

PROOF OF PURCHASE

Software Giveaway with Purchase of this Magazine

See back cover

HOME COMPUTER™ magazine

FOCUSING EXCLUSIVELY ON ● APPLE ● COMMODORE ● IBM ● TEXAS INSTRUMENTS

Vol. 5 No. 3

\$3.50 in USA
\$4.50 in Canada

Computerized Budgeting

—A Mighty Ally Helps Slice The Pie

Featuring A Ready-To-Use Budget Processor for:

- Visualizing Spending Trends
- Monitoring Expenses
- Planning Purchases
- Boosting Savings

Special Apple, Commodore, IBM, & TI Software:

- Honing Geometry Skills

Contains
33
Type-In
Programs!



—Reviews Galore:

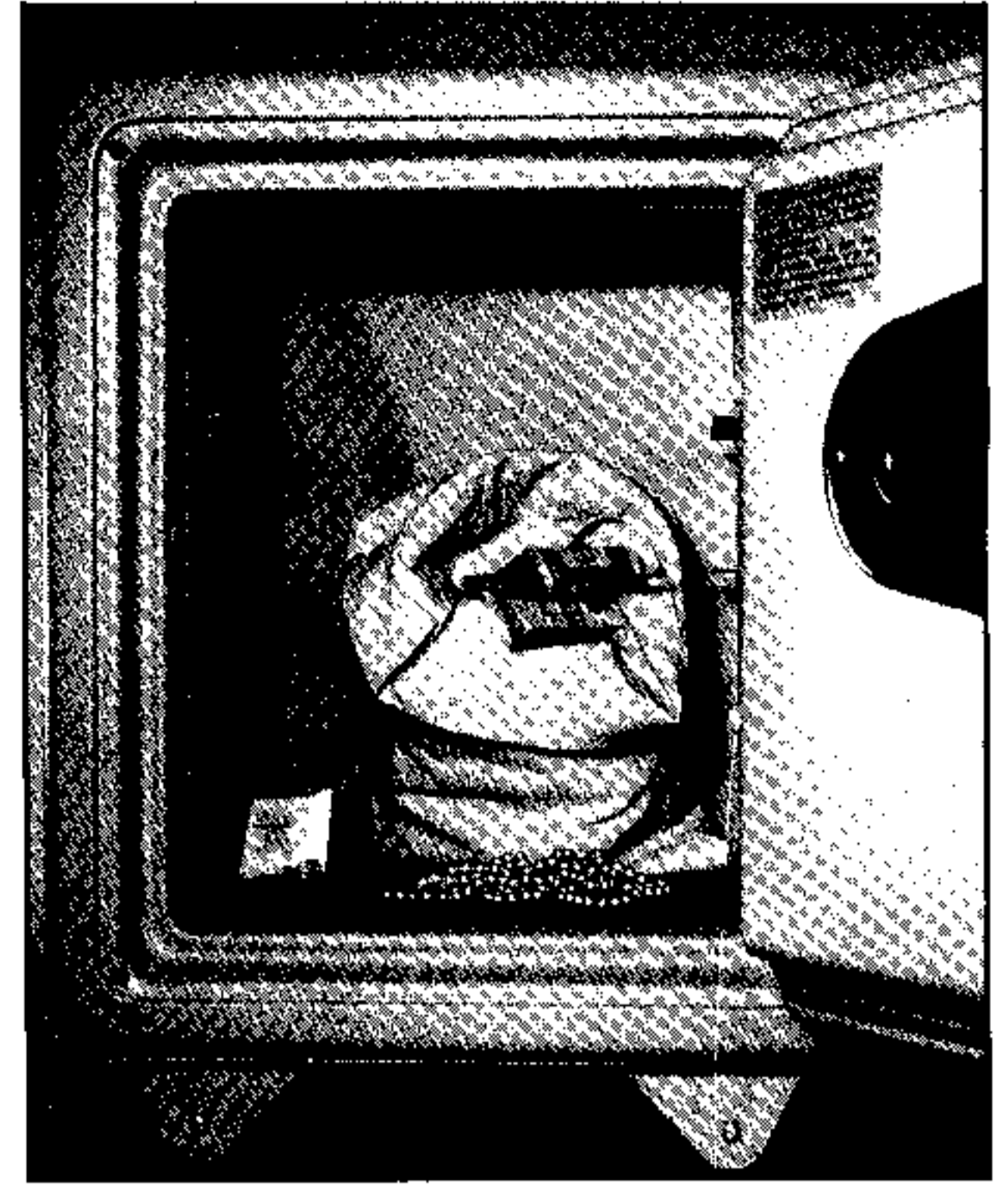
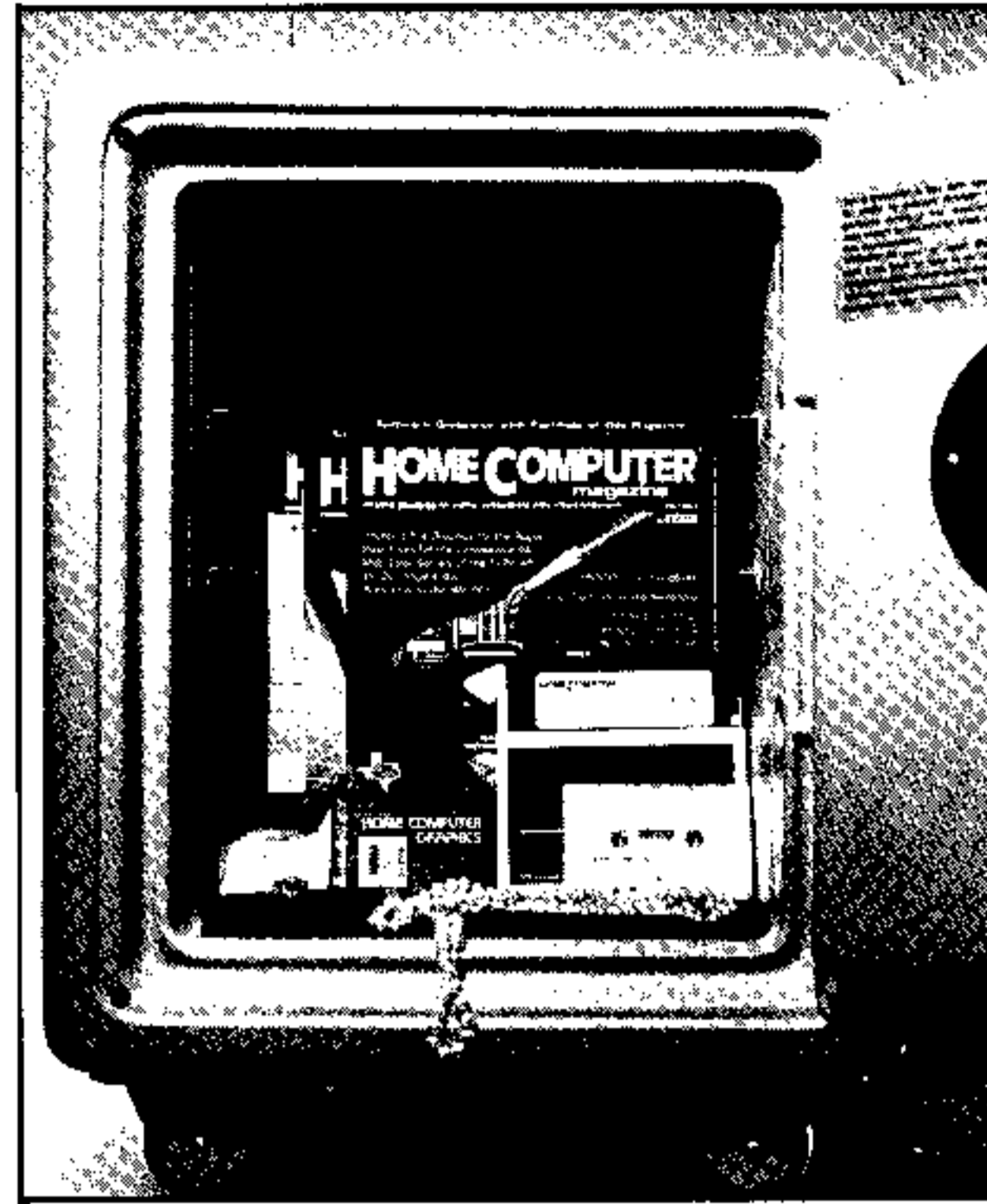
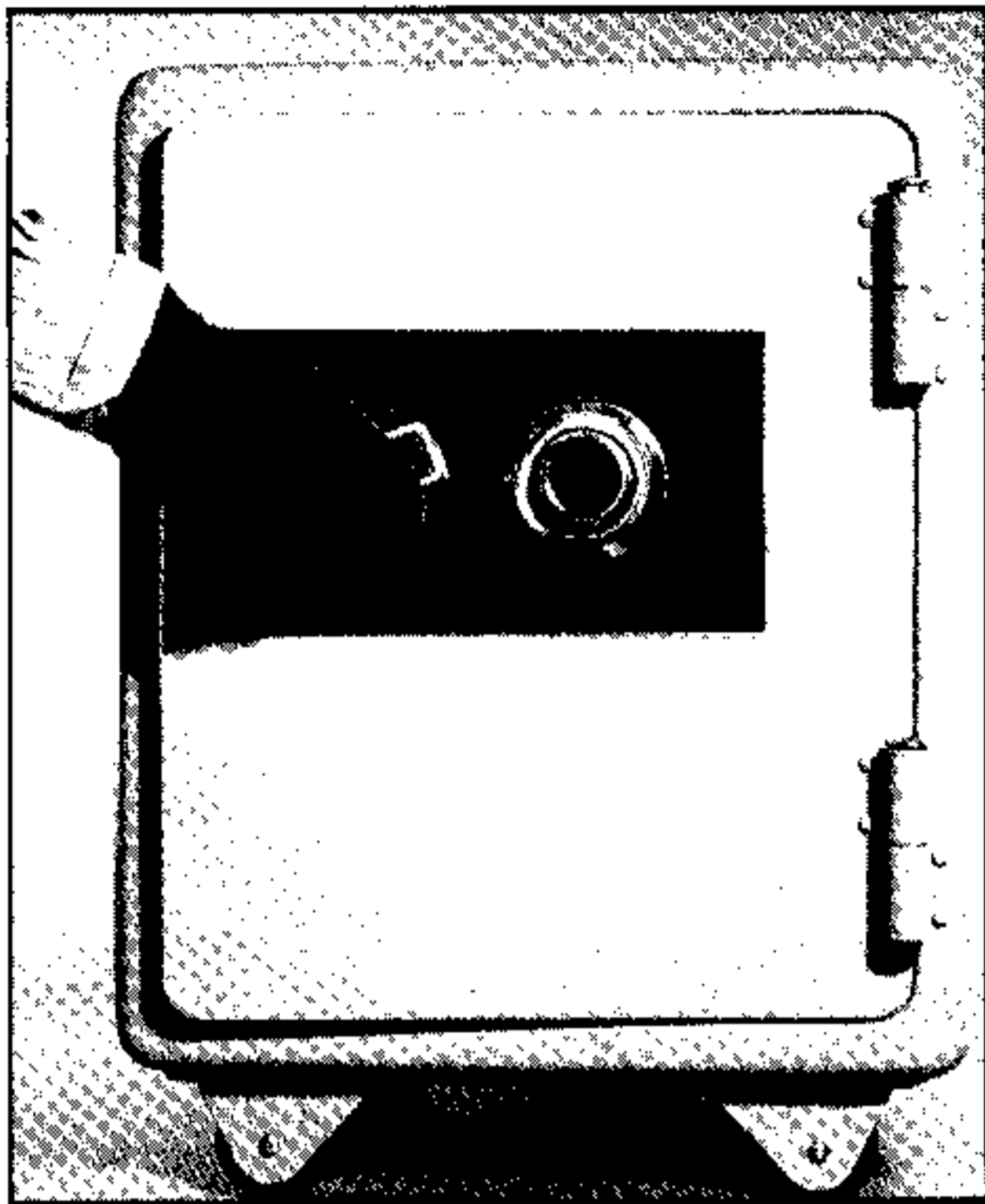
- ★ Artist & Programmer Tools
- ★ Typing Tutors & Word Processing
- ★ Computerized Speech & Scrabble
- ★ Music & MIDI Link-Up to Home Computing

45806COURTNE V05N10 H11135070
JAMES COURTNEY
RT 4 2744 B HUME RD
CRIDERSVILLE OH 45806

03



"477465"



MISSING ANY VALUABLES?

If you're missing any back issues of **HOME COMPUTER** magazine™ you're missing more than you'll ever know . . . ▶

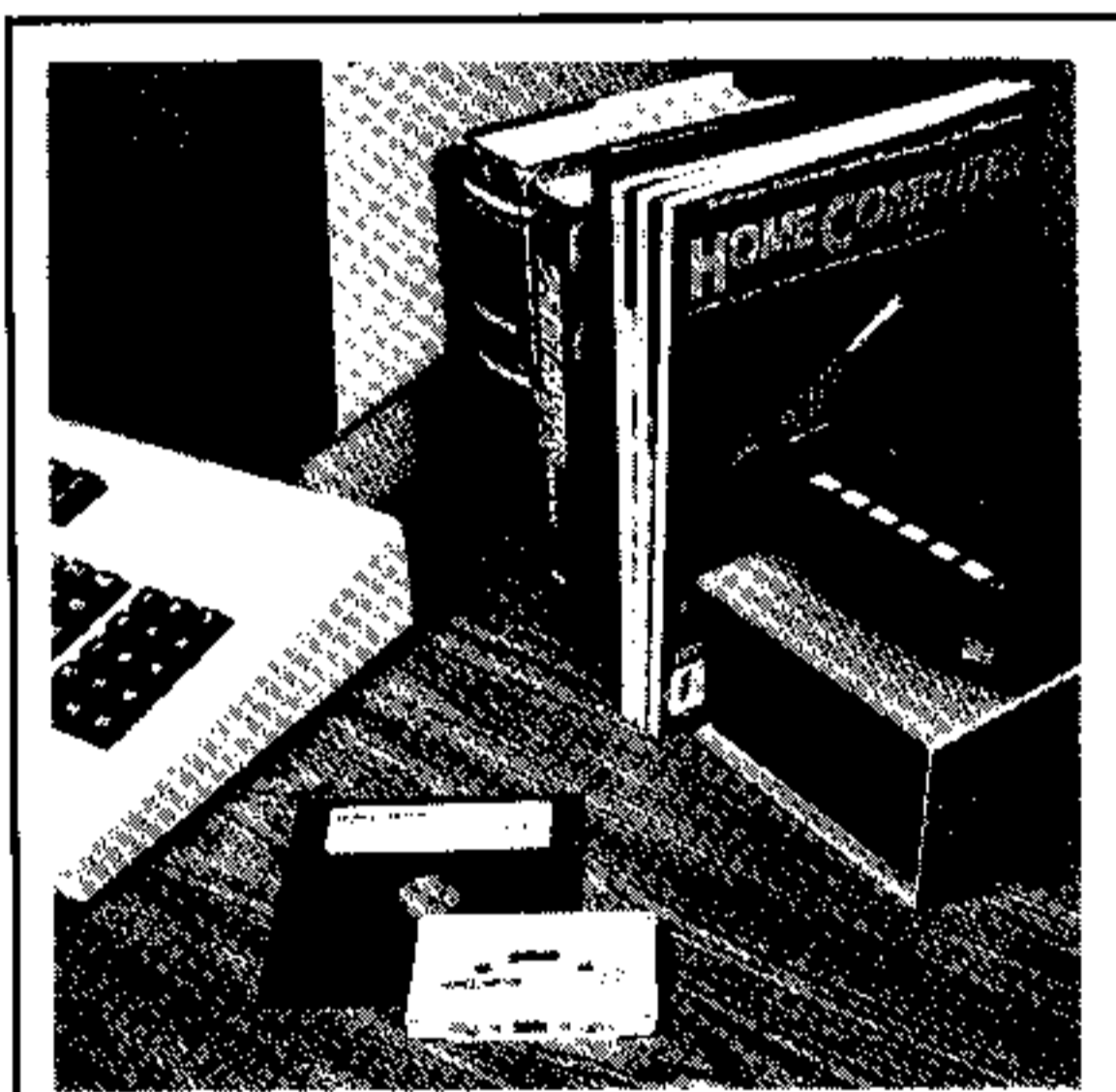
Having each issue of *Home Computer Magazine* readily at hand provides you with direct access

to a valuable reference library of home computer knowledge —unequaled anywhere!

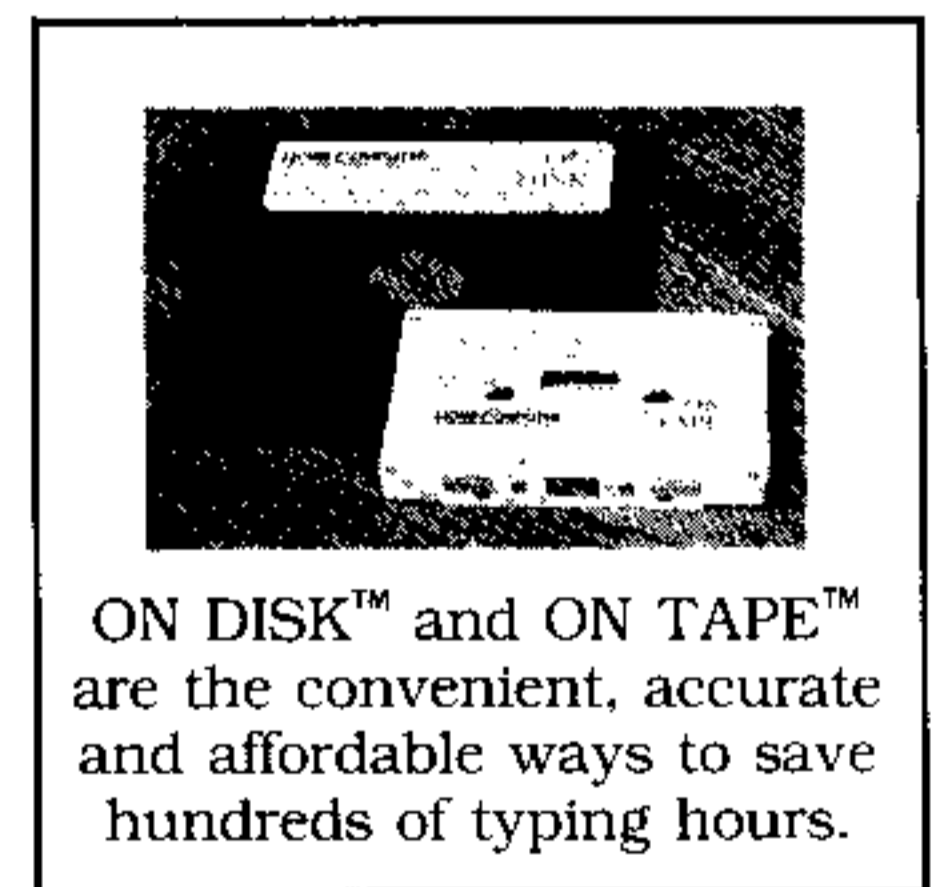
Back issues of HCM's program service—**ON DISK™** or **ON TAPE™** are also available.

Collect all the programs from each magazine issue on a ready-to-RUN quality floppy disk or cassette tape available in

separate versions for Apple, Commodore, IBM, and Texas Instruments home computers.



A valuable reference library of each *Home Computer Magazine* issue is the One Essential Peripheral™ for your home computer.



ON DISK™ and **ON TAPE™** are the convenient, accurate and affordable ways to save hundreds of typing hours.

**“Safeguard” Your Home Computer Knowledge—
Order Valuable Back Issues Today!**

To Order, Use Bind-In Card at Center of Magazine.

... and discover what's in store for you— an extraordinary resource value!

VERSIONS SUPPORTED:

Machine	Media
APPLE II Family (A)	ON DISK™
COMMODORE 64 (C)	ON DISK™/ON TAPE™
IBM PC/PCjr (I)	ON DISK™
TI-99/4A (T)	ON DISK™/ON TAPE™

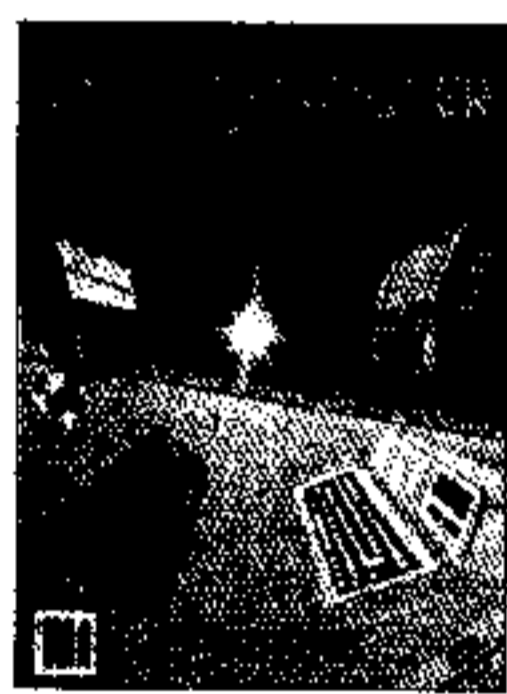
* = No ON TAPE™ available, even if normally supported
TX = Extended BASIC programs only
PCjr = Available for PCjr only

Apple Owners: Please note that ON DISK™ Media for HCM 4.1-4.3 is in DOS 3.3 format only, and all Apple programs beginning with HCM 4.4 are in ProDOS format. All programs will RUN on a 64K Apple II+ (with Applesoft BASIC in ROM), an Apple IIe, or an Apple IIc.

