the Amiga's custom chips are largely the work of Jay Miner, who designed the graphics chips in the Atari 2600 and eight-bit Atari computers. So the Amiga is actually a mid-1980s Atari, while the Atari ST computers—the product of Jack Tramiel & Sons—are really mid-1980s Commodores.

If all the energy that has been spent on these types of discussions were channeled into more productive pursuits, I have no doubt that many of the real mysteries of the universe could have been solved—including where all of the jackets for my floppy disks keep vanishing to.

Finally, and perhaps most important, is to cultivate an attitude of gentle tolerance toward those who insist on pontificating and regaling you with their personal vision of the "right" way to do things. Having made it through this list of dos and don'ts, you're well on your way to that end. ©

www.commodore.ca

The World Inside the Computer

Fred D'Ignazio, Associate Editor

Training For Tomorrow's Jobs: High-Tech Skills And Beyond

The reindustrialization of the United States is underway. The U.S. economy is rapidly modernizing its smokestack industries and converting them to futuristic, high-tech companies. High-tech tools—including robots, computers, office automation, and telecommunications—are steadily becoming part of the daily working environment in most U.S. companies. According to a recent issue of the *Kiplinger Washington Letter*, "Even old-line manufacturers are becoming high-tech companies, using computers to orchestrate what goes on where and when in their plants. Formerly machines replaced brawn. Now they replace brains or skilled help."

For instance, in Rochester, Michigan, just east of Pontiac, older companies are swiftly going high-tech. Electronic Data Systems, recently acquired by General Motors, is transferring operations to Rochester. Chrysler Labs is expanding its operations. And a multimillion-dollar high-tech industrial park is under construction.

That means new jobs are opening up in Rochester. But are Rochester's schoolchildren receiving the training they'll need for these jobs?

High-Tech Thinking

Rochester already has more computers in its schools, per capita, than any other city in the country. However, according to Dr. Anne Porter Jaworski, a teacher at Rochester's Oakland University, "Training on computers and other high-tech machines will not guarantee future jobs for our children. We must also teach our children higher-level thinking and communications skills. By the time they become adults, all other skills will be automated, and the jobs will be done by machines."

Jaworski's Canadian colleague, Craig Stirton, a former schoolteacher, agrees: "I've talked with executives from several large companies. They are seriously worried about the kinds of skills young people have when they leave school. The companies have been forced to supplement their new employees' formal schooling with creative-thinking and communication-skill seminars. I think it's time children began learning these skills as part of their basic education."

In Tennessee, General Motors is building what may become the largest high-tech industrial complex in the world. Known as the Saturn Project (after the Saturn, GM's car of the future), it will give a major boost to the Tennessee economy and employ thousands of workers. Tennessee hopes that the Saturn workforce will be drawn from people already living in Tennessee. But according to William D. Hoglund, Saturn's president, applicants must pass a rigid test. They must be competent in language and math, in computer and other high-tech skills, and they must be "risk takers" and demonstrate "a commitment to creative change and growth, and lifelong learning."

Anticipating GM's and other corporations' new human resource needs, Tennessee Governor Lamar Alexander initiated a statewide Better Schools program in 1984, funded by a one-cent sales tax. Through this program, Alexander and the state board of education are revamping the curriculum in Tennessee's schools. Children who graduate from high school in coming years will have a solid foundation in ten different areas, including the basics (reading, writing, arithmetic), computers, and other job-related skills, such as accounting, telecommunications, and engineering.

Dr. James Kelley of Tennessee's Department of Education is confident that public schools can provide the training young people need to meet Saturn's challenge. Stirton and Jaworski agree. "In Tennessee," Kelley says, "we are putting renewed emphasis on high-level thinking skills as part of present subject areas like math and reading."

Engrossed In Learning

In Michigan, Stirton and Jaworski have designed a learning program, called a "transformational environment," that would focus on these high-level skills beginning with elementary school pupils. According to Stirton, "We want a place where children would be so engrossed in their activities that they would become oblivious to everything around them."

"This kind of peak experience is common to hobbies," adds Jaworski, "but I don't see why it couldn't happen in school." It could be part of an activity center in a classroom or a special class during the day or after school, she explains. "The energy released would be tremendous. Children would feel exhilarated by doing something they defined as a goal for themselves. Many teachers are already doing this. We should all be communicating together, and we need support from parents."

To get in touch with Stirton and Jaworski, write: Craig Stirton, 6275 Atherly Crescent, Mississauga, Ontario, Canada L5N 2J1 (his CompuServe I.D. is 72777,1054); or Dr. Anne Porter Jaworski, School of Human and Educational Services, Oakland University, Rochester, MI 48063. To learn more about Tennessee's Better Schools program, write Dr. James Kelley, Assistant Commissioner for General Education, Tennessee Department of Education, Suite 200, Cordell Hull Building, Nashville, TN 37219.

www.commodore.ca