Apple & IBM "clone" Owners: Please note that some HCM programs may not RUN on your machines, because of differences in hardware and/or BASIC interpreters.

Issue 4.1:

Premier Issue * Uncle Larry's Fiddle Tunes * Electronic Sheet Music * Music in Mini Memory * PCjr: A Look Inside the Peanut's Shell * 66 Keys to Graphics Success: A Primer for the Commodore 64 * Have No Fear: Assembly Language Won't Byte, Part 3 * Porsches and other Pipedreams: Computer Assisted Savings * 3Dile: Apple Graphics in Three Dimensions, Part 1 * Biting Into Your Apple * Don't Be A SlowPOKE * Down Memory Lane: Don't let programmable characters gobble up your memory * Easy As Pie: Apple programming for intricate works of art * Microcomputer Accuracy * What is LOGO? * Lyrical LOGO * LOGO Shoots for the



Moon: A lesson in structured problem-solving * Product Reviews * Flak Attack * Slots * Meltdown * Challenging the Tower of Hanoi * HCM TECH NOTES: Apple, C-64, IBM, and 99/4A * Product News * Group Grapevine, and much, much more!

CONTENTS: ON TAPE™ & ON DISK™

Flak Attack (A,C,I,T)	Slots (T)
Air-to-ground battle game	An intriguing Las Vegas simulation
Applesoft 3D (A)	Uncle Larry's Fiddle Tunes (C, I, T)
Apple graphics in three dimensions	Play ten beloved fiddle tunes
Tower of Hanoi (A,C,I,T)	Music Magic (TX)
An ancient brain teaser	"Joy to the World" in harmonious BASIC
Saving (A,C,I)	Music Assembler (T)
Computer-assisted savings plan	Assembly language simplifies composition
LOGO Poet (A,C*, I)	Autosprite (C)
Recursion frees the poet in your console	Routines to keep your graphics lively
LOGO Apollo (C*, T)	Meltdown (TX)
A lesson in structured problem-solving	Debug the reactor and save the world

Issue 4.2:

Graphics * Sea of States * San Francisco Tourist * Building Your Character: A Graphics Editor for the VIC-20 * Quick Pixel Tricks: A Graphics Editor for the C-64 * Follow the Bouncing Ball: On the rebound with graphics fundamentals * 3Dile: Apple Graphics in Three Dimensions, Part 2 * Double Your Color, Double Your Fun: Sprites try on a layered look * Musical Mystery Words * Matrix Muncher * Elementary Addition and Subtraction for the VIC-20 * IBM Animation: Controlling the pallet on the PCjr * Jr. Sounds Off: Access Jr's Special Sound Enhancements * The Electronic Home Secretary * Files in LOGO * LOGO Spans the Generation



Gap: A review of Commodore LOGO * FROGO: LOGO Invades the Arcade * Product Reviews * Tablut * Cannibals * HCM TECH NOTES: Apple, C-64, IBM, and 99/4A * Product News * Group Grapevine, and much, much more!

CONTENTS: ON TAPE™ & ON DISK™

Cannibals (A, C, I, T)	Matrix Muncher (C)
Livingston Stew, I presume?	Solve unknowns simultaneously
FROGO (T)	Graphic Editor (C)
A logical LOGO learning lesson	Pixel tricks create easeful graphics
The Home Secretary (A, C, I, T)	Mystery Words (A, I)
Address & inventory recordkeeping	Reading between the treble clefs
LOGOFILES (A, C*, I, T)	PCjr Animation (PCjr)
Access your DATA files in LOGO	Exploring Junior's graphic modes
Sea of States (C, TX)	Applesoft 3-D Ile (A)
State Capitals and dive for booty	Edit your 3-D graphic shapes
Tablut (C, I, TX)	
14th-century strategy revisited	

Issue 4.3:

Productivity * Snap-Calc: A Homespun ready-to-use spreadsheet * Bars and Plots: Create colorful graphic charts of your records * Elementary Addition and Subtraction for the 99/4A and C-64: A powerful children's learning tool * Spider Graphics: Spin a colorful web on screen * Convertible for Comfort: Automatically convert your machine-language programs to DATA statements * Programming: The Name of the Game: Designing your own game—a complete tutorial * Colorfun on your VIC-20 * Product Reviews * Binary Forest: Branching out with LOGO * LOGO Flakes: Creative explorations with snowflake designs * Robochase * Cyber-Cipher * Wild Kingdom * Speeder * Boolean Brain * Missile Math * HCM TECH NOTES: Apple, C-64, IBM, and 99/4A * Product News * Group Grapevine, and much, much more!



CONTENTS: ON TAPE™ & ON DISK™

Snap-Calc (A, C, I, TX)	Binary Forest (A, C*, I)
Home sweet spreadsheet	Branching-out with leafy LOGO trees
Robochase (A, C, I)	Bars & Plots (T)
Run from the rampaging robots	Color your chart factfully
Spider Graphics (A, I)	Cyber-Cipher (T)
Spin a myriad of rainbow filaments	Decode correct color combinations
Boolean Brain (A, I)	Elem. Addition & Subtraction (C, T)
A graphic Adventure inside computers	BASIC preschool arithmetic skill-builder
Wild Kingdom (A, C, I, TX)	LOGO Flakes (T)
Flee ferocious felines	Snowflakes in June? This must be LOGO
Missile Math (A, C, I, T)	Convertible for Comfort (C)
Launching grade-school arithmetic	Machine Language DATA auto-conversion

Issue 4.4:

Computer Sports * Ilc: The Core of a New Machine * On the Home Court: Computer Sports Simulation * Razzle Dazzle: Quick Graphics Magic for the 99/4A * Simon Sez: Plug in 114 new BASIC commands to the Commodore 64 * Tax Deduction Filer: A complete tax recordkeeping program convinces you that makes tracking of deductions a breeze * Kaleido Computer: Creating a myriad of mosaic designs on your home computer * Multiplan Medium, Part 8 * Have No Fear: Assembly Language Won't Byte, Part 4 * The RS-232 Interface: Understanding Your Link to the Periphery * One for the Money,

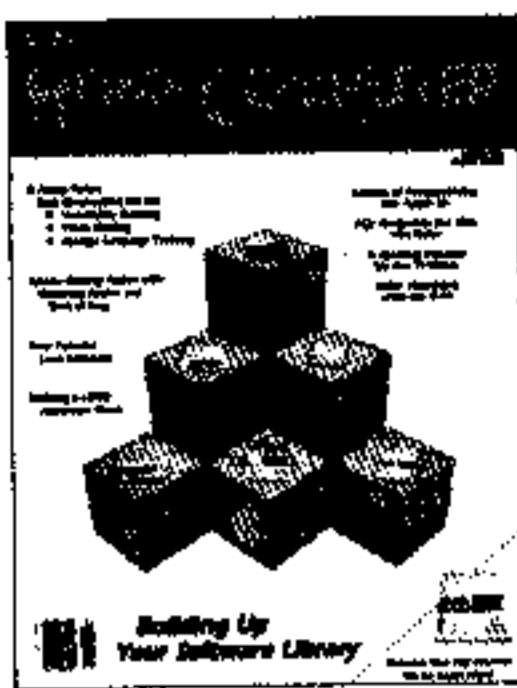


Two for the Slow—Adding a Second Drive to the PCjr * Missionary Impossible: A Logic Puzzle in LOGO pits you against hungry Cannibals * Product Reviews * Boolean Brain * Stadium Jumping * Market Madness * Elementary Addition and Subtraction: An arithmetic tutor (for Apple and IBM PC and PCjr systems) * HCM TECH NOTES: Apple, C-64, IBM and 99/4A * Product News * Group Grapevine, and much, much more!

CONTENTS: ON TAPE™ & ON DISK™

Boolean Brain (C, TX)	LOGO Spreadsheet (A, C*, I, T)
A graphic Adventure inside computers	And you thought LOGO was kidstuff
Tax Deduction Filer (A, C, I, TX)	Missionary Impossible (A, C*, I, T)
SAVE-ing with your tax deductions	Watch out for Cannibals with LOGO
Market Madness (A, C, I, TX)	Elem. Addition & Subtraction (A, I)
Exciting Stock market simulation	BASIC preschool arithmetic skill-builder
Stadium Jumping (A, C, I, T)	
Horsing around an Olympic Stadium	

And there's More On The Next Page



Issue 4.5:

Building Up Your Software Library * Quiz Construction Set: Create a Quiz or Take a Quiz—a must for students and teachers * Personal Loan Calculator: Find out where your interest lies * Jumping Ahead With Game Programming: A complete game programming tutorial includes a program example * Sketch-64: Joystick graphics with just a flick of the wrist * Simon Sez: New string-related commands explained * Razzle Dazzle: Character manipulation on the 99/4A * Division Tutor: Teaching BASIC math learning skills * Putting The Puzzle All Together: Apple IIc Programming Considerations * Bird Brain * Slither * LOGO Clones: TI Graphics In a Turtle-Shell * Build A LOGO Adventure, Part 1 * Product Reviews * HCM One Liners * HCM TECH NOTES: Apple, C-64, IBM, and 99/4A * Product News * Group Grapevine, and much, much more!

CONTENTS: ON TAPE™ & ON DISK™

Bird Brain (A, C, I, T)
Keep your fishing feathers dry
Division Tutor (A, C, I, TX)
Expand elementary math skills
Personal Loan Calc (A, C, I, T)
Find out where your interest lies
Sketch-64 (C)
Use a joystick to draw graphics
Quiz Construction Set:
Quiz-Make/Quiz-Take (A, C, I, T)
Complete tutorial with file examples

Peg Jump (A, C, I, T)
Learn BASIC game programming
Slither (A, C, I, T)
A maze of snake-like proportions
LOGO Clones (T)
TI-Graphics in a Turtle-Shell
LOGO Adventure (A, I, C*)
Pt. 1: Creating interactive fiction



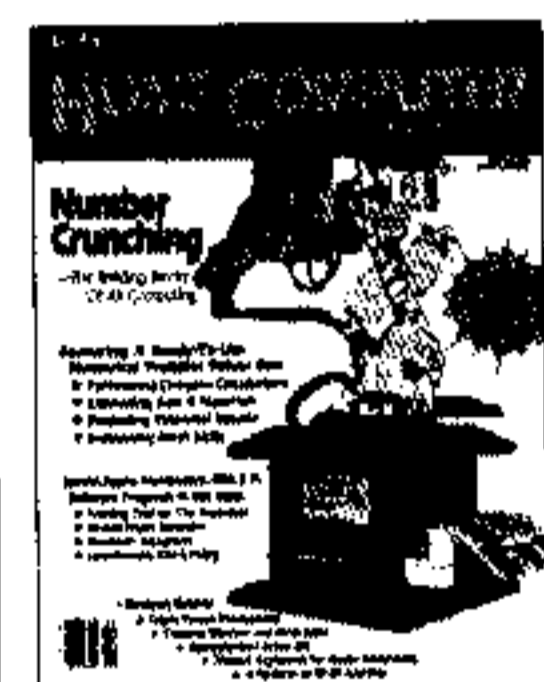
Issue 5.1:

Thought Processing: A New Frontier in Home Computing * The Organizer: Store and organize your thoughts * Orbital Defender * Quiz-Print/Quiz-Print Tutorial: This educational enhancement is a tool for use with your Quiz Construction Set (see HCM 4.5) * Electronic Backgammon: A modern version of an ancient game of skill * Razzle Dazzle: Screen patterns with graphics characters on the 99/4A * Kors-Elf: An Arcade Typing-Tutor Game * Personal Loan Calculator: Find out where your interest lies * Apple Seedlings: A ProDOS Date-Setting Utility * IBM Impressions: Create a beautiful pie chart * Build A LOGO Adventure, Part 2 * LOGO Sailing: A Premier Yachting Event * Simon Sez: Composing music is simple * HCM TECH NOTES: Apple, C-64, IBM, and 99/4A * Product News * Group Grapevine, and much, much more!

CONTENTS: ON TAPE™ & ON DISK™

Orbital Defender (A, C, I, T)
Split-second battle decisions
Electronic Backgammon (A, C, I, TX)
Pit your pips against the computer
Kors-Elf (A, C, I, TX)
An arcade typing adventure
The Organizer (A, C*, I, TX*)
A versatile Thought Processor
Quiz-Print (A, C, I, T)
Format printouts of your quizzes
Apple Seedling (A)
BASIC utility dates ProDOS files

LOGO Adventure (A, C*, I)
Pt. 2: Creating interactive fiction
Merging Files (C)
Experienced hackers only!
Personal Loan Calc (T)
Find out where your interest lies
Razzle Dazzle (T)
Wormwood your character graphics
LOGO Sailing (T)
Turtles race for the America's Cup
IBM Impressions (I)
Create a beautiful pie chart



Issue 5.2:

Number Crunching: The Building Blocks of All Computing * It Figures: An equation calculator that'll crunch your numbers accurately * Evacu-Pod: See if you can rescue all the miners in this challenging space game * Switch 'n' Spell: Electronic anagram brain teasers to puzzle over (for children, and adults) * Laserithmetic: Strut your math skill with this space fantasy edu-game * Organizer Reports: An enhancement to print-out your organized thoughts (see The Organizer HCM 5.1) * Razzle Dazzle: Tinker with musical sounds, or Play it Maestro! * What is CP/M?: Learn the Basics of Control Programming for Microcomputers * Apple Seedlings: Sorting out your ProDOS Catalog * Commodore Hornblower: Discover what's inside the Commodore 64's SID chip * IBM Impressions: Create 3-D surface drawings in BASIC * Field & Screen: A tutorial for using a Data Base System—correctly * Product Reviews * HCM One Liners * HCM TECH NOTES: Apple, C-64, IBM, and 99/4A * Product News, and much, much more!

CONTENTS: ON TAPE™ & ON DISK™

Evacu-Pod (A, C, I, TX)
Miner rescue in space
It Figures! (A, C, I, TX)
A mighty equation calculator
Laserithmetic (A, C, I, T)
Blast aliens with your math skills
Organizer Reports (A, C*, I, TX*)
Print your organized outlines

Switch 'n' Spell (A, C, I, T)
A spelling aid that's fun to boot
Apple Seedlings (A)
Sort your ProDOS catalogs
Commodore Hornblower (C)
Inside the SID chip
IBM Impressions (I)
3-D surface drawing in BASIC

This space reserved for Issue 5.3

This space reserved for Issue 5.4

This space reserved for Issue 5.5

FOR NEW READERS



The Plain & Simple Truth About **HOME COMPUTER**[™] magazine

Chock Full of Valuable Software & How-To Articles Without Filler

Every issue is a software "horn of plenty" with dozens of type-in-and-RUN programs printed in an easy-to-read listings format. Our programs are also available on inexpensive disks or cassettes for those who prefer the convenience of ready-to-RUN software. Step-by-step tutorials round out each issue, providing the solid facts you need without fluff or filler. Thus, each issue functions as an excellent reference work, as well as a valuable software source.



No Outside Advertising

Freed from the pressures of servicing *advertisers*, we concentrate on serving our *readers*. Each issue provides uninterrupted editorial flow and graphic layouts for better comprehension—plus unbiased product reviews which focus on true strengths and weaknesses, wherever the chips may fall . . . And we don't have to worry about losing advertisers because of publishing software in the magazine that is "too good." Consequently, we can provide the best free software available anywhere.



Focused on the 4 Hot Home Brands

We are 4 system-specific magazines under one wrapper—not a sprawling, "general interest" publication which attempts to cover too wide a field, only to spread itself too thin. The other side of the coin to this focused approach is the knowledge you gain from being exposed to the many tips, ideas, and techniques we provide for 3 of the 4 systems you may not even have. You'll learn more about your Apple, Commodore, IBM, or Texas Instruments home computer from this one magazine than from a host of more limited sources.



A Balanced Mix For a Perfect Recipe

In each issue we strive for a perfect balance of productivity, entertainment, education, utilities, and computer literacy—serving the needs of novice and pro alike. Every issue is a full-course meal, with a smorgasboard of tasty dishes for all palates. Whereas other computer magazines may dish out lumps of "editorial indigestion," we serve up a satisfying blend—one digestible byte at a time.



—Welcome to Our World of Home Computing

Home Computer Magazine (ISSN 0747-055X) is published ten times per year by Emerald Valley Publishing Co., P.O. Box 70288, Eugene, OR 97401. The editorial office is located at 1500 Valley River Drive, Suite 250, Eugene, OR 97401 (Tel. 503-485-8796). Subscription rates in U.S. and its possessions are \$25 for one year, \$45 for two years, and \$63 for three years. In Canada and Mexico add \$11 per year. Other foreign countries \$43 for one year surface mail. Inquire for air delivery. Single copy price in U.S. and its possessions is \$3.50, and \$4.50 in Canada and Mexico. Foreign subscription payment should be in United States funds drawn on a U.S. bank. Second-class postage paid at Eugene, OR 97401, and Columbia, MO 65201.

POSTMASTER: Send all address changes to Home Computer Magazine, P. O. Box 70288, Eugene, OR 97401. Subscribers should send all correspondence about subscriptions to above address.

Address all editorial correspondence to the Editor at Home Computer Magazine, 1500 Valley River Drive, Suite 250, Eugene, OR 97401. Unacceptable manuscripts will be returned if accompanied by sufficient first class postage and self-addressed envelope. Not responsible for lost manuscripts, photos, or program media. Opinions expressed by the authors are not necessarily those of Home Computer Magazine. All mail directed to the Editor or to the "Letters to the Editor" column will be treated as unconditionally assigned for publication, copyright purposes, and use in any other publication or brochure, and are subject to Home Computer Magazine's unrestricted right to edit and comment. Home Computer Magazine assumes no liability for errors in articles, programs, or advertisements. Mention of products by trade name in editorial material or advertisements contained herein in no way constitutes endorsement of the product or products by Home Computer Magazine or the publisher unless explicitly stated.

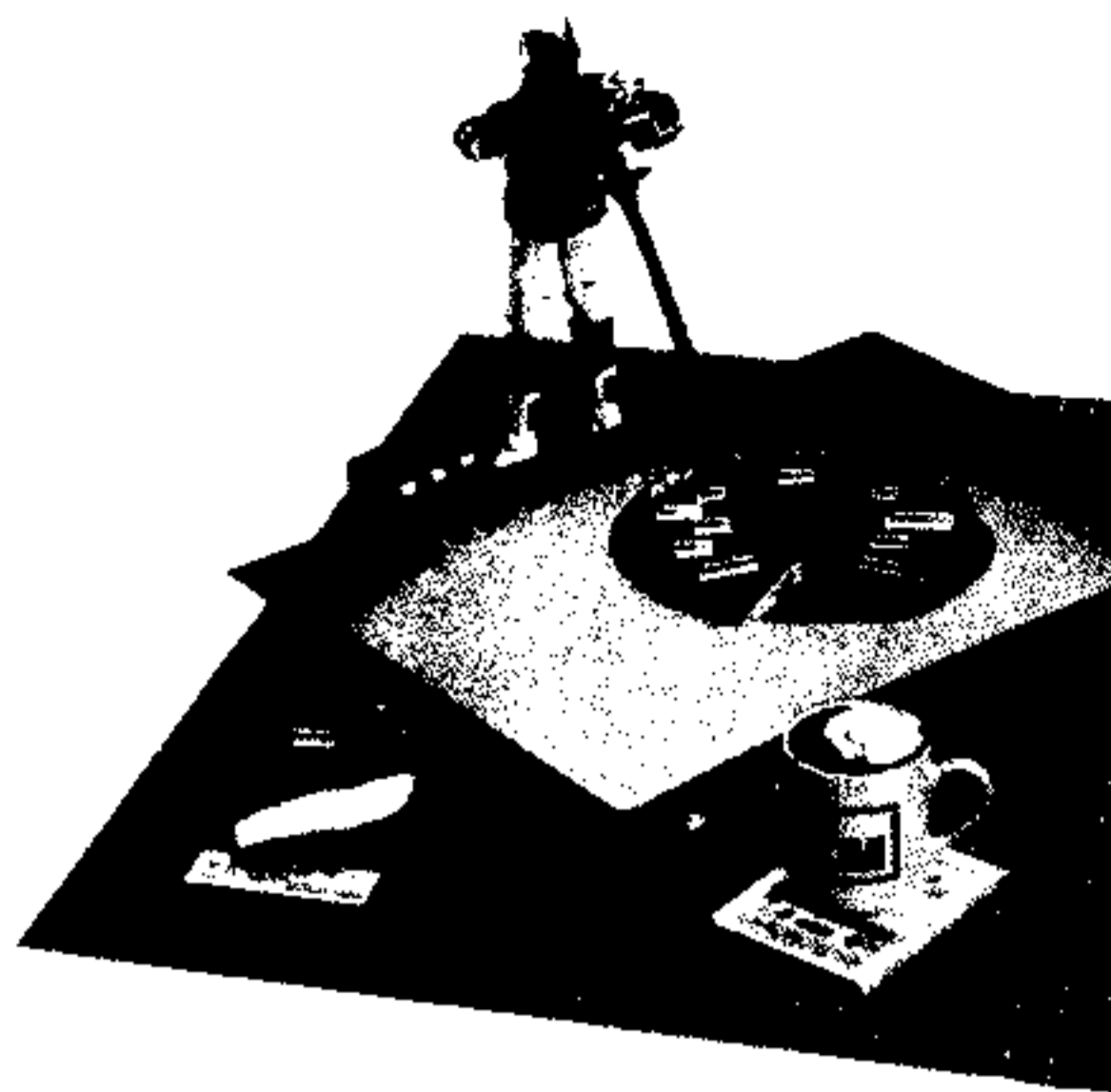
Each separate contribution to this May 1985 issue and the issue as a collective work is Copyright © 1985 by Emerald Valley Publishing Co. All rights reserved. Copying done for other than personal or internal reference use without the permission of Emerald Valley Publishing Co. is prohibited. Requests for special permission or bulk orders should be addressed to the publisher.

Limited License for use of programs in Home Computer Magazine. Emerald Valley Publishing Co. (EVP) is the owner of all rights to the computer programs and software published in this magazine. To allow for use of the software by the purchaser of the magazine, EVP grants to such purchaser only, the limited license to enter these programs into the purchaser's computer, and to place such programs on a diskette or cassette for the purchaser's personal use.

Any other use, distribution, sale, or copying of these computer programs without the written consent of EVP is expressly prohibited and in violation of this limited license and the copyright laws.

Home Computer Magazine, HCM, and Home Computer Digest are trademarks of Emerald Valley Publishing Co.

Publisher/Editor-in-Chief Gary M. Kaplan
Executive Editor David G. Brader
Managing Editor Walter Hego
Associate Editor Wayne Koberstein
Sr. Technical Editors
William K. Balthrop, Roger Wood
Technical Editors
D. Donaldson, Tom Green, G.R. Michaels,
Steven P. Nelson, Patricia Swift
User Group Editor Judy Campbell
Assistant Editor Dana M. Campbell
Program Translators
Hendrik Broekhoff, Stephen A. Gordon,
Jeff Fund, Paul Klissner, Robert Paschelke,
Randy Thompson, Nancy Vendelin,
Jim Whitaker
Asst. to the Publisher Rhea J. Grundy
Production Manager Norman Winney, Jr.
Creative Director Gei-Lei Gom
Photography
Nelson Stevens, K.D. Wainsworth
Production Assistant Rachel Knight
Customer Service Tel. (503) 341-1029
Dealer Sales & Distribution Tel. (503) 341-1036
Main Switchboard Tel. (503) 485-8796



Outside HCM

Here it comes to save the day . . . *Budgetron* is on the way! No matter how fancy the pie—from Ma's apple to Sara Lee's rich chocolate cream—someone has to slice it, making sure there's enough to go around. And from your house to the White House, computer budgeting may be the mighty hero of the day. So instead of spending straight out of your back pocket, put that pie on your computer screen. In time, your savings could, in fact, be the biggest slice.

Inside HCM

Is it work . . . or is it play? Computers often perform real work by making it look like play. Absorbing vast amounts of tedium, they let us humans have all the fun. At its best, computer software generates enjoyment as a way of coaxing out of us just the right information to get the job done. Perhaps the ultimate productivity package of the future will seem more like a video game—with an intriguing scenario that turns "user-input" into a series of strategic game "plays."

It's in the same spirit of productive play that we have developed this issue's handy key-in-and-RUN software. For example, take *Budgetron*—a monthly finance calculator with a dollar-wise sword. If you've had trouble slicing your budgetary pie, stop and let our friendly cybernetic hero "hack" your budget into shape. Not only will your pie reach farther—but by accounting for every crumb, you may find it to be even richer to the taste.

As the pie slices, the world divides into measurable bytes. Playing with basic two-dimensional shapes is one way we learn how to "measure the earth." If learning (or teaching) is your aim, train your sights on *Geometrix*—a tricky computer exercise with both a practical and an educational purpose.

Play often imitates work—and a lot of work comes disguised as play. In this issue, two game programs imitate the workaday world of war and power. *Torpedo Alley* simulates a lively contest of death beneath the waves. And, departing from the usual "Fire when ready, Gridley" format, *Over-Reaction* puts you in charge of a dynamic nuclear energy system where the order of the day is, "Stay Cool."

Among our regular features, *Apple Seedlings* nurtures helpful utilities such

as this issue's character graphics editor. Our second *Commodore Hornblower* column continues to build a BASIC synthesizer for the C-64. Sound is also the subject of *IBMpressions*, which in this issue illustrates simple sine waves blending into complex audible patterns. And *Razzle Dazzle* shows how TI-sprites bring depth to an animated sequence.

There's even more work at play in our LOGO program features: *Achilles and the Turtle*; and *Build a LOGO Adventure, Part 3*. We also provide a real programmer's gem in Part 1 of *Speeding Up a BASIC Program*.

Software is our work, but we also take time for some serious play—reviewing a full-range of products for your home computer. In this issue, we key-in on a host of typing instruction packages; talk sense about *Dollars and Sense*, a best-selling financial planner; and take a sharp-eyed view of the *Gibson Light Pen*. Or, if you want to really get your computer off the ground, come fly with us as we reconnoiter *Flight Simulator II*.

Fortunately, our reconnoitering doesn't end up in the clouds. Back down on the ground, we uncover a snappy utility for the C-64, tongue-lash a speech synthesizer for the IBM PCjr, dig up a *Missing Link* to connect the TI-99/4A to a parallel printer, and unscramble a computer version of *Scrabble*. Then, in Part 2 of *The Music of Sound*, we compare the C-64 musical software reviewed in Part 1 to the Casio CT-6000, a professional-level keyboard with similar sound and music capabilities.

So, to turn your work into play—or to make your play work for you—start turning the pages of *Home Computer Magazine* today.

Until next time, have fun reading, learning, and RUNing

HCM

By Gary M. Kaplan
Publisher & Editor-in-Chief

Keepp 'em coming—the “On Screen Feedback” letters, that is. In last issue’s column, I asked for your ideas on ways to “fine tune” this magazine so that we can provide more of what you want, and less of what you don’t want. Your response has been, as my son would say, “totally awesome!” Poring over the daily sacks of mail has, above all, convinced me of two things: First, I must have been slightly delirious when I invited this mass response upon myself; and second, *HCM* reader feedback in the form specified last issue is undoubtedly an effective way to unleash a staggering tidal wave of creativity, constructive criticism, and fresh new ideas . . .

Even though I asked you to focus your comments exclusively on the *content* of the magazine, many of you didn’t. Instead, you also voiced your dissatisfaction with the long wait from the time your order for *ON DISK* or *ON TAPE* is placed, until the disk or tape actually arrives by mail. All along, we assumed that you would favor the trade-off in timeliness versus cost—hence utilization of Third-Class bulk mail (with its inherent delays caused by requisite batching and sorting up front, plus the slower delivery and “third-class” handling that the postal service relegates to bulk mail). Well, as the line goes—we’ve finally seen the light! Effective immediately, all *ON DISK* and *ON TAPE* media will enjoy speedy First-Class treatment, plus stronger mailing containers to ensure that your software arrives in “first-class” condition (with automatic forwarding upon change of address). To pay for all this, we’ve been forced to pass along a one-dollar shipping and handling charge on all single-disk and single-tape orders. We’re sure you’ll soon realize that this small extra cost will make all the difference in the world.

Incidentally, many of you have thanked us for the extra file or template that we occasionally put on our magnetic media when appropriate. We’ll be doing more and more of this for “extras” that we can’t put in the magazine because of space limitations and key-in problems. One piece of related good news: With our current *ON DISK* and *ON TAPE* Program Subscription service, we’re able to absorb the lion’s share of the extra first-class shipping charges due to a subscription’s much lower one-time transaction processing cost. Thus, we can pass on to you even greater value and savings. I suggest that you take a look at our special “get acquainted” offer found on the multi-color side of the *center* bind-in card. You’ll be very glad you did!

Many of our PCjr readers have already written and expressed concern with IBM’s decision to cancel further production of their machine. I’d like to take this opportunity to assure all Junior users out there that we will definitely continue our full support of the PCjr. This complete tutorial, review, program, and media support is especially crucial when a manufacturer “orphans” a



“One day, our program-design staff found itself musing over the idea of doing something very different in the pages of our magazine . . .”

machine. Our commitment to excellence cannot be altered by a manufacturer’s decision—even one the size of Big Blue. So please, spread the word to all your fellow users of PCs, PCjrs, and PC-compatibles that *HCM* is the *only* magazine that can, or is willing to, provide this total user support.

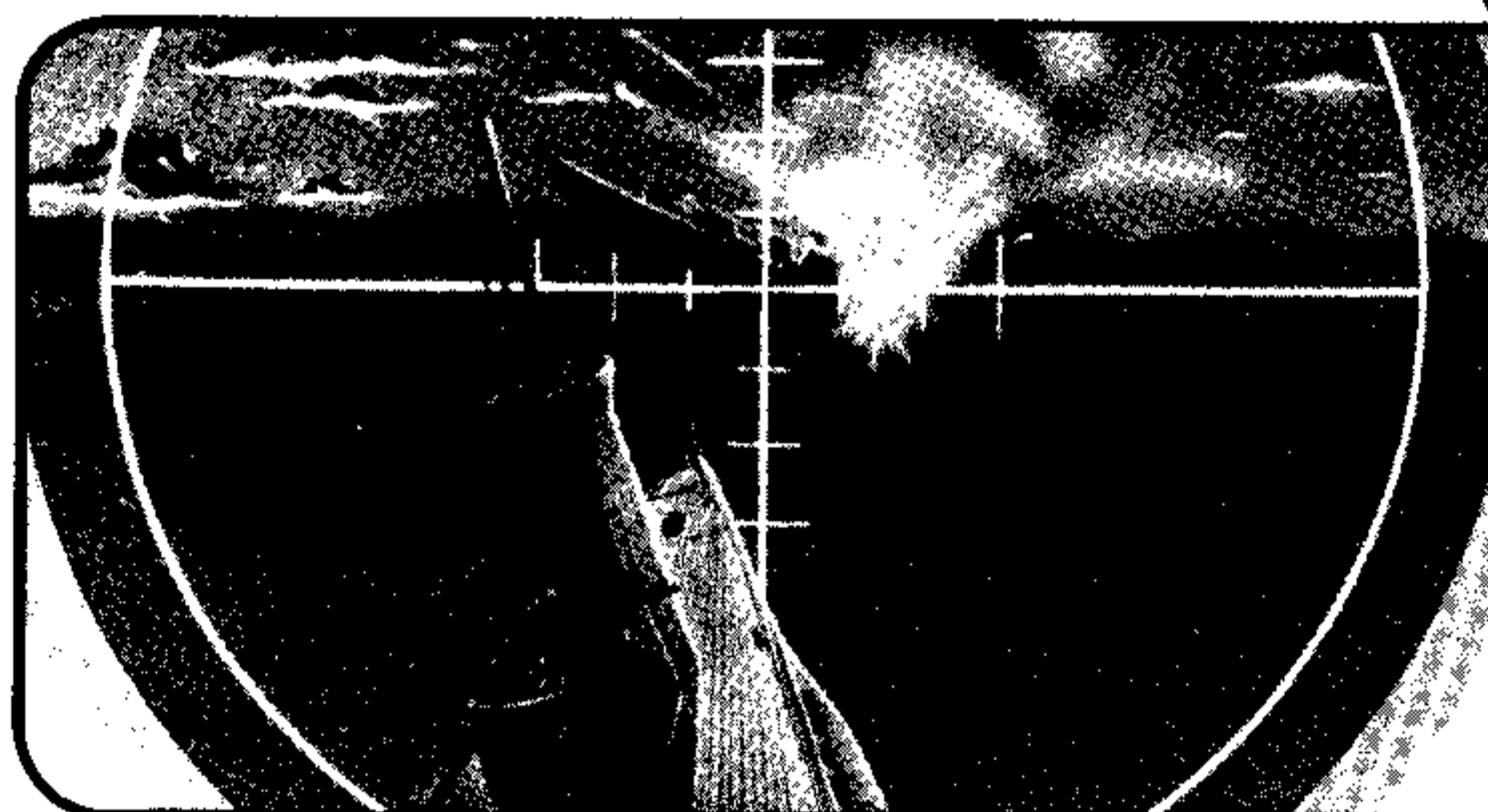
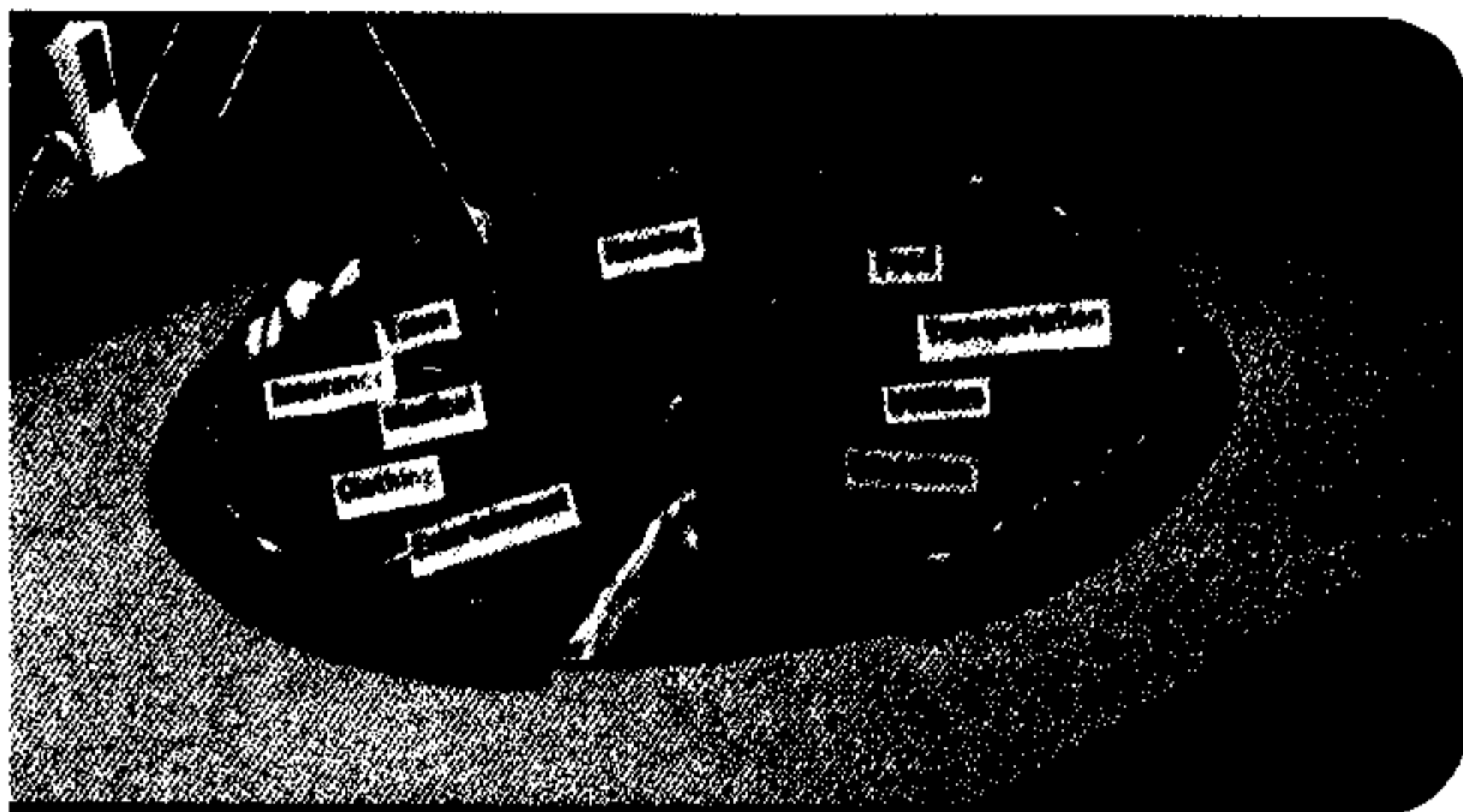
Id like to now discuss one of the programs included in this issue. We’ve called it *Over-Reaction*. First, I’d like to state that this program is *not* just a game. To call it “just a game” is like calling a Rolls Royce “just a car” (or like calling *HCM* “just a computer magazine” . . .). So, those of you who tell me you’re not interested in games, please take note:

One day our program-design staff found itself musing over the idea of doing something very different in the pages of our magazine—providing a realistic, process-control activity in which our readers could participate in a stimulating learning experience.































Our goal was thus to design a real-time activity that required dynamic strategy, rather than the instant responses and quick trigger-finger demanded by a typical video game. We wanted to simulate the experience of maintaining a complex system—one that is subject to many outside influences beyond the player’s control. The perfect scenario eventually suggested itself: a runaway nuclear power plant. In this model, Murphy would have the upper hand. Everything that could go wrong, would. Fortunately, one of our programmers had actually worked in a nuclear plant, and we were thus able to build-in some first-hand realism to boot.












Our first version of the program, however, turned out to be too stable. In fact, our in-house “play-testers” facetiously groaned in unison, “it’s *boring* to the core.” In subsequent versions, we finally assuaged the snide snickering from our sarcastic staff by introducing more entropy into the algorithm. In short, we made more things go wrong, more often—thus, *compressing time*. The result: an exciting real-time simulation—a “game” that should appeal even to those who have no taste for the usual shoot-em-up video fare. It’s one I thoroughly enjoy “playing.” I think you will too.

HOME COMPUTER™ magazine



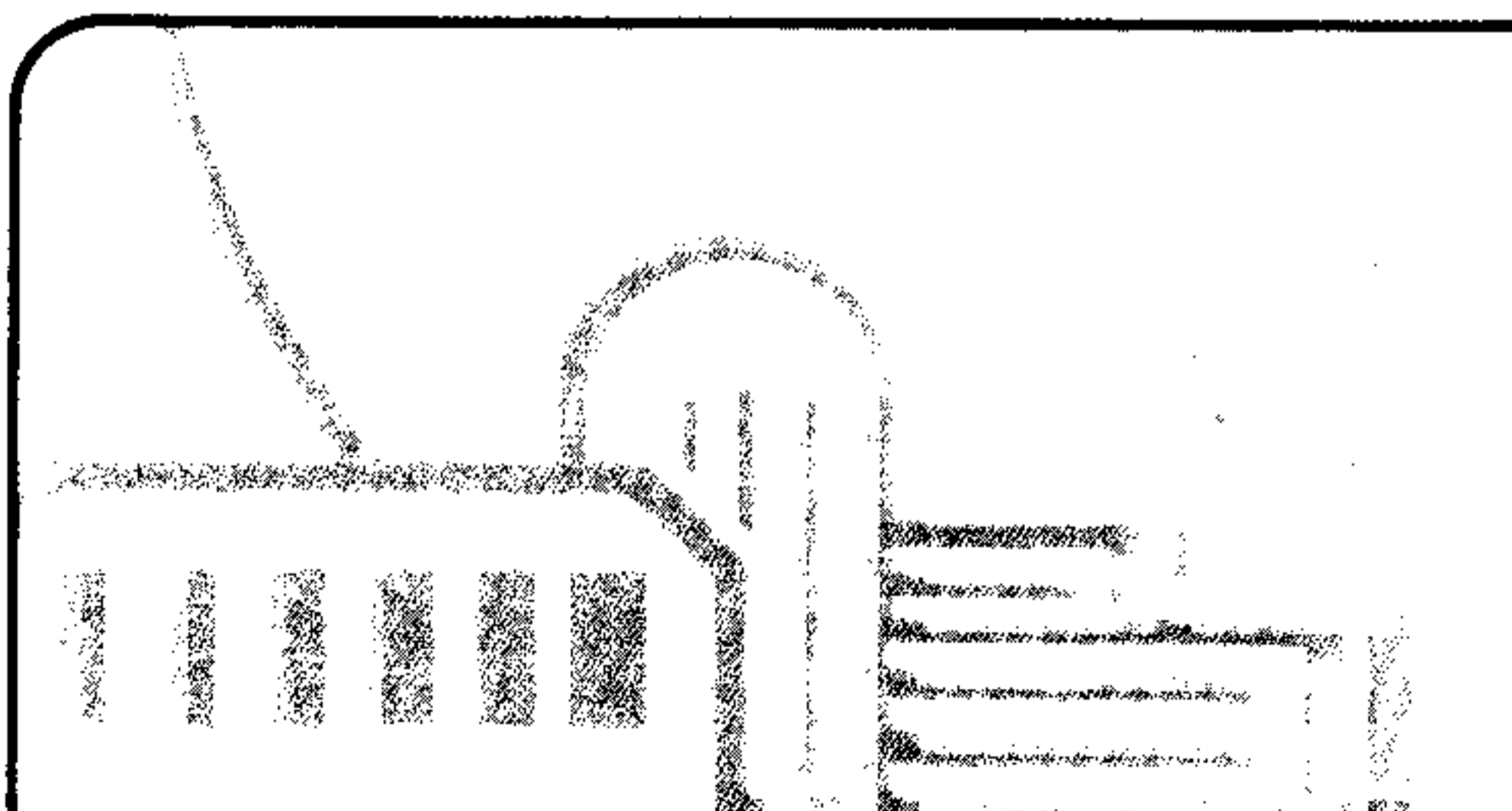
FEATURES

- 16 Budgetron**      
A mighty hero to rescue your budget. *by Scott Williams and the HCM Staff*
- 20 Over-Reaction**      
Keep this reactor cool—but don't overreact. *by HCM Staff*
- 23 Torpedo Alley**      
Fire torpedoes, then dive, dive! *by William K. Balthrop*
- 26 Geometrix**      
Basic shapes compose the world. *by Wayne Koberstein*
- 55 Build A LOGO Adventure**     
In part 3, we add objects to our Adventure Land. *by Andrew Keith and the HCM Staff*
- 58 Achilles and the Turtle** 
Can Achilles ever beat the turtle? *by William M. Goodman and the HCM Staff*

- 62 Apple Seedlings** 
Generate character graphics on the hi-res screen. *by Steve Cordon and the HCM Staff*
- 64 Commodore Hornblower** 
Select waveforms and envelopes from SID. *by Roger Wood*
- 68 IBMpressions**  
Blending sine waves into complex patterns. *by William K. Balthrop*
- 70 Razzle Dazzle** 
Multi-layered animation with TI sprites. *by William K. Balthrop*
- 74 MAC-ROS** 
Expanding BASIC on Macintosh. *by Mac N. Tosh*
- 78 Speeding Up A BASIC Program**     
Part 1 teaches the first steps to speed up BASIC. *by John P. Russo and the HCM Staff*

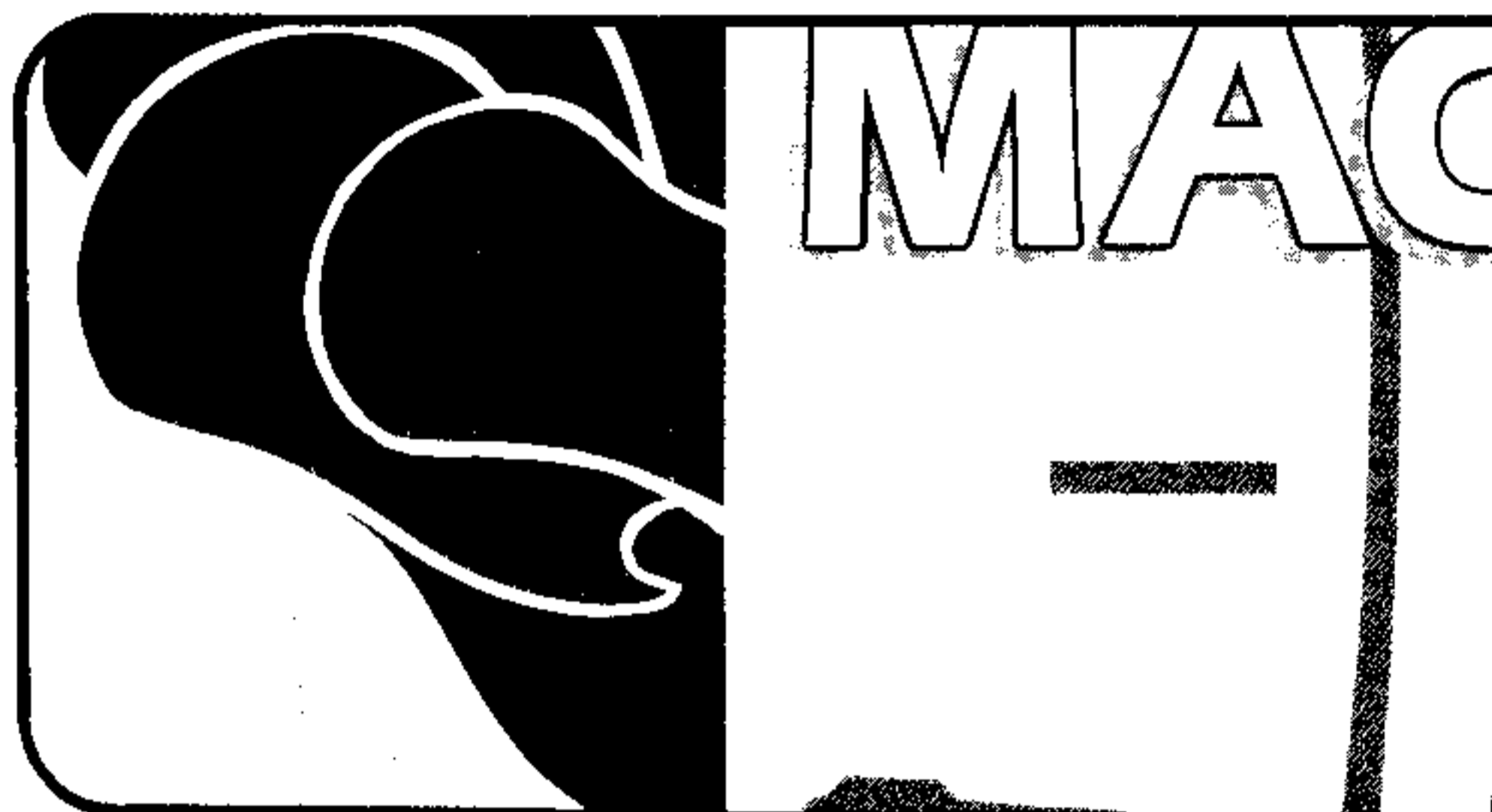
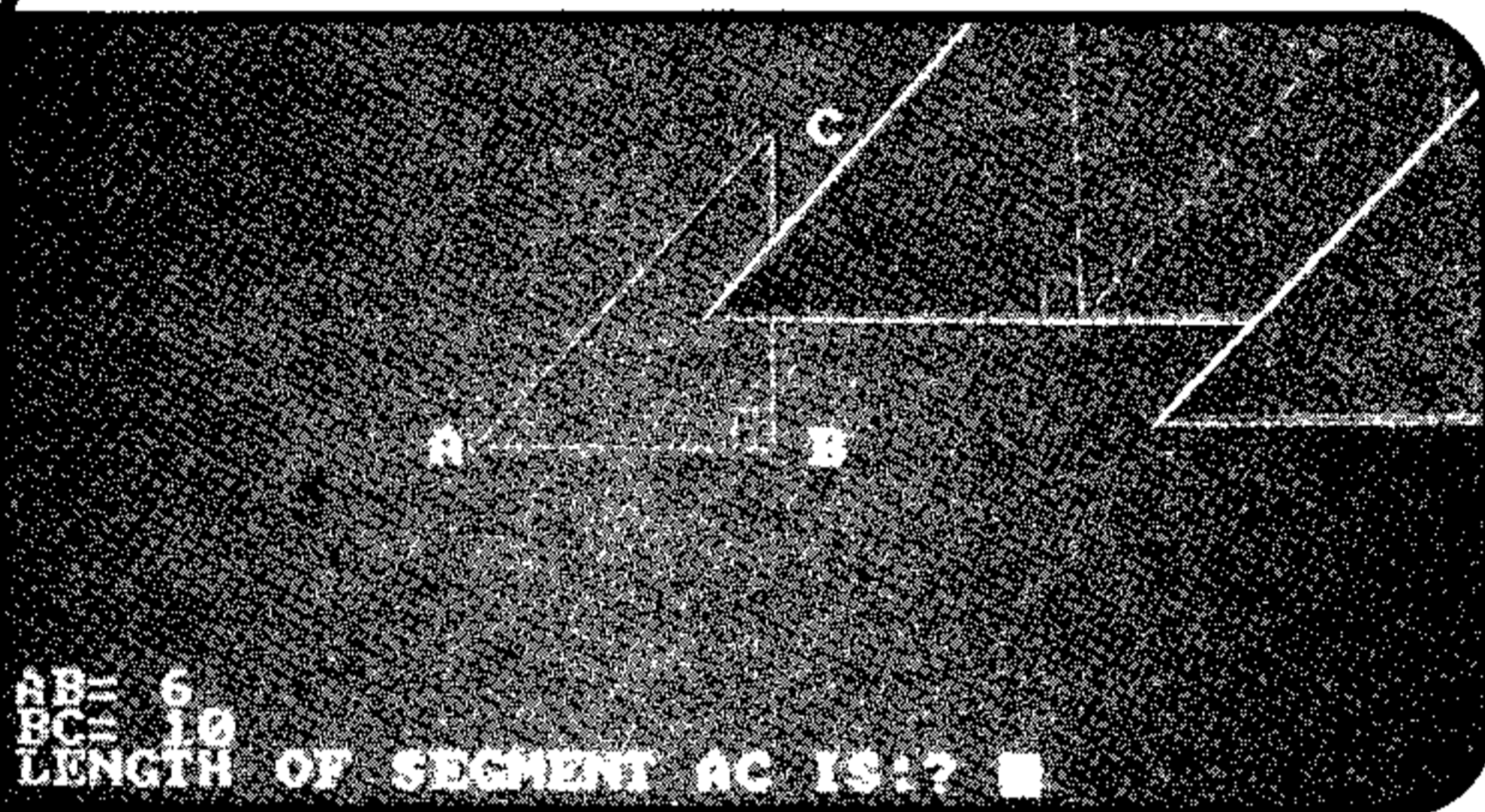
PRODUCT REVIEWS

- 32 The Gibson Light Pen** 
Sketch or paint right on the screen. *A Review*



CONTENTS

VOLUME 5 NUMBER 3



35 Monty Plays Scrabble    
Is the computer a worthy word opponent? *A Review*






36 The Missing Link for the TI-99/4A 
Hook your computer to a parallel printer—inexpensively. *A Review*

38 Dollars and Sense    
Does it make sense to spend dollars on this program? *A Review*



**40 The Music of Sound Part 2:
Music Software for the C-64 vs.
Casio's CT-6000 Keyboard** 
Comparing musical hardware to musical software. *A Review*

**44 Junior's First Words:
A Review of the PCjr Speech Attachment** 
This speech synthesizer has little to say.

**46 Commodore-Taming for the Shrewd:
A Review of SysRes** 
Coming to the aid of C-64 programmers.

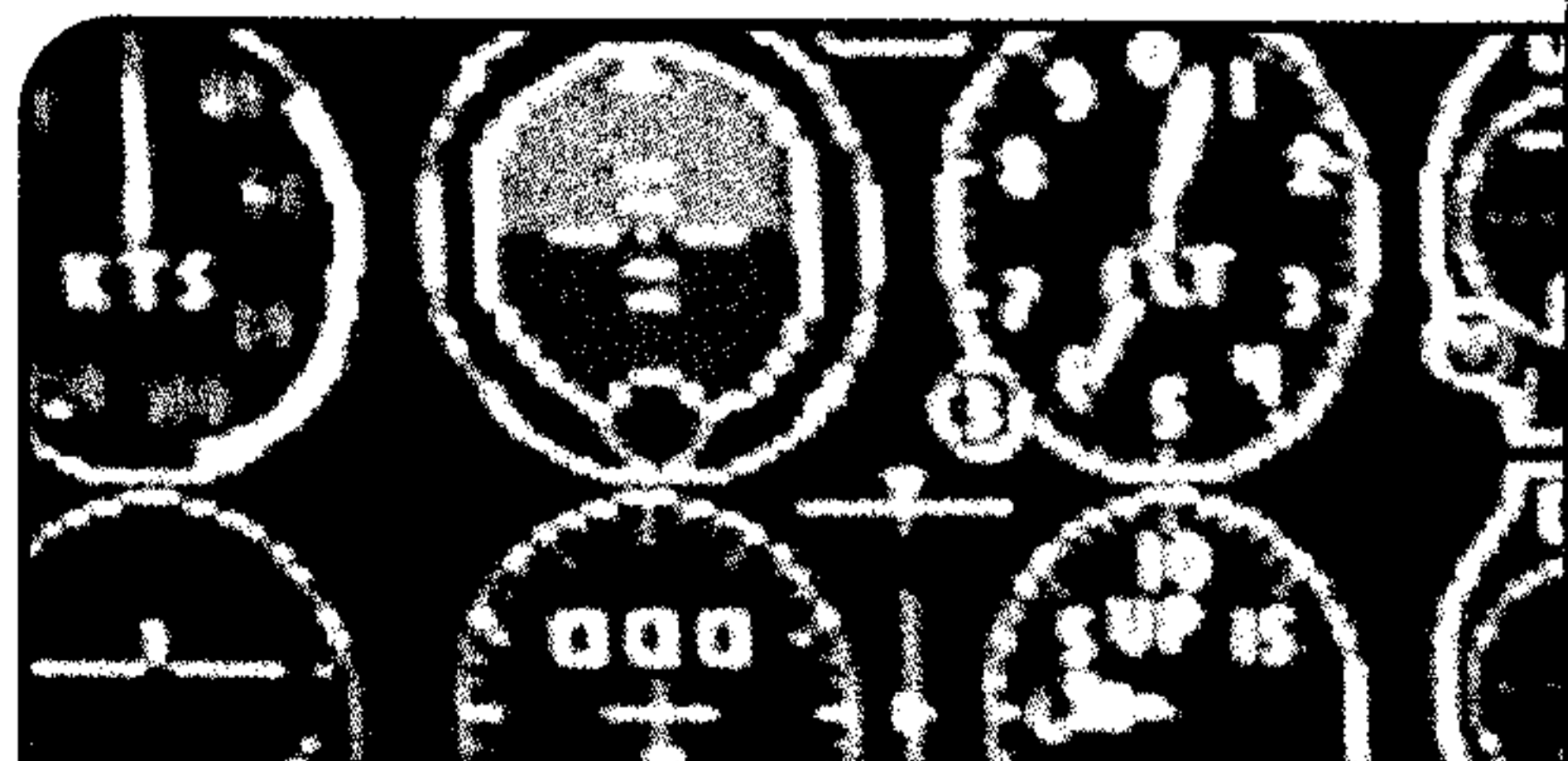
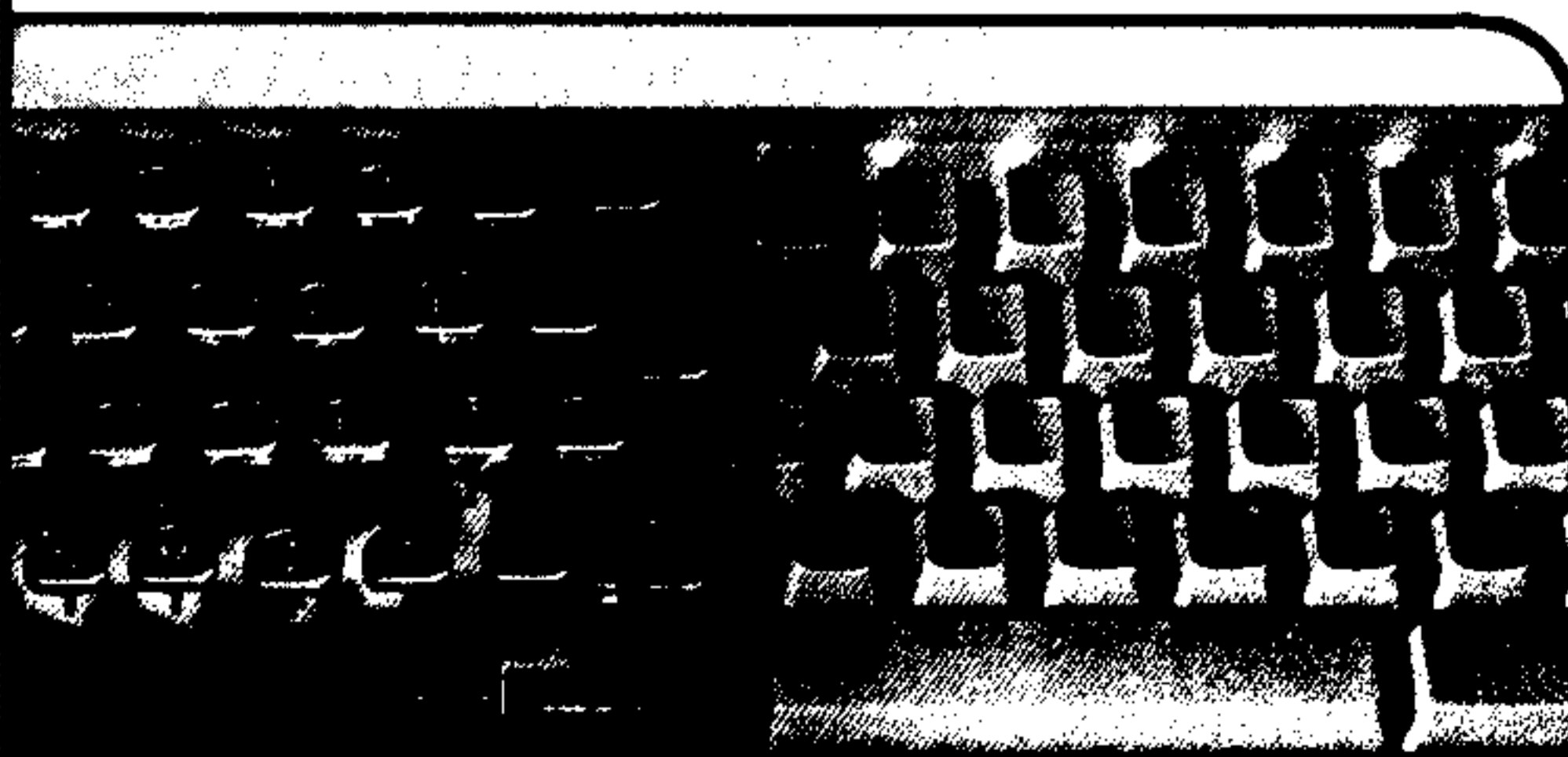
**48 Computer-Age Typing:
A Look at Some Key
Typing-Instructor Programs**     
Learning to type ain't what it used to be. *A Review*

**52 Going Solo:
A Review of Flight Simulator II**    
Flying by the seat of your computer.

82 Moon Patrol    
Eliminate the nasties in Sector 9. *A Review*

DEPARTMENTS

5	Welcome to HCM	89	Program Listing Contents
6	Inside/Outside HCM	90	Program Typing Guide
7	On Screen	127	DeBugs on Display
11	Letters to the Editor		
30	HCM One Liners		Home Computer Tech Notes:
31	HCM Review Criteria	66	Apple
60	Industry Watch	67	Commodore
76	Group Grapevine	72	IBM
83	HCM Product News	73	TI



HISTORICAL NOTE

99'er Magazine (founded in December, 1980) was the forerunner of Home Computer Magazine.

THE BEST

**SUPER
CLOSE-OUT
SPECIAL
FOR
TI-99/4A
USERS**

A Giant Home Computer Compendium™
for the Texas Instruments 99/4A

OF 99'er™

The largest, most comprehensive collection of programs and articles ever assembled for the TI Home Computer

VOLUME 1

- Over 200 thoroughly tested key-in-and-RUN programs and sub-programs typeset in a grid format for maximum clarity.
- Programming instruction in 4 languages—learn to use BASIC, Extended BASIC, LOGO and Assembly Language—for everything from record keeping and money management to arcade-quality action games.
- A selection of sensational game software featuring full-color graphics, animation, and sound effects.
- Beyond the owner's manual—tips and techniques for getting the most out of your computer system.
- Computer-Assisted Instruction—The home computer becomes your private tutor.
- Page after page of innovative applications—transforming your computer into a home productivity center.

Regular (Pre-Close-out) Prices:

Best Of 99'er (Book alone)	\$19.95 + \$3.00 shipping
Best Of 99'er (Tape Set alone)	\$35.00 + \$2.50 shipping

USE BIND-IN CARD AT CENTER OF MAGAZINE

SPECIAL OFFER

Buy the Tape Set for **ONLY \$35.**
And Get the Book **FREE** + **FREE SHIPPING**

FREE BONUS WHILE SUPPLIES LAST

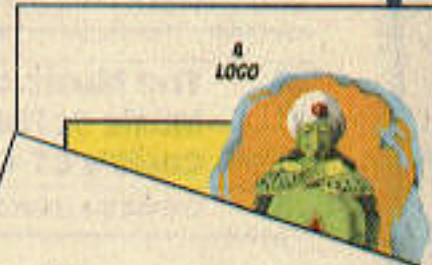


ORDER THE BOOK & TAPE PACKAGE AND RECEIVE A Simon's Saucer AND A 99'er Programmer's Guide **ABSOLUTELY FREE!** This Additional \$18 gift IS YOURS IF YOU ACT TODAY!



FREE BONUS WHILE SUPPLIES LAST

PLUS



THE BEST OF 99'er ON TAPE
A Choice Selection of 37 Full-length Programs On 5 Quality Cassette Tapes

★ Save Typing Time and Frustrating Key-in Errors.
★ Own the Most Comprehensive Software Library for the TI-99/4A.
★ Enjoy Hundreds of Hours of Exciting Computer Activity.

TO ORDER—USE BIND-IN CARD AT CENTER OF MAGAZINE



People Who Know the Home Computer Best™

HOME COMPUTER™

magazine



IF CENTER BIND-IN ORDER CARD IS MISSING—
MAIL WRITTEN ORDER TO:
P.O. Box 70288 Eugene, OR 97401
FOR CREDIT CARD ORDERS CALL 1-(800)-828-2212

Expiration Issue Clearly Marked

Dear Sir:

Congratulations! Last issue you said your new schedule would have approximately 5 to 6 weeks between issues, and my latest issue (Vol. 5, No. 1) arrived 6 weeks later. After all the shifting around you've done since production stopped on the 99/4A this is a real milestone. Just one comment though. Your recent issues have just had a volume and issue number—no date—and my subscription label gives an expiration date (month and year). This makes it hard to figure how many issues are left in my subscription. Why not continue putting the month on the magazine with two issues each year covering two months, such as June/July and December/January? This would make it easier for your subscribers to keep track of the time on their subscriptions.

Thomas Steinhart
Decatur, IL

No sooner said than changed, Thomas. If you will look at your present address label, you will see that it no longer has an expiration "date." It now has the volume and issue number of the last issue to which your subscription entitles you. This is the clearest method that we could devise for subscribers, now that we are on a ten-issue per year publication schedule. To avoid confusion at the newsstand, we must presently refrain from putting a date on the cover.

Your Days are Numbered

Dear Sir:

In the Letters column of Vol. 5, No. 1, Mr. Donald Mefford asked if anyone knew an algorithm to determine the number of days between two dates. This sort of calculation is quite common in astronomy where Julian Date numbers are used. The Julian Date is the number of days since January 1, 4713 B.C. To determine the number of days between two dates you simply subtract their Julian Day numbers. A subprogram written in TI Extended BASIC to determine the Julian Date follows:

```
100 SUB JULIANDAY(M,D,YR,JD)
110 IF M = 1 OR M = 2 THEN F = YR - 1
ELSE F = YR
120 IF M = 1 OR M = 2 THEN MF = M + 12
ELSE MF = M
130 A = INT(F/100) :: B = 2 - A + INT(A/4)
:: C = INT(365.25*F) :: DP = INT(30.6001
*(MF + 1))
140 IF YR <= 1582 AND M <= 10 AND
D <= 15 THEN B = 0
150 JD = B + C + DP + D + 1720994.5
160 SUBEND
```

The Julian Day is returned in JD. This routine will find JD for any date; if you're only interested in dates this century you have to enter the full four-digit year. The day of the week = (JD + 1.5)/7 where Sunday = 0, etc.

Max A. Shelhorse
Edina, MN 55435

Say, that's great, Max, but in testing your program we seem to have found a "short-term" bug. We used your suggestion to see how many days there are between January 1st to March 1st of the same year and arrived at 65! Still,

it gives very close results even for periods of several hundred years. Maybe one of you other readers could shed some light on this Julian date mix-up.

More on Adding a Drive to PCjr

Dear Sir:

Absolutely delighted with Mr. Brader's article "Adding a Second Drive to the PCjr" (HCM Vol. 4, No. 4). The added drives work like a champ! I added a third drive also, since I had one on hand and it was especially easy because the extra logic was already on board with the piggyback chips. Now I believe my Junior is even better than a PC because of the additional colors available in medium and high resolution, and of course, the more versatile sound.

I had the same trouble initially that Mr. Beifuss reported in his letter (Vol. 4, No. 5), getting an error on bootup. The problem turned out to be that the 74LS10 gate just wasn't able to sink enough current to fully select my Shugart half-height drives. Swapping the chip with a standard 7410 did the trick.

Your readers might be interested to know that the software which comes with Tecmar's jrCaptain add-on memory board enables the PCjr to recognize the new drives without having to make the MODBOOT.BAT file described in the article. Tecmar's CONPCJR.EXE program does the job, and perhaps it can be purchased without purchasing the memory board (although I highly recommend the board).

Keep up the good work.

Ron Sutherland
Vienna, VA 22180

Thanks, Ron, for the appreciation shown and the information on the Tecmar CONPCJR.EXE program. For those readers who had not been aware of the construction article and would like to get a copy, write or call the HCM subscription line and ask for the HCM Vol. 4, No. 4 back issue for \$3.95. Also, our PCjr Second Disk Drive Kit is still available for \$49.95. This kit contains the special cable and the two ICs mentioned in the article. With the copy of the article and the kit of parts, all that is needed is an external IBM-compatible disk drive.

Delay Loop Trick for the C-64

Dear Sir:

Simon's BASIC contains a convenient function, PAUSE, which allows the programmer to specify a pause in the program for a specified number of seconds. In Commodore 64 BASIC, a pause is often created with a FOR . . . NEXT loop.

Using a FOR . . . NEXT loop to create a delay has two disadvantages. First, unless you use the technique frequently enough to have a feel for it, you must experiment to find the proper range for your loop counter. Second, if you use one of the compilers now on the market, you will need to redetermine all of your loop counters, since the compiler makes the loops run faster.

There is another way to create a delay in the program for a specified number of seconds, without the bother of experimentation, and

which will be unaffected by compiling the program. Set the value of S in line 500 equal to the desired delay in seconds:

```
. . . program ready for pause
500 S = 30:GOSUB 900:REM PAUSE 30
SECONDS
. . . more program after pause
. . . END
900 T = TI + 60*S:REM PAUSE
SUBROUTINE
910 IF T = TI THEN RETURN
920 GOTO 910
```

Since this technique makes use of the internal clock, rather than the time required to execute a FOR . . . NEXT loop, it requires no experimentation and is immune to a compiler. And, if desired, you could even insert a GOSUB between 910 and 920 for a routine which does something during the pause.

Perhaps your readers would find this technique useful.

Jack Ryan
El Dorado, AR 71730

Pretty tricky, Jack! You notice we didn't "delay" too long in publishing this little routine. In fact, some readers may consider it "just in time."

Locating a TI Disk Manager 2

Dear Sir:

In Vol. 5, No. 1 of HCM there was a letter from James A. Canter, who was looking for a Disk Manager 2 cartridge. After reading the same article ("2 for TI"—Vol. 4, No. 4), I attempted to locate a Disk Manager 2 cartridge and after calls to numerous dealers and to Texas Instruments, this is what I found.

There are only three ways to get a Disk Manager 2 cartridge:

1. Try a user group (which you suggested).
2. Buy another disk controller card (with DM 2).
3. Send your old Disk Manager to Texas Instruments.

Texas Instruments said that if we send in our old Disk Manager cartridge, they will upgrade it to a Disk Manager 2 for a fee of \$14.25 plus state tax plus \$2.00 shipping/handling.

They did ask that we call in advance before sending the cartridge to them. The number to call is 1-800-842-2737.

I hope this can be of help to James and anyone else wanting to upgrade their system.

Edward T. Stack
Edwardsville, IL 62025

For those of you who are not familiar with this solid-state cartridge for the TI-99/4A, it allows you to format and catalog double-sided disk drives (assuming that you have a double-sided disk drive connected to your system). The software in Disk Manager 2 is also set up to allow double-density recording, but the standard Texas Instruments disk drive controller card does not allow this. Some of the third-party disk controllers on the market may allow double-density recording. Watch for reviews of this third-party hardware in HCM in the near future. By the way, Edward, Texas Instruments does not require that you send the old cartridge.

Continued on next page

How To Buy An Apple

Dear Sir:

I am a soon-to-be owner of a computer. I know very little about the computer field. Words like interface, CP/M, extended memory cards, peripheral cards, buffers, hard and floppy disks, disk drives and more confuse me terribly!

I wish to buy an Apple IIe, a printer, one or two disk drives, a voice synthesizer, and have ProDOS, 128K, also a color monitor and surge protector.

I'd like to know how it would all fit together. Do I open the computer? If so, how and what tools would I need, if any, to put it all together? Do I need interfaces, cards, and other things along with the actual named products?

As you can probably see for yourself, I am really confused.

I'm writing to you because your magazine seems to be the best. I've bought other magazines yet none compares to yours. I know you will give me honest opinions on cost and quality—a trait your magazine solely seems to carry.

Dorene Pope
Niagara Falls, NY 14304

Judging from your letter, Dorene, we recommend that you buy your Apple computer from an Apple dealer in your hometown, and let him set up the computer configuration as you require. Often when purchased this way, you will get a package discount (something to ask about). When making the decision to purchase a system, try to get price quotations on the package from as many dealers in your area as possible (including the length of warranty on each item and the cost of any service policies). Once you have the entire picture, you can make an informed purchase decision. After you select a dealer, it is advisable to consider several alternatives. Your request for 128K, a printer, etc. may mean that the Apple IIc, with its many peripheral functions already built-in, is perfect—and cheaper! Much depends upon your particular application. We've found that AppleWorks, for example, works well on the IIc (see our review in HCM Vol. 5, No. 2). Your choice of a printer should be made with your needs in mind. If you will only be doing word-processing, a letter-quality printer might be the right choice. If, however, graphics is your cup of tea then a dot-matrix (like Apple's Imagewriter) or even a color printer might be more what you want. These are just some of the considerations that you should discuss with the dealer you select.

In Search of King's Quest for PCjr

Dear Sir:

Enclosed is my payment for a one-year subscription to a superb magazine. I have purchased the last two issues and find your magazine extremely interesting. Since our family recently bought a PCjr, we have read your articles extensively and programmed a few of the games. Incidentally, your game reviews and tech articles catch the interest of my children, which reflects that your magazine is geared for all ages.

We found your review of King's Quest in the Vol. 4, No. 5 issue so favorable that we tried to buy the game. The store that specializes in software tried to order it for us, but to no avail. They informed us that the game was only available for the IBM PC. They claimed their distributor called Sierra On-Line to confirm this. Can you verify this? The article in Vol. 4, No. 5 indicated PCjr.

Also, in our programming of your game Bird Brain, we have a problem while playing. During the course of the game we get an illegal function call in line 640 which we typed correctly. Can you help?

In Stadium Jumping, on Grand Prix level we are unable to play because of an illegal function call in line 510. Again, can you help?

Nick Tomasi
Defiance, OH 43512

Kings Quest was designed specifically for the PCjr and can be used on the PC if it has the special color graphics adapter and a color monitor. We recommend that you contact your closest IBM Product Center for availability. As to the two program bugs you mention—these appear to be key-in errors, as we have never had the difficulties you describe. You must realize that just because the error message names a line, it is not necessarily where the error occurs. For example, if one of the variables in line 640 of Stadium Jumping were a zero, the illegal function call would result. To save yourself a lot of time solving these pesky key-in bugs, we suggest you get the programs ON DISK—the small price you pay can save you hours of aggravation.

Making the Reset Button Work

Dear Sir:

I've noticed in your latest magazine that a couple of your readers had installed a reset button on their C-64 and were having problems listing and running their programs after depressing the reset button.

This little program I picked up may possibly be of some help to your readers.

Load and run this program first. Then type in your program. Should it become necessary to hit the reset button, enter the SYS 49152 and you should be able to list and recover your program.

```
10 AD=49152:FORI=0TO21
20 READ D:POKE AD+I,D:NEXT
30 DATA 169,8,141,2,8,32,51,165,24
40 DATA 165,34,105,2,133,45,165,35
50 DATA 105,0,133,46,96
60 PRINT"TO EXECUTE,USE SYS";AD;
":CLR"
```

Bill Asher
Detroit, MI

Thanks Bill. This is a good addition to the reset button Tech Note. Because other folks have written in saying that they could not recover their data, your little program will help immensely, and will make the use of the reset button more "user friendly." One change that some of you might want to make though concerns the starting address. Setting the variable AD in line 10 to equal 53225 would decrease

the possibility of this routine conflicting with other machine-language programs. Because machine-language programs "crashing" is a major cause of needing a reset button, storing this program at 53225 might be safer.

Tension Sensed Over Review

Dear Sir:

Calmpute is the outgrowth of 11 years of research, development, and production of biofeedback equipment by Thought Technology.

The Calmpute sensor accurately monitors changes in skin conductivity—a measure responsive to the activity of the body's sympathetic nervous system which is responsible in part for heart rate, blood pressure, peripheral temperature, and adrenaline responses to stress.

Unlike electromyographic monitoring which provides information solely about the specific muscles being monitored (in the case of RELAX, the frontal region of the forehead), Calmpute detects physiological changes related both to physical and mental stress. The anomalies you reported ("high responses to neutral words such as "wood" and "door") are indeed responses you produced, perhaps in frustration observing such simple words or because your mind wandered to something provocative. Another aspect involves "the perception of relaxation." One might report feeling relaxed, however his/her mind may not be relaxed. For example, simply watching the feedback trace on the screen going towards a stressful reading often produces stress in the novice user.

The point I wish to stress is that to develop relaxation skills requires practice.

To assist users, Sunburst Communications has developed a new more comprehensive manual and an optional 6-week stress management program in a 150-page book which they will offer with Calmpute in June 1985.

As well, we have rewritten Calmscope and Calmbar to provide color feedback, to print and save the displays, and have developed an interface to connect our temperature, heart rate, electromyogram, blood volume pulse, and respiration monitors to Calmpute.

Dr. Hal. K. Myers
President
Thought Technology, Ltd.

Thank you, Dr. Myers, for the update on the Calmpute product which we reviewed in Vol. 5, No. 1 of Home Computer Magazine. If our comments on your product caused you to upgrade it—making the package more useful and a better value—we're really glad.

Ti'ers in Great Britain Miss Ads

Dear Sir:

Congratulations on yet another wonderful edition of HCM Vol. 4, No. 5. I have been a subscriber for two years. I have the magazine posted from my nearest stockist (450 miles away). The content as usual was excellent. In many ways Ti'ers in England miss the ads because it was our only way to obtain information as to what and where anything was available for the 99—in particular, add-ons (disks, etc.)—because the supply of just about

everything for the 99 in the UK has dried up (tears and more tears!!).

Could you ask any readers if they can come up with any circuit diagrams and lists of parts for stand-alone 32K expansion or any other hardware because it seems our only method left is to build them ourselves.

Could you also give me information as to how I can receive software on cassette from your issues, as we buy our magazines in England. Hope you can help.

Malcolm Pryke
Newcastle Upon Tyne
ENGLAND

Malcolm, we will send you the information via post regarding purchasing HCM software and cassettes in the United Kingdom. Plus, we'll send some recent back issues of our sister publication, Home Computer Digest, which contains information on merchandise from U.S. vendors of TI-related wares. We believe that purchase from U.S. sources to be a better solution for you than building from scratch.

Apple Snap-Calc SNAFU?

Dear Sir:

I have a slight problem with the SNAP-CALC and SNAP-FIX software. I am using an Apple IIc with a single disk drive, and therefore am not able to convert SNAP-CALC from DOS 3.3 to ProDOS. When I received the SNAP-FIX update on a ProDOS disk, I looked on the back side of the disk for the DOS 3.3 version, as the instructions said, and found nothing. I am therefore not able to get the enhanced version of your spreadsheet program.

Now that I have my complaining out of the way, I'll get to the important stuff. I love the mag! Your format and concept are unchallenged by the others in the publishing field, and your coverage of technical articles seems to correlate perfectly with the questions that I have. You always seem to include the most informative pieces in the Letters to the Editor as well.

I never miss an opportunity to recommend your magazine to my friends who also are interested in computers, and am considering the possibility of giving a gift subscription of the magazine and the ON DISK to my children's school. Keep up the good work—I look forward to every issue.

Ron G. Reist
Kalamazoo, MI 49007

Sorry, Ron, that your disk didn't have the SNAPFIX file on it. The oversight has been corrected in more recent releases of the ON DISK 4.5. In addition, an even more recent update file (which includes all the changes that were in the SNAPFIX file) has been included on the back-side of the ON DISK for Vol. 5, No. 2 entitled CALC.FIX. Your idea of presenting a subscription to your children's school is an excellent one. We hope that other readers also consider this as well, because of the difficulty that many schools have in finding enough money to purchase software for their students' use.

Using More Memory with 2nd Drive

Dear Sir:

After overcoming several component problems I was able to complete the installation of the second disk drive on my PCjr. All is well when I use my Junior as the standard 128K machine, but I have been unable to get the additional memory to work with the second drive. I can use one or the other, but not both at the same time.

My PCjr is up to 256K. Can you tell me what code I have to change in order to take advantage of the additional memory and the second disk drive?

Thank you for your time and consideration.
David J. Calabretta
Havertown, PA 19083

David, it's difficult to give you an answer to your question without knowing which manufacturer's memory add-on you are using with the PCjr. But most manufacturers do supply some software that allows configuration of the memory. In general, this software should be run after the system has been booted as a dual-disk system and before application software is run. For related information see the letter entitled "More on Adding a Drive to PCjr" on page 11.

TI Wiring Diagrams Available

Dear Sir:

Reynaldo Rivera (Letters to the Editor, HCM Vol. 5, No. 1) can obtain wiring diagrams for his TI-99/4A from:

Howard W. Sams and Co., Inc.
4300 West 62nd Street
P. O. Box 7092
Indianapolis, IN 46206 U.S.A.

Their toll-free telephone number for orders is (800) 428-7267. Ask for SAMS COMPUTERFACTS-of-the-month Set #CF1 Folder CC2 for the TI-99/4A Model PHC004A.

This folder contains diagnostic procedures, parts lists and plenty of other information for true hardware junkies.

I found my copy of this folder at a local electronics store amid other SAMS folders on Apples, IBMs, Commodores, and various printers and disk drives. So SAMS should be able to provide technical information to any of your readers.

Keep up the good work on the magazine, and I would appreciate a review of Turbo Pascal for the TI-99/4A.

Jon Ruth
Boulder, CO 80303

That is really good information for the TI "hackers" who like to dig around inside the hardware. Thank you, Jon. By the way, Sams refers to the packet as Product #8901 for \$19.95. Regarding a review of Turbo Pascal for the TI-99/4A, it is our understanding that no such product exists. Turbo Pascal by Borland International runs on MS-DOS machines (such as the IBM PC, PCjr, and the Texas Instruments Professional Computer), CP/M, and

CP/M86 machines, but Borland does not presently plan on producing any products for the TI-99/4A home computer.

Ask and You Shall Receive

Dear Sir:

I have a Commodore 64 with disk drive, cassette, printer, and TV. We have various programs already—games, educational, home, and programming.

I bought the computer for all the above reasons, but mainly for educational use. I have three boys, ages 8, 10, and 12. I am interested in an educational program that deals with daily decisions of growing kids. Such decisions as are presented by word problems. But not ones like, if two trains are on the same track going a certain speed, when will they meet? I'd like one that deals in quantity problems. One that can combine math, measurement, choice, and logic.

I would appreciate it if you could let me know if there is such a program available.

You had a program in an earlier issue that dealt with a bills budget called "Cash Flow." Do you have a Commodore version of this program or something similar?

Dennis Rodgers
Bartonville, IL 61607

Dennis, first may we suggest that you turn to Geometrix in this issue and see what your boys think of that as an educational program. We feel pretty proud of it ourselves. In answer to your question regarding the cash flow program which first appeared in Vol. 2, No. 10, we do not at this time have a Commodore version of the program. But we do recommend that you look at the program, Budgetron, in this issue and see if it will satisfy your needs for controlling your cash flow.

Apple One-Liner Error?

Dear Sir:

I found an issue of your magazine by accident. I purchased an Apple IIc the day before Christmas in hopes of enriching my family's life. So far, I've had nothing but expensive lessons with very little interest from my family and a nagging wife. I was utterly shocked at the cost of the commercial software—\$58 to \$60 per item—that's crazy!

I purchased your magazine, Vol. 4, No. 5 and attempted the one-liner on page 34 for the Apple. All my kids were around me anxious to see me do something creative. When we went through the program twice, item for item and then ran the program, nothing happened. My screen showed # of sides, and if I pressed any key, I would get a SYNTAX ERROR. I could not remedy the program.

In spite of everything my family and I are anxious to become involved in computers.

Michael Wlodavczak
Bay City, MI 48706

Michael, the only keys that are allowed in that one-liner program are number keys. If you

Continued on next page

press any key other than a number key, you will get a syntax error and the program will stop. A one-line program just doesn't leave much room for error checking. By the way, take a good look at what you are holding in your hand, Michael. We think you'll soon discover that HCM's program content is the "quintessential software bargain" in the industry.

Using A Serial Printer on PCjr

Dear Sir:

I am writing in response to Ed Quenzer's letter in Vol. 5, No. 1. I too have moved from a TI-99/4A to an IBM PCjr and am using my Smith Corona TP-1 serial printer connected to the serial port on Junior. I initially had some problems with the cabling, but getting the port to work was not a problem. The answer lies in the DOS 2.1 program. The serial port on my system is opened with the following AUTOEXEC.BAT file that I have included on all my system disks:

```
MODE COM1:60,,,P
MODE LPT1:=COM1
DATE
TIME
```

This file configures the output to COM1 (the serial port) for the TP-1 printer, and redirects all output to LPT1 (the IBM printer port) from LPT1 to COM1, and then brings up the date and time prompts. Other than the cable, no other changes were necessary to get my system running.

I find that your magazine is ideal for a user like myself, and it has been very useful to me in making the transition from one system to another. With the demise of the PCjr-specific magazines, there is no other source of good, pertinent information concerning problems like the above. Keep it up.

Gordon K. Jones
Lake Forest, IL 60045

That's good information to pass along to other folks who are trying to use the serial port for a printer. Thank you very much, Gordon. Just an added note: the first line which states MODE COM1:60,,,P contains the specifications for the set-up of the serial port. The 60 represents a baud-rate of 600 (the first two digits are the only significant ones in this parameter). The commas make the operating system use the defaulted parameters for the parity (E), the number of databits (7), and the number of stop-bits (1). The P at the end of the statement indicates that the asynchronous adapter is used for a serial printer. Consult your PC-DOS manual to learn more about how the MODE command can be used to redirect printer output to RS232 devices.

Locating Commodore Printer Ribbons

Dear Sir:

In your Vol. 4, No. 5 issue, a Mr. Hugh Valliat wrote on the problems of the replacement of the ribbon for the Commodore printer, MPS 801.

Radio Shack DMP-110, Part No. 26-1283 is a perfect replacement ribbon for the printer.

Thought I would pass this information along. In closing, I would like to say, keep up the good work—HCM is the best computer magazine out today.

Joseph J. Forte
Philadelphia, PA

That's great information, Joseph. We've also noticed that Radio Shack carries many useful items for the home computer user, including spare parts for the TI-99/4A home computer.

Multiplan Solves Again

Dear Sir:

Your letter published in Vol. 4, No. 5 from John R. Johan asked for software that would look up data in tables. I too have done machine tool estimating, so I know what he wants. Microsoft Multiplan has a table LOOKUP function which I use on the C-64 and the TI-99/4 for estimating income taxes. Multiplan is excellent for estimating, if the jobs are repetitive enough and do not vary in format. One could make up a spreadsheet for each of 5-10 different situations. I entered a tax table on the spreadsheet, and positioned it where it will not print out. When required, values are looked up in it and included in the calculation.

The disadvantage of this is that if the table is large so that the percentage of free space figure in the bottom right of the screen is less than 70%, the program execution slows annoyingly. The solution is to run Multiplan on a computer with larger memory, like the IBM PC or PCjr.

Another, more cumbersome possibility is to have the tables on external worksheet files with each entry given a name like INFO12, INFO13, etc., on external sheets with names such as POWER, COST, and DISCOUNT. You have to manually enter the external COPY command unless you establish permanent links. Data in permanent links is loaded each time a worksheet is loaded, so you wouldn't want to load all of them on your active sheet or it would fill up your memory.

In summary, Multiplan will do the job if each estimate is not too unique in format and the tables are not too large.

Cal Lamoreaux
Shelbyville, MI 49344

Very interesting! Thanks, Cal. Multiplan—and one of our knowledgeable readers—comes through again with another solution to a sticky problem.

Thinkjetting the TI Home Computer

Dear Sir:

I read with interest your review of the HP Thinkjet printer in the Vol. 5, No. 1 issue of HCM. Having purchased a Thinkjet six months ago at a discount through my company's buyer plan, I am thoroughly satisfied with its performance. As you correctly reported, use of the special short-fiber paper is a must. In my opinion, its small footprint (9" x 11"), speed, quiet operation, and quality print (11x12 dot matrix) make the Thinkjet perfect for home use and well worth the price.

I did, however, experience a little difficulty interfacing the Thinkjet to my TI-99/4A. The solution turned out to be a parallel cable with special electronics—the same required for interfacing a Smith Corona TP-1 to the 99/4A. I also found that I had to open a file with a fixed 81 length in order to stop the Thinkjet from splitting and printing my text on two lines. I hope this information is of some help to other Thinkjet-99/4A owners—it took me two cables and two weeks to solve.

B. Caldwell
White Oak, PA 15131

Several of our readers have written expressing the same problem that you uncovered in interfacing the Thinkjet to the TI-99/4A. We are therefore very happy to publish your solution.

BASIC Understanding on the Apple

Dear Sir:

We have had and enjoyed our Apple IIe for over a year now, and while my older son and I live with the word processing and my husband values his VisiCalc, I was delighted with your latest issue of HCM (Vol. 4, No. 5). I know just enough BASIC to understand what is going on, but I certainly don't have the know-how to create long, involved programs. So I typed in your "Division Tutor" and my daughter (fourth grader) was delighted—it was perfect for her. We have also "made" a quiz and "taken" it with equal success. [See "Quiz Construction Set" in HCM Vol.4 No.5.—Ed.] I want to thank you for remembering that computers can do something else besides balance budgets. Please keep up your imaginative programming geared toward the "at home" market.

Mrs. Katharine Kohudic
Sumerduck, VA 22742

Thank you, Katharine. You can be assured that we will continue to maintain a balanced editorial and program content for home users.

A Tough Act to Follow . . .

Dear Sir:

As the owner of an IBM PCjr, I am pleased to see that your magazine is providing useful how-to articles for this home computer, such as that dealing with the addition of a second disk drive. Now that you have supplied PCjr owners with a cost-effective method of overcoming the single-drive limitation, I would like to suggest that you address two other areas in regard to expanding the PCjr's capabilities.

First, is it possible to expand the PCjr's memory by replacing any of the banks of 64K chips inside the enhanced PCjr or on the IBM memory expansion "sidecars" with 256K chips? These chips have decreased significantly in price in the past few months, and their use would result in a lower average cost per K of RAM relative to the alternative of multiple-expansion sidecars. As this modification (particularly the internal substitution which superficially appears the most attractive) is likely to require some changes to both memory circuit

boards and power supply, it would appear to be a good candidate for a separate how-to article.

Second, can you suggest any alternative uses for the modem slot inside the PCjr? As the serial port and an external modem can be used for communications, it might be more cost-effective to utilize the internal slot for other purposes.

Thank you for your assistance.

Paul E. Tubb
North Vancouver, BC, CANADA

Challenges, challenges . . . always challenges. Thanks for the ideas, Paul. While we're investigating, perhaps some of our readers can share ideas on how to accomplish these projects. Has anyone actually produced a homebrew memory addition for PCjr? Perhaps they have actually used the external modem slot to accomplish this sort of thing. Let's hear from all you PCjr hardware hackers.

TI Printer With C-64 Revisited

Dear Sir:

A Vol. 4, No. 4 *Home Computer Magazine* Letter to the Editor mentioned "Using TI Printer With C-64." Can you tell me how to connect this system or set it up?

I have a TI printer, C-64, and Cardco + G parallel adapter. Will this setup print Commodore graphics?

Thank you for your help.

Craig Clark
Gresham, OR 97030

First, Craig, you must remove the top of your TI printer and take out the serial interface card, then replace the cover carefully. Also, you should check to see that the DIP switches in your printer, and in the Cardco adapter, are set according to the specs given in the appendix of the manual that comes with it. Now, the parallel port located on the left-side rear can be connected directly to the Cardco parallel adapter and, yes indeed, you can print the Commodore graphics on that TI printer.

Following the KISS Principle

Dear Sir:

Regarding your "Apple Seedlings" Dating Game in Vol. 5, No. 1, remember to KISS while dating. Anders Nereim was on the right track using powers of two, but lines 420 through 460 may be replaced by:

420 NN = YR*512 + MM*32 + DD

430 A = INT(NN/256)

440 B = NN - A*256

450 DAY = A: YEAR = B

Line 210 no longer needs the DIM statement.

Elsewhere in the same issue, the need for a date "ordinal" in MMDDYY format was expressed. Here's a TI Extended BASIC function for such a problem:

```
10 DEF DATE = 365*YY + DD + 31*
  (MM - 1) + ((MM > 2)*(INT(.4*MM +
  2.3))) + INT((YY + (MM < 3))/4) - INT
  (.75*(INT((YY + (MM < 3))/100 + 1)))
```

Nice things happen after this. . .

If your program places values in DD, MM and YY, then the day of week (Sat., Sun., . . . etc.)

is indexed by D when you execute the following:
100 D1 = DATE/7 :: D = INT(7*(D1-INT(D1)) + .5)

Warren A. Hall
New Hartford, NY 13413

Thanks, Warren, for the information on the "ordinal" function—it works great. However, we found that your Apple Seedling update must be overlooking something. The variable names chosen above do not agree with those in the program. Plus, even if they are changed to agree, it still doesn't POKE the proper values into the ProDOS memory areas. Any Apple programmers out there who have been able to shorten the Set The Date program?

Snap-Calc Columns Do Not Align

Dear Sir:

When I received my first issue of *Home Computer Magazine*, (August 1984), I laboriously typed in the IBM PCjr SNAP-CALC program. I loved every minute of it and it worked to my satisfaction. It is carrying many of the programs I needed it for, including 1984 IRS taxes.

When Vol. 4, No. 5 issue came, I updated the program and was very pleased that it also worked very well. But, as I started to print my output, I saw that I have lost the relationship of the column numbers to the columns themselves. The only difference that I can find in my copying of the lines of the update is that line 890 calls for a one-bar straight vertical line, which I do not have on the Junior, and I used the vertical line, which was in the original printing.

Please tell me where the error is so that I can get back the original relationship of column numbers to the columns themselves. If you have sent out a disk with the update on it, I did not get one as of this date. Thank you very much. It's a great program.

Edwin C. Mutzer
Bradenton, FL 33507

That one-bar straight vertical line is accessed by simultaneously pressing the [ALT] key and the [(bracket/brace) key at the same time. Once this is entered correctly, you should have no problem at all, Edwin.

Clones Will Not Run ProDOS

Dear Sir:

I recently purchased a disk (Vol. 4, No. 4) for the Apple II computer. I have a Franklin Ace 1200 computer. Since I do not have ProDOS on my machine, how can I transfer all the DOS 3.3 programs to another disk? Right now the disk is useless. I can't boot it since ProDOS seems to cover every program on the disk. Any suggestions?

Is there any way, since you are going to use ProDOS, whereby two disks for the Apple Family can be produced and sold (one with ProDOS and one without)?

Robert Hose
Stockton, CA 95209

Apple ProDOS is set up to work with an Apple II computer with 64K of memory and the Applesoft BASIC language in ROM. An Ap-

ple IIe with two disk drives and ProDOS utilities can be used to transfer ProDOS files to a DOS 3.3-formatted disk. Sorry, but it is impractical at this time for us to produce two Apple formats.

Building an MS-DOS Boot Disk

Dear Sir:

The first thing I would like to say about your magazine is that it is by far the best magazine I have read. The programs are very good, the articles interesting, and the reviews have helped me purchase some good software for my PCjr. I have written in to other computer magazines, but to no avail. My questions were not answered. I have been doing some intense programming lately and have run across a problem that is hindering me. How can you insert a disk after a system reset and get a designated program to run by itself? It's like when you turn on your computer and don't want to load in DOS, switch disks, and then load in the next program. I just want it to run as soon as the computer's drive is activated.

Thomas Dahbura
Hagerstown, MD 21740

To boot your own programs automatically upon system power-up, Thomas, you must put it on a "bootable" disk. You can create this disk from a blank disk by placing the MS-DOS system master disk in the drive and using the following MSDOS command: FORMAT A:/S. The operating system will then prompt you to put an unformatted disk in the drive and will FORMAT it for you. Once you have done this, copy the program file onto this disk using the MS-DOS COPY command. Finally, create an AUTOEXEC.BAT batch file on this same disk with the name of the program file you wish to boot. See the Home Computer Tech Note for IBM in Vol 4, No. 3 for details on Batch files. The next time you start the system with this disk in the A: drive, your program will boot automatically.

Use More Computer ID's in HCM

Dear Sir:

May I suggest you extend identification symbols of the computers applicable when writing the "Home Computer Product News?"

Also, on the Letters to the Editor, I feel that most readers are only interested in reading about what applies to their equipment. I find I waste time looking for items that concern my equipment.

Donald H. Farr
Coquille, OR 97423

Yes, Donald, we'll consider what you say regarding the Product News column. As to the Letters to the Editor, we feel that the cross-pollination of ideas is very worthwhile, and we receive a lot of positive reader comments indicating that they enjoy reading about other user's experiences with different machines.

HCM

BUDGETRON

by Scott Williams
and the HCM Staff

With this program in hand, the only hard part about making and following a budget is resisting the temptation to spend . . .

This year you've resolved to become more practical and start planning and saving early for your vacation. The planning part is easy—Bermuda in late fall for two weeks. Being practical about saving is more difficult. If you could make up a budget that would reasonably limit your expenses and allow you to save \$1700 between May and October, you could be on the beach sipping tropical drinks the first week of November. Plus, when you returned you could adjust your evidently successful budget to plan for new purchases, or relax it and enjoy a more liberal spending style.

Budgetron is just the tool to help you easily set up a custom home budget that allows you to estimate and track your monthly expenses. You may then multiply the net savings (or net losses) of one month to extrapolate your budgeting results over a series of months. Graphs and reports of your budget results can be displayed in a variety of ways.

Budgeted expenses may be listed in up to 16 of your own categories. For each category, you must enter a value which is the maximum amount of money that you wish to spend each month for items in that category. Actual expenses can then be itemized under each category and totaled for comparison against your budgeted values. Category 1 is used exclusively for income. (See Figure 1 for our sample budget. Actual expenses are indented.)

The Main Menu

- 1) WORK WITH BUDGET
- 2) REPORTS
- 3) BAR CHART
- 4) FILES
- 5) EXIT PROGRAM

NOTE:

In this text, we've color-coded the hierarchy of menus and sub-menus for clarity.

1) WORK WITH BUDGET

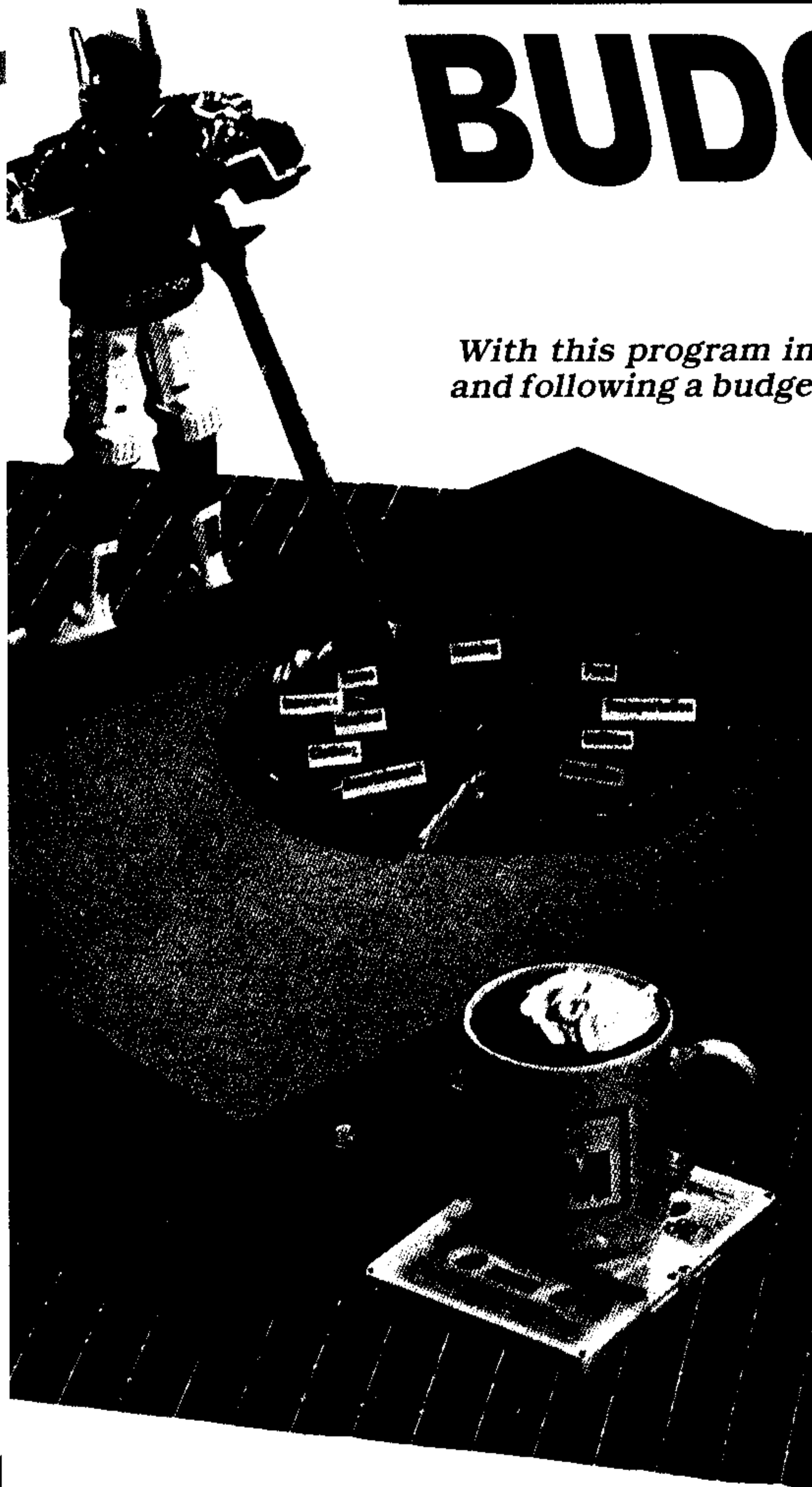
This is the first option that you will need to choose to set up a new budget. Selecting it will take you to the WORK WITH BUDGET menu screen:

- 1) BUDGET CATEGORIES
- 2) INCOME AND EXPENSES
- 3) EXIT

Budget Categories

You should select this option first to set up your budget categories. Selecting it will display this next menu screen:

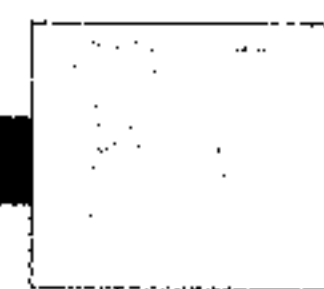
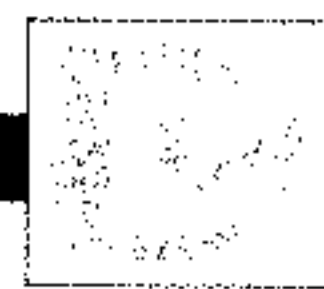
- 1) INITIALIZE BUDGET
- 2) ADD A CATEGORY
- 3) CHANGE A CATEGORY
- 4) EXIT



2) HOUSE PAYMENT	\$	375
3) UTILITIES	\$	300
4) PHONE	\$	125
5) GAS	\$	75
6) INSURANCE	\$	75
7) ENTERTAINMENT	\$	250
8) MISC	\$	250

PRESS ← TO CONTINUE

The Bar Chart option from the IBM version of Budgetron shows a comparison of budgeted (green) and actual (orange) expenses for eight categories of a monthly budget.



Initialize Budget

If you already have a budget in memory and would like to start over again with a new budget, select this suboption. It will erase all of the existing budget categories, leaving the program ready for a new set. The actual-expense items are not changed by this option. If you wish to also erase the expense items in the budget, you must use the Initialize Items option under Income and Expenses. Do not use Initialize Budget if you don't want to lose whatever information is in memory. You will be asked to confirm your selection of this option before it executes.

Add a Category

By selecting this suboption, you will see a list of existing categories, if there are any, and will be asked to enter a new category description. If there are more categories than the program can display on one screen, the program prompts you to either press the (SPACE BAR) to view more categories, or press (ENTER) or (RETURN) to get the prompt for entering new categories. After entering a description for the new category, you will be asked to enter its budget value. Remember, this budget value is the maximum amount that you have decided to spend on items in this category for one month. After entering the budget value, you will be asked if you would like to add another category. If so, press Y and the categories will be listed again.

Change Category

After selecting this suboption you will see a list of the categories you have created so far. Select the category that you wish to change by entering its number. You will then be prompted to re-enter both the category description and its value. If you attempt to change category 1 (INCOME), however, you will be allowed to change only the value. You can't change the income category description.

Income and Expenses

Select this option from the WORK WITH BUDGET menu when you wish to work with actual expense or income items in the budget. A menu screen will display the following options:

- 1) INITIALIZE ITEMS
- 2) ADD ITEMS
- 3) CHANGE ITEMS
- 4) EXIT

Initialize Items

As mentioned earlier, you should use this suboption only if you want to erase all expense and income entries from the budget so that a new budget can be started. This option is generally exercised in conjunction with the Initialized Budget option, but it can be used alone.

Add Items

This suboption will allow you to enter your actual expenses under each category, or an income value in category 1. You may enter as many items under each category as your system's memory will allow.

Select a category from the list displayed by entering the number beside it. You will then be asked to enter

a description for the item. After entering a description, you will be asked for the item's value. For example, under our sample budget's UTILITIES category, we entered the description PHONE and a value of \$60 when we paid our phone bill. After entering a value, you will be asked whether you would like to add another item. If you reply by entering Y, the list of categories will be shown again for your selection. Otherwise, you will be taken back to the main menu.

Change Items

A list of all of your categories will appear when you choose this suboption, though they may not all fit on

FIGURE 1

Sample Monthly Budget

\$\$ INCOME	1500	
PAYCHECK		1500
		<hr/> 1500
HOUSE PAYMENT	400	
HOUSE		400
		<hr/> 400
UTILITIES	240	
PHONE		60
ELECTRIC		45
WATER		35
GAS		100
		<hr/> 240
INSURANCE	100	
CAR		40
HOUSE		60
		<hr/> 100
CABLE TV	25	
BILL		25
		<hr/> 25
FOOD	195	
GROCERIES		75
DINE OUT		100
		<hr/> 175
MEDICAL	40	
DENTIST		40
		<hr/> 40
AUTO	75	
GAS		50
MAINTENANCE		60
		<hr/> 110
ENTERTAINMENT	51	
NEWSPAPERS		7
MAGAZINES		12
MOVIES		12
MISC.		20
		<hr/> 51
CLOTHES	50	
JEANS & SHIRT		50
		<hr/> 50
MISC. EXPENSES	40	
HAIRCUT		18
		<hr/> 18

one screen. Press the (SPACE BAR) to see more, (ENTER) or (RETURN) to select an item to change. When all of the items have been listed (or if you change an item description or value), you will be asked whether you would like to make another change.

2) REPORTS

Selecting this option will take you to another menu screen to select the report type:

- 1) ALL CATEGORIES AND ITEMS
- 2) MONTHLY SUMMARY
- 3) PRINT CHART
- 4) EXIT

These report options allow you to print to the screen or to a printer. When the screen is used to output a report, you are occasionally asked to press (ENTER) or (RETURN) to continue to the next screen.

All Categories and Items

This report will generate a formatted listing of every category, and all expenses or income items assigned to each category. Below each category's list of itemized values is the total value of expenses in that category. That is the number which should be compared to the value next to the category description on the first line of each category listing. This option is illustrated in Figure 1. A Monthly Summary Report is always printed following this report.

Monthly Summary Report

This report is printed at the end of the other reports, but it can also be printed by itself with this option. This is a short report which prints the total budgeted amount and total expenses. They are compared to the income total, and the difference is calculated. As you can see in Figure 2, our income exceeds our expenses by \$291 in this first month, and we actually managed to save 7 more dollars from what we had budgeted. At this rate, in six months we will have saved \$1746! By sticking to our budget, we can count on spending early November in Bermuda.

FIGURE 2

Budget Summary

BUDGET INCOME:	1500
ACTUAL INCOME:	1500
ACTUAL EXPENSES:	1209
CASH BALANCE:	291
BUDGETED EXPENSES:	1216
DEVIATION FROM BUDGETED EXPENSES:	7

Print Chart

This option will print a simple chart for each category. The chart is printed horizontally, with the letter B repeated in a bar to indicate budgeted amounts. If the expenses for a category are within the budget, an E is printed in bar form for the expenses bar. Otherwise, if the expenses are more than what was budgeted, asterisks are used for the bar.

3) BAR CHART

This option allows you to visually evaluate how well you are doing with your budget. After a brief pause while the computer calculates the totals, a bar graph of your budget and expenditures will be displayed for each category. The top bar for each category indicates how much you allotted for it in your budget. The bottom bar shows your actual expenses for the month.

4) FILES

Here you may load and save your budget information to disk (with all 4 versions) or tape (Commodore 64 and TI-99/A versions only). The steps involved in saving or loading your data file will vary from machine to machine. Check your system's manual.

5) EXIT PROGRAM

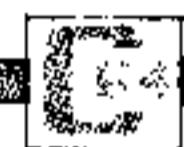
Before the program halts, you will be given one last chance to return to the main menu. A message will warn you that any data in memory will be lost if you stop the program, and you will be asked to reaffirm that you want to exit. Press Y to stop the program, or N to return to the main menu and continue without losing data.



The Apple II Family computers have a large number of ROM (Read Only Memory) routines which can be accessed from BASIC with the CALL command. Two of these routines are used in *Budgetron*. The HOME command allows you to clear the entire screen, but if you want your program to clear a single line from the screen, then CALL -868 is just what the doctor ordered. This accesses a machine-language routine contained in the system's ROM. The routine will clear the current screen line of text from the cursor position to the right edge of the screen. This is most useful when returning to an INPUT statement because the first entry was illegal. On the IIe and IIc you can do this at your keyboard by pressing (Esc) E.

A second routine, CALL -958 will clear all of the screen to the right and below the current cursor position. All text above, and to the left of the cursor will not be affected. This command is extremely useful, and it saves a considerable amount of time and programming effort. If you have a data-entry area, or options displayed at the bottom of the screen, you can easily clear them without having to clear the entire screen.

A third call to -936 is not used in this program. This call will erase the entire screen just as the HOME command does.



The Commodore 64 version of this program utilizes a data-compaction algorithm to place in a single string all of the information for an item, such as its category, description, and value. Each string is made up of four parts. The first two parts occupy only one character each. The first part, category, is the category number with which the item is associated. It is a single character. Subtracting 100 from the ASCII code of this character will yield the category number. The second byte of the string contains the length of the description. Subtracting 200 from the ASCII code of this character yields the number of characters in the description. The value of the item follows the description in the string. The length of the description—derived from the second character in the string—is then used to locate where in the string the value for the item resides.

Format for string:

Category # / Description Length / Description / Value

The items are all stored in the A\$() array, and the data for the array is constructed in lines 1480 through 1500. The variables used in the construction are as follows: ZZ = item number; \$\$ = item description; C = category; V = item value.

The reporting features of the IBM PC and PCjr version are very handy. You can either print a report to the screen for a quick review, or get a hard copy of it on the printer. Fortunately this can all be handled by the same print statements. The IBM PC and PCjr allow you to open a file which outputs to the screen just as if it were a printer. However, this does not work if you want to use the LPRINT statement to output data to the system's printer. The program needs to open a file with either SCRN: to send output to the screen, or LPT1: to send it to the printer. After opening the file, you can print and the system doesn't care whether the data is going to the printer or to the screen.

Ever notice the arrow used in many of the continue prompts that stand for the (ENTER) key? It is constructed from two of the IBM's standard graphics characters. In this program, we have assigned these two characters to the variable ENT\$.

```
ENT$ = CHR$(17) + CHR$(217)
```

This presents a problem on the PC, however, because the graphics characters with ASCII values above 127 are not available when using graphics screens 1 or 2 (this may vary depending on the type of graphics card you are using). These characters are always available on the PCjr. To solve the PC problem, we have designed a graphics string which is drawn with the DRAW command. Whenever the shape is needed (only on the graphics bar-chart screen in this program), you can simply draw it on the screen. The string is defined in line 1390 and then used in line 1630. ENTDR\$ contains the graphics string.

When developing a program which prints a report of some form—such as this one for the TI-99/4A—you may often have the need to print that same form to the screen as well. That way you can get a quick review of the data before committing to a hard copy. In addition, if you are developing the software to be marketed, or if it is intended for use by other people, you may want to attract the largest audience possible. But many potential software buyers may not own a printer, and would not reap the benefits of the printed report. This means you would have to write a completely different section in the program to handle screen output as well as printer output.

A little-used and seldom-discussed function of the TI-99/4A computer may be able to save you hours of frustration and headaches: Just open a file with channel 0. When the report gets ready to print, you can provide an option which asks for output to the printer or the screen. You may then set a variable to 0 if the screen is selected, or to some other file number if the printer is selected. You will need a little logic to branch around any OPEN or CLOSE statements if the channel variable is set to 0 (this is illegal for those statements). In this program the variable F is used. When you print to the file, use the variable after the # symbol instead of a number:

```
PRINT #F: "THIS GOES TO THE SCREEN IF F=0"
```

HCM

For your key-in listings, see HCM PROGRAM LISTINGS Contents.

Budgetron (Apple II Family) Explanation of the Program

Line Nos.	Explanation
100-200	Program header.
210-610	Initialization and main menu.
620-1780	Menu selection screens.
1790-2400	Work with the budget.
2410-3080	Work with items.
3090-3150	Select a category.
3160-3680	Print reports.
3690-3780	Display categories.
3790-4090	Number-entry routine.
4100-4220	Calculate totals.
4230-4310	Turn on the printer port.
4320-4410	Get input for menu selections.
4420-4590	Error routine.
4600-4770	Delete routines.
4780-4990	File-name-entry routine.

Budgetron (IBM PC & PCjr) Explanation of the Program

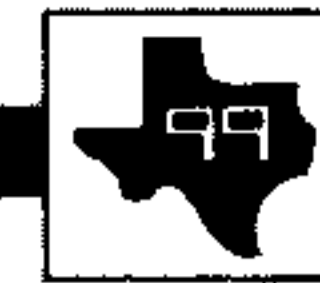
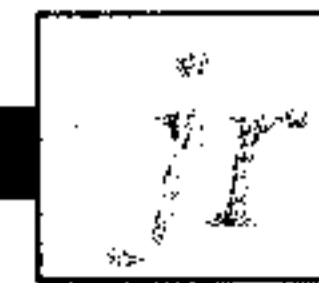
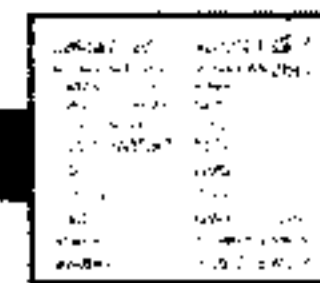
Line Nos.	Explanation
100-210	Program header.
220-310	Initialization and main menu.
320-490	Work with budget.
500-660	Work with items.
670-780	Category routines.
790-820	Display prompts for another.
830-1000	Change-items routine.
1010-1090	Delete routines.
1100-1650	Reports.
1660-1830	Files routines.
1840-2140	Misc.-input routines.
2150-2200	Print-categories routine.
2210-2230	Exit-program routine.
2240-2290	Data for menu screens.
2300-2360	Error routine.

Budgetron (C-64) Explanation of the Program

Line Nos.	Explanation
100-200	Program header.
210-360	Initialization and main menu.
370-650	Menu selection screens.
660-1070	Work with the budget.
1080-1540	Work with items.
1550-2030	File routines.
2040-2200	Display bar chart.
2210-2730	Report routines.
2740-2860	List-categories routine.
2870-2940	Data-entry routines.
2950-2960	Beep-and-pause routine.
2970-3020	Exit program.
3030-3310	Misc.-input routines.
3320-3330	Move-cursor routine.
3340-3460	Delete routines.

Budgetron (TI-99/4A) Explanation of the Program

Line Nos.	Explanation
100-200	Program header.
210-670	Initialization and main menu screens.
680-1460	Work with the budget.
1470-2250	Work with the items.
2260-2640	Display-the-chart routine.
2650-3530	Reports routines.
3540-3880	Load and save data routines.
3890-3970	Routine to initialize expenses.
3980-4010	Key-input routine.
4020-4160	Get totals for each category.
4170-4320	Validate numeric entries.
4330-4360	Time delay with a beep.
4370-4420	Routine to exit the program.
4430-4440	Graphics character data.
4450-4710	Delete categories and items.



Over-Reaction

by HCM Staff

A nuclear reactor is raging out of control; the world is on the brink of war—earthquakes, riots, and sabotage are the order of the day—and you are the one at the controls . . .

Over-Reaction is a game that simulates the operation of an electric-power-generating nuclear reactor. It's the player's job to keep the reactor operating smoothly, producing a steady supply of electricity at a frequency of 60 cycles per second (Hz). If the frequency deviates by more than 2 Hz (below 58 or above 62), the reactor automatically "trips" (shuts down), ending the game.

The reactor's power output is controlled by 24 fuel rods set up in a 4-row by 6-column grid; each rod is designated by a letter (except in the Apple version, which uses numbers). The grid is divided into four quadrants—each has a temperature gauge which monitors the average temperature of the quadrant. By pressing the proper key, you will push in a rod—this tends to raise the temperature around the rod. By pressing the same key with the shift key depressed, you will pull out the rod, lowering the temperature. The overall temperature of the entire grid controls the reactor's steam pressure, which drives a turbine to produce electricity.

Built-In Variations

As the steam pressure fluctuates, so does the frequency of the electricity you are producing. It is therefore your job to keep a constant steam-pressure level. If the temperature in one quadrant drops too low or increases too high, you will hear a warning buzz. If it goes outside of a safe range, it will cause an automatic shut-down or trip—costing you points and ending the game until the reactor can be repaired.

You must also deal with the effects caused by changes in load demand. If the load demand increases, it tends to pull down the system, and drop the frequency. If the frequency begins to rise unexpectedly, it probably means that the load demand has dropped, and you will need to decrease the overall temperature of the reactor to compensate. You will find that as in real life, the system won't react to your control immediately—it takes time for your actions to take effect.

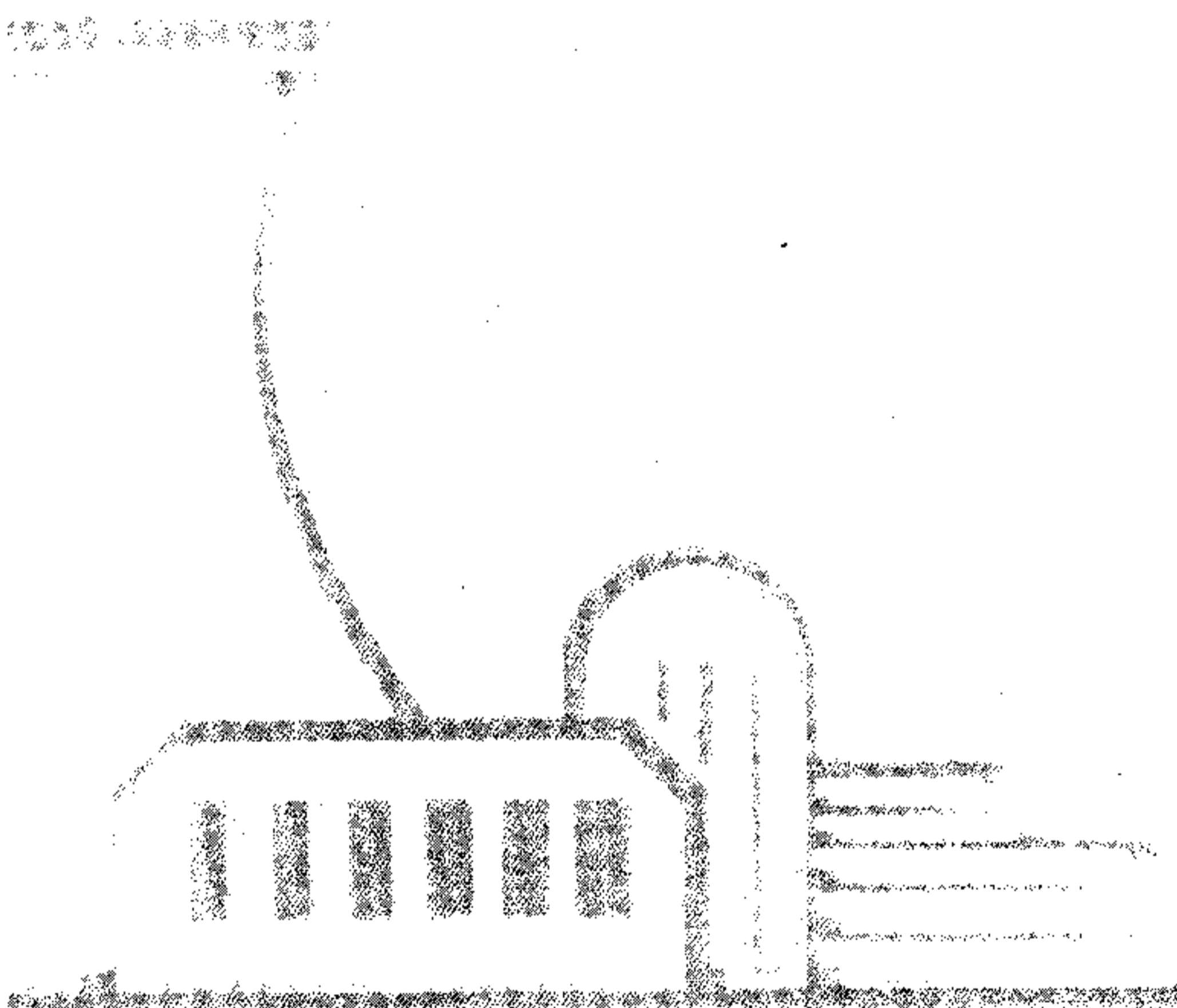


Photo 1. This screen from the Apple version shows the normal operating conditions of the reactor. Note the use of numbers rather than letters to indicate the control-rod drives.

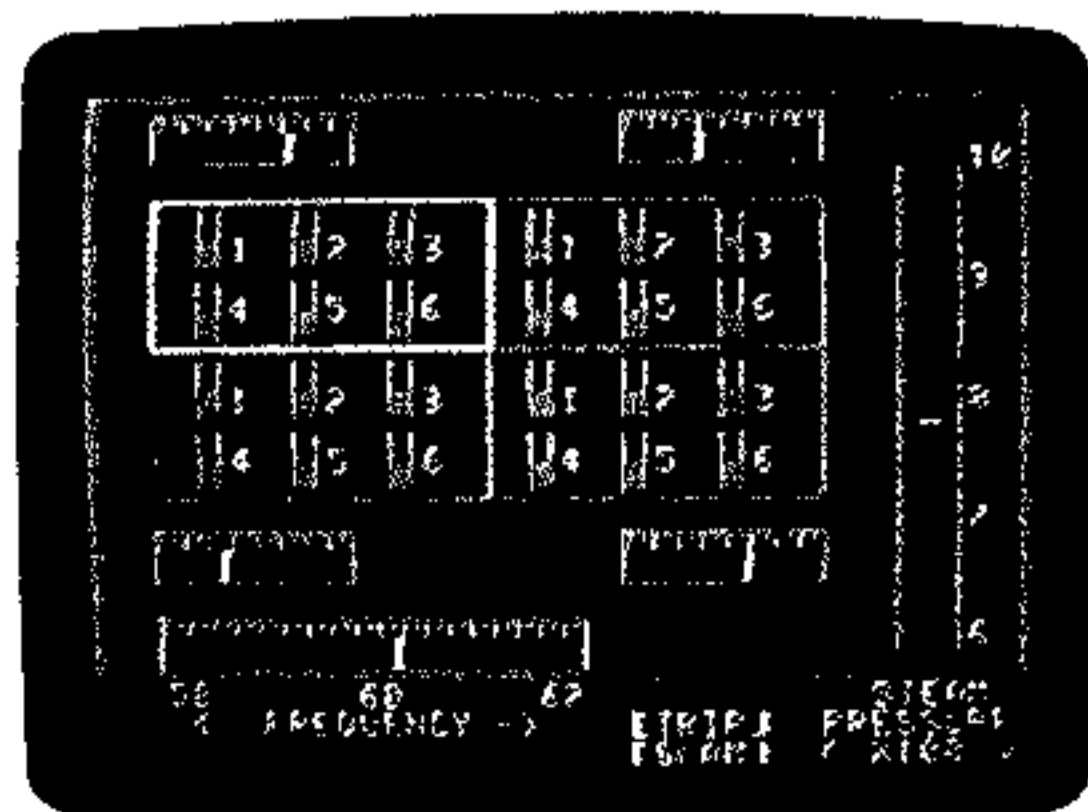


Photo 2. This screen from the C-64 version shows a broken temperature gauge. Notice the temperature variations in each quadrant. Try to maintain uniform temperatures.

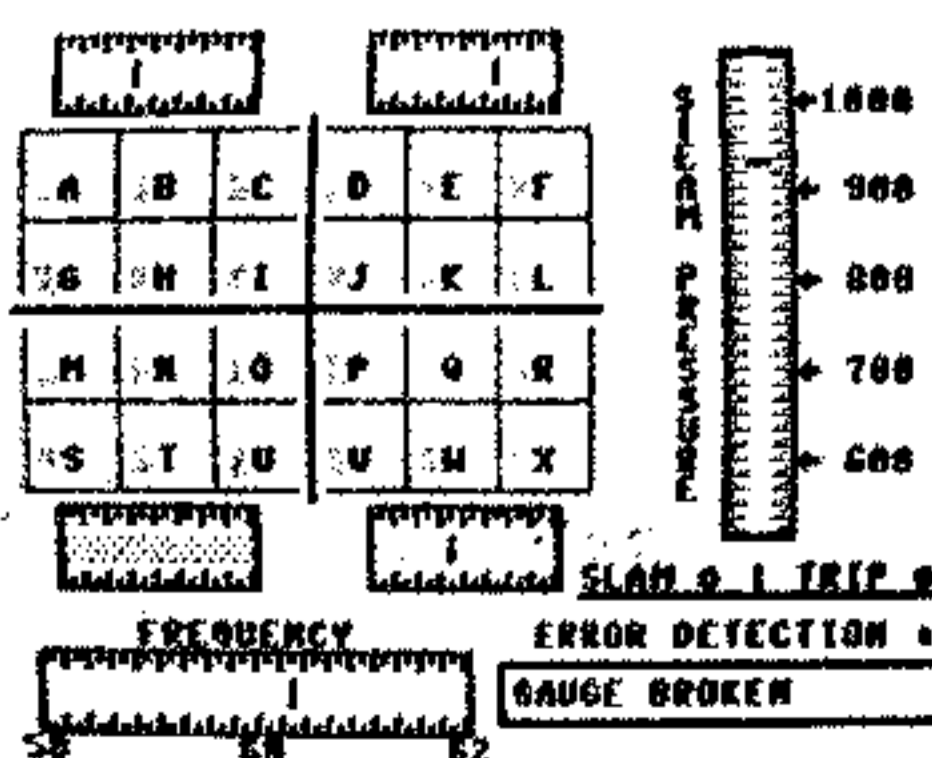
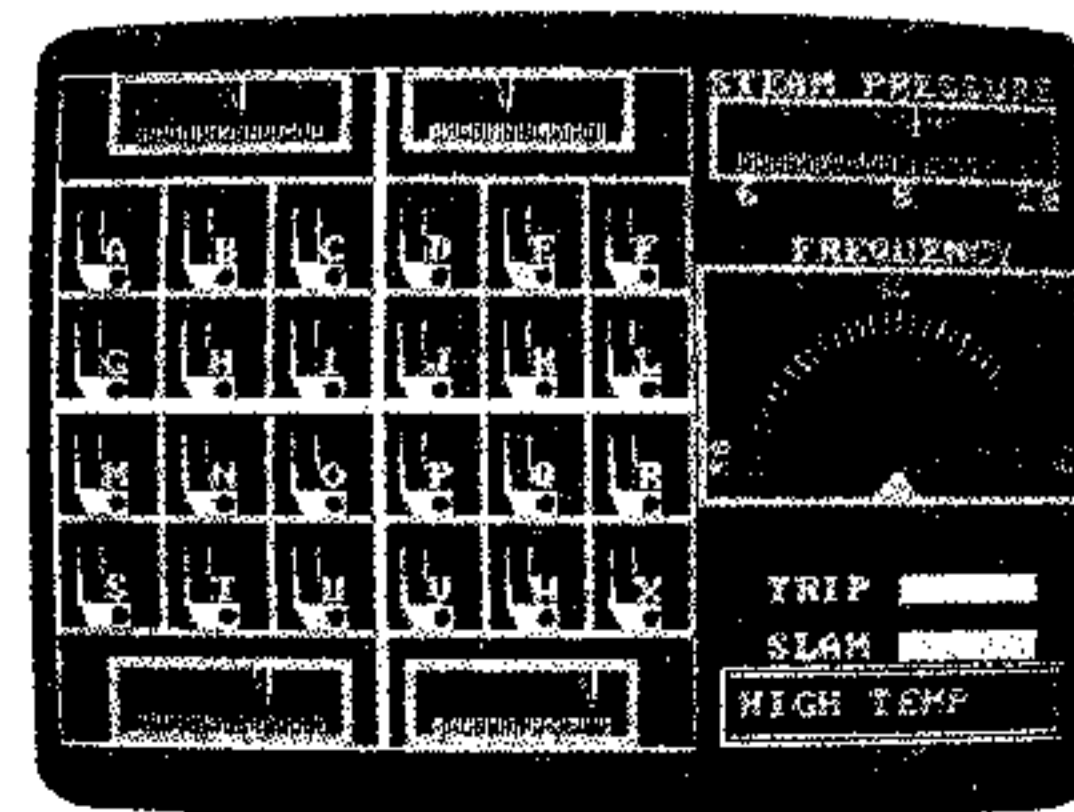


Photo 3. This screen from the IBM version shows a high temperature warning. Notice that the frequency gauge is dangerously high. Check the 4 quadrants for high temperatures and adjust the control rods accordingly.



Controlling the Power Output

The difficulties you will encounter vary according to skill level:

- 1) Normal operation
- 2) Faulty systems
- 3) Earthquake country
- 4) Armageddon

In the first level, you merely have to watch for minor temperature fluctuations and small changes in the load. This mode is best for training, so that you will be prepared to handle the higher levels. Experiment with raising and lowering the control rods—watch how long it takes for your action to cause the reactor to change. Once you feel like you can control the power output, you are ready to tackle the higher levels.

The second level introduces more problems.

The temperature here can fluctuate more radically, and the load can increase or decrease as well. Also, any one of your four temperature gauges can break at any time, making it impossible to monitor the temperature of a quadrant. Photo 4 displays an example of some fluctuations. When a warning sounds, you must be prepared to take corrective action where needed.

At level 3, you're faced with all of the problems that can occur in level 2, plus steam leaks caused by earthquake tremors. Here you'd have to increase the temperature to compensate. Photo 5 illustrates what can happen if you don't react fast enough to correct a problem. In this example, the overall temperature of the reactor dropped low enough that the frequency level fell below 58, causing the reactor to trip off-line.

Finally, at level 4, a nuclear war is imminent—riots, sabotage, and even more drastically fluctuating loads will plague you. At this level, be prepared to quickly make adjustments to changes that you see on the gauges. But remember, it takes a little time for the reactor to "react" to your manipulations of the control rods. Before you play the game, review *all* of the accompanying photo examples (regardless of which machine is indicated) to familiarize yourself with some of the different problems that can arise during the reactor's operation.

One final tip for being more successful at the control panel: learn to identify and listen for audible warnings—steam leaks, high temperature warnings, gauges breaking, etc. And, if you should realize that you have over-reacted, or that you just can't bring the frequency levels into the safe range and a meltdown is inevitable, fear not. Just push the slam button ((FUNCTION) 1 on all but Apple systems, where it is the (ESCAPE) key). This will keep your score from being lowered by a forced shutdown of the system.

Photo 4. This screen from the C-64 version shows power fluctuations—two temperature gauges are broken. Fluctuations will make the frequency level behave erratically, so try to anticipate "hot" spots.

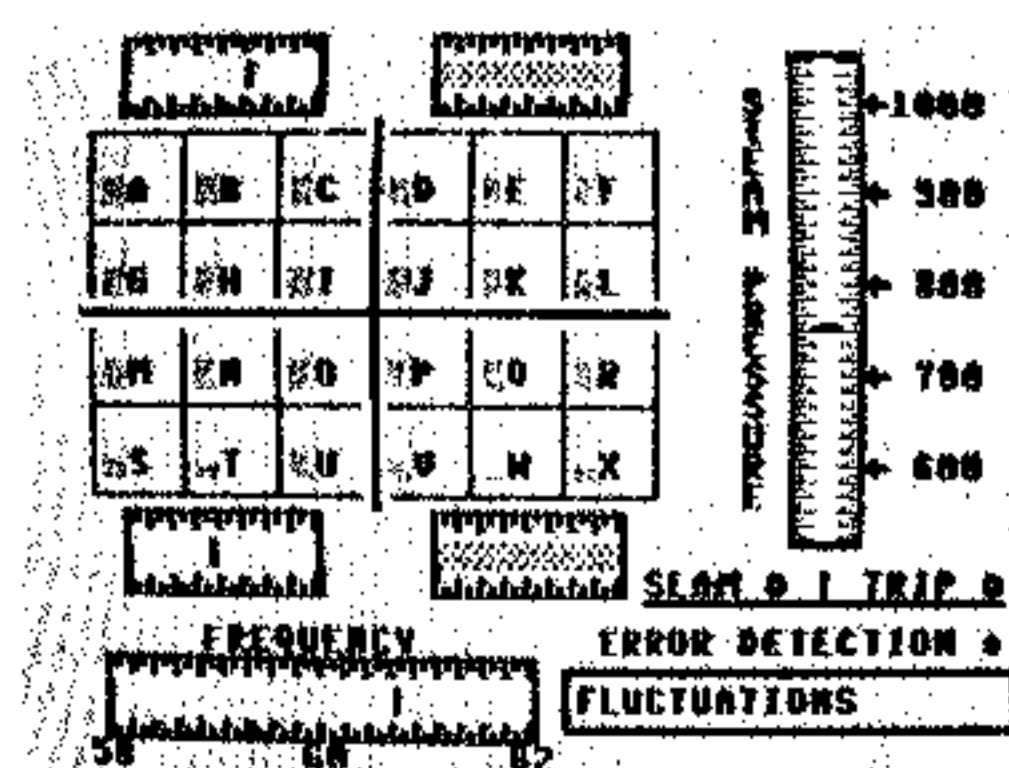


Photo 5. This screen from the IBM version displays a bad-frequency warning. The frequency level is extremely low; too low—the reactor has tripped. At this point, the game ends and you can try again next time.

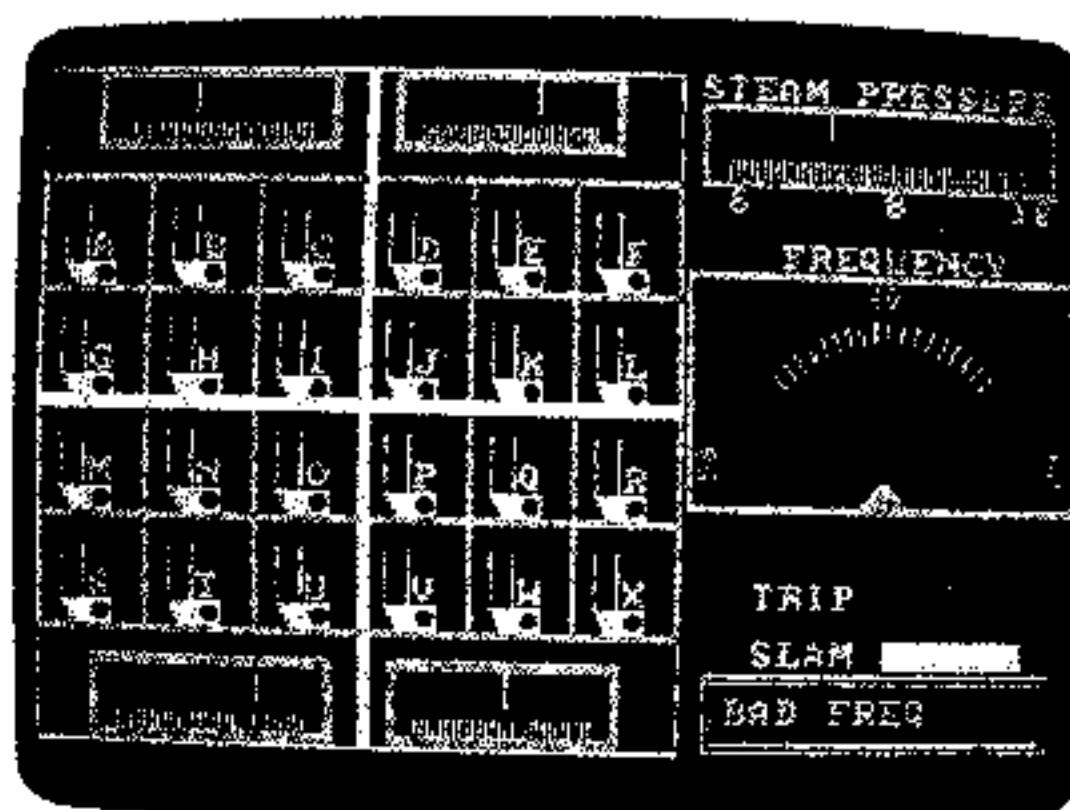
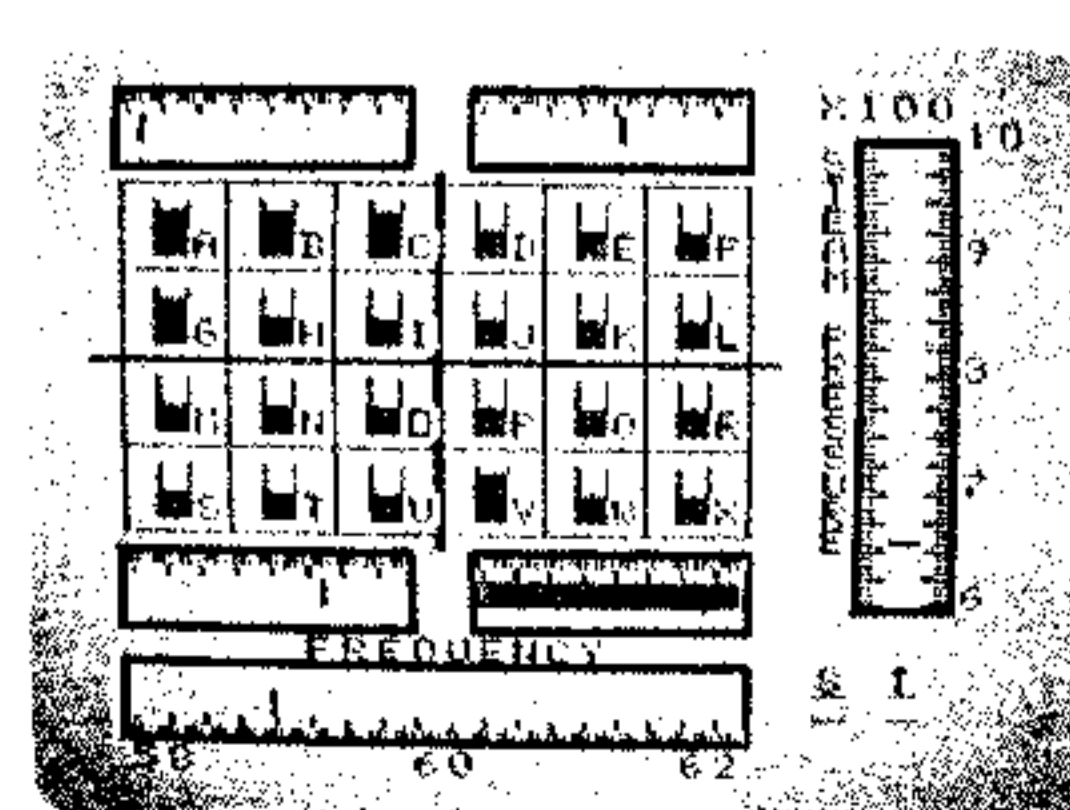


Photo 6. This screen from the TI version shows normal reactor operations, but notice the faulty temperature gauge. Try to keep the frequency as close to 60 Hz as possible.



The original, Apple IIe version of *Over-Reaction* was designed for use with shifted alpha characters (lowercase). Unfortunately, the Apple II+ does not have this capability. Thus, the Apple II Family version of this program is controlled in a slightly different manner. You can select a quadrant with the keys A, B, C, and D. The active quadrant will be highlighted by a color different from those not active; it will stay active until you select another quadrant. There are six control rods in each of four quadrants in the control panel. You can access the rods in the active quadrant by pressing the numbers 1 through 6. Pressing the unshifted numbers will push in the rod. If you press a shifted number, the rod will be pulled out.

"You will find that the system does not react immediately . . . it takes time for your actions to take effect."

numbers will push in the rod. If you press a shifted number, the rod will be pulled out.

The Commodore 64 contains an extremely versatile set of built-in graphics characters, and this program takes full advantage of them. The temperature, steam pressure, and output frequency meters use these characters to position their needles. Two strings are used to contain the desired characters: L\$ is used for the horizontal needles of the temperature and frequency meters; the string U\$ is used for the steam pressure meter. Each string contains all possible needle positions that can occur within a character. When the meter setting is evaluated, the position of the character desired is calculated and extracted with the MID\$ function. The character is then printed on the screen as the new meter position.

The IBM PC and PCjr version of *Over-Reaction* uses two methods to display the meter needles on the control panel. The temperature and steam pressure indicators both use a sliding-scale meter which moves horizontally, both left and right. The needle is displayed using the PUT command. This command allows you to display a graphics shape that was previously saved into an array using the GET command. The PSET option is

used with an area for the needle which extends to the left and right of the needle itself. In doing this, the needle will erase its old image every time it is moved. This method produces much less "flicker" than the default method of erasing the old image with the XOR option. The frequency meter looks like an old-fashioned analog meter, with a long needle sweeping across a semi-circular dial. The needle produced here is drawn with the LINE command. The program will redraw a new line only if a calculation shows that the needle position has changed. **NOTE:** the (CAPSLOCK) key *must not* be activated before and while you run the program. Also, do not hold down a key after the acceptance beep—it may cause the keyboard buffer to overflow, which may result in erratic program behavior.



The TI version of *Over-Reaction* takes advantage of the 99/4A's ability to redefine character graphics, pro-

ducing what appears to be a high-resolution motion of the indicator needles used in the the control panel's instruments. Only four characters were needed to define the horizontal meter needle, and four more for the vertical meter needle. A calculation determines which character is required for a given position in the meter. The character is then placed on the screen using the HCHAR command.

Eight characters were defined for the control rods. There are sixteen positions in each control rod, requiring that two characters be used. These algorithms can easily be employed in a multitude of applications where the relative position of a value is more important than the value itself. As you may know, it takes quite a while to display messages or numbers on the screen by using TI BASIC without scrolling the screen. Although this program doesn't burn up the track with speed, you will nevertheless be impressed with the amount of activity going on, and the speed with which the program executes.

Over-Reaction (Apple II Family) Explanation of the Program

Line Nos.	Explanation
100-200	Program header.
210-230	Relocate program above hi-res screen.
240-350	Initialize the program.
360-390	Get skill level.
400-570	Draw the control panel.
580-630	Initialize the variables.
640-790	Main control loop.
800-940	Select-problems routine.
950-1110	Start-of-game messages.
1120-1370	End-of-game messages.
1380-1420	Controlled shutdown—slam.
1430-1480	Control updating gauges.
1490-1520	Meltdown routine.
1530-1560	Update steam gauge.
1570-1600	Update frequency meter.
1610-1720	Scan keyboard to control the reactor.
1730-1780	Update rods & block temperature.
1790-1800	Draw a box routine.
1810-1820	Routine to flash the screen.
1830-1970	Program DATA.

Over-Reaction (IBM PC and PCjr) Explanation of the Program

Line Nos.	Explanation
100-220	Program header.
220-340	Initialize the program.
350-430	Main control loop.
440-470	Branch to problems for skill level.
480-500	Temperature meter malfunctioned.
510-520	Cooling system malfunctioned.
530	Control problems for levels 2 and 3.
540	Steam leak develops. Pressure drops.
550-600	Scan the keyboard. Reactor control.
610-650	Move rods in or out.
660-720	Update the block temperature meters.
730-790	Update steam pressure meter.
800-850	Update frequency meter.
860-1010	End-of-game messages.
1020-1150	Get skill level. Display messages.
1160-1310	Draw the control panel.
1320-1350	Routine to scan the keyboard.
1360-1400	Coordinate the rods.

Over-Reaction (C-64) Explanation of the Program

Line Nos.	Explanation
100-190	Program header.
200-390	Initialize program.
400-450	Skill-level option screen.
460-700	Start-of-game messages.
710-1210	Draw the reactor control panel.
1220-1250	Initialize variables for a new game.
1260-1390	Main control loop.
1400-1530	Determine types of problems.
1540-1620	Scan keyboard. Reactor control.
1630-1660	Position the reactor rod graphics.
1670-1740	Update the block temperature.
1750-1810	Trip and slam routines.
1820-1910	Update temperature gauges.
1920-1960	Update steam pressure meter.
1970-2000	Update frequency meter.
2010-2080	Error-condition routines.
2090-2440	End-of-game messages.
2450-2460	Move-cursor routine.
2470-2680	Program data.

Over-Reaction (TI-99/4A) Explanation of the Program

Line Nos.	Explanation
100-200	Program header.
210-310	Select skill level.
320-890	Initialize program.
900-1140	Main control loop.
1150-1410	Select problem routine.
1420-1650	Start-of-game messages. Determine skill level and problems.
1660-1980	End-of-game messages. Replay option.
1990-2090	Slam routine.
2100-2210	Update temperature meter.
2220-2290	Update steam-pressure meter.
2300-2370	Update frequency meter.
2380-2660	Key input routine-position rods.
2670-2770	Update block temperature.
2780-2800	Key-scan routine.
2810-2840	Routine to update graphics position.
2850-2980	Character graphics & color DATA.
2990	DATA for meter position array BP().
3000-3070	Screen display format DATA.

HCM

For your key-in listings, see HCM PROGRAM LISTINGS Contents.



TORPEDO ALLEY

by William K. Balthrop
HCM Staff

War has traveled to the far reaches of the Earth—even to the depths of the sea. How long will you survive in this underwater game of cat and mouse?

From deep in the tumbling sea, your submarine rises to peek above the surface. A single warship enters from the edge of your viewing field. As it approaches the vertical crosshair at the center of your scope, you issue the order: *Fire Torpedo!* You see the wake of the first shot climb along the center line, diminishing in the distance as it nears its deadly rendezvous. Finally, a flash of flame and following thunder announce the demise of yet another enemy vessel. But the next ship to appear zig-zags around your second torpedo, and swings around on a collision course as its guns pump out deadly salvos and depth charges are made ready. There's no time to waste—*Dive! Dive!*

Torpedo Alley is a game of timing and wits. The object of the game is to sink as many enemy ships as you can with your limited supply of torpedoes. But watch out—the enemy can shoot back. Fortunately, at any time you may submerge and take evasive actions.

But submerging has its drawbacks as well. Your sub is not fast when fully submerged, and the enemy ship drops depth charges. You must try and maneuver your sub around the charges or you will be blown-up. If you survive the depth charges, you can resurface and take another stab at the ships with your torpedoes.

Fire Away

Start the game by deciding whether to use joysticks. If you do use joysticks, the following actions will result:

Main Screen - Up Periscope
Left—Move sub to the left.
Right—Move sub to the right.
Down—Dive; down periscope.
Fire button—Fire torpedo.

Second Screen - Submerged
Left—Move left.
Right—Move right.
Down—Move down.
Up—Move up.

If you select the keyboard option, you can consult your machine's Control Capsule for the proper movement control keys.

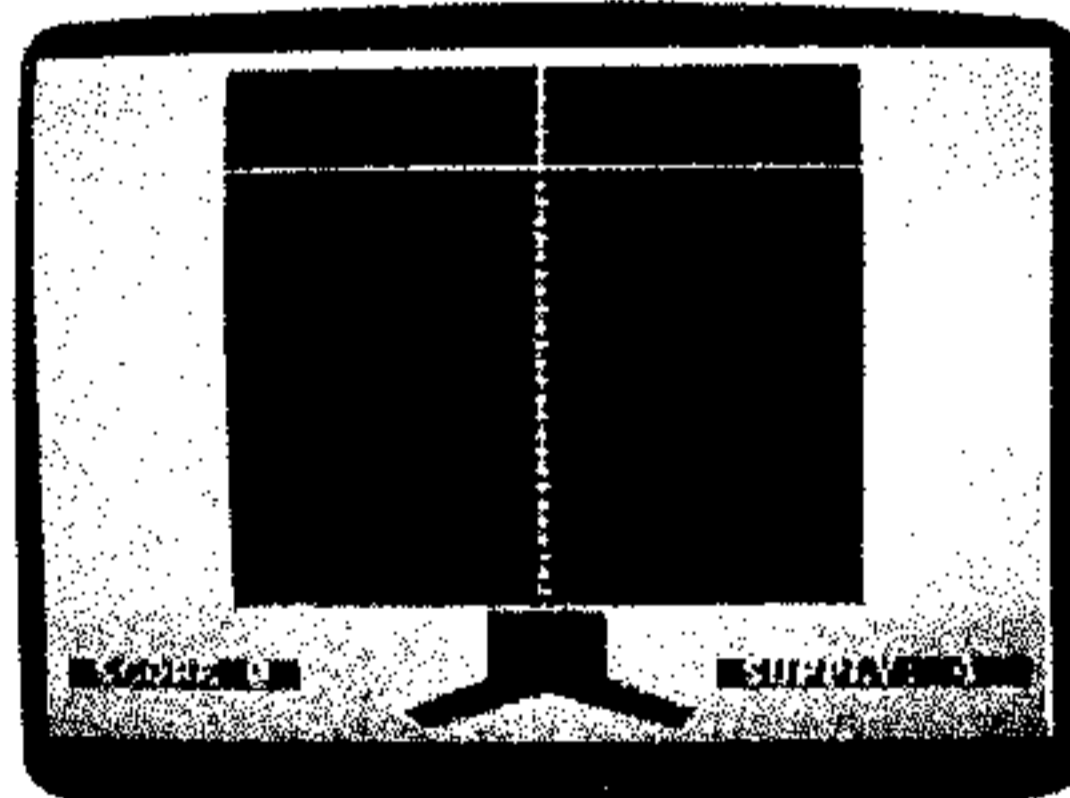
After you select the joystick or the keyboard, the screen will clear and the main game screen will appear. This screen displays the view of enemy waters as seen through your periscope. Occasionally, an enemy vessel will cross your path. You must try to judge the precise time to fire your torpedo, based on the ship's distance, and its direction of travel. The destroyers are aware that you are in the area, so most of them will be navigating a zig-zag pattern. This makes them a more difficult target because distance is harder to judge.

The enemy ships will fire at your sub if you allow them to pass beyond the center of your periscope. You can turn your sub either left or right to try and line up the ship by using either the keyboard controls or the joystick. If you are unable to get off a good shot before the enemy ship reaches the other side of your periscope, your only hope of avoiding a hit from its main armament is to dive.

When you dive, the screen will redraw, showing your sub diving

below the surface of the water. A ship will then start traversing the top, dropping depth charges. Resurfacing after out-maneuvering all of the depth charges is automatic; at this point, you needn't do anything. After you surface, the main screen will display again, and you will be ready to fire your torpedoes at the next ship.

Every time you are hit by the main armament from the enemy's ships, some of your torpedoes become damaged, lowering your supply. When your supply of torpedoes reaches zero, or when you are destroyed by depth charges, the game is over. Your score and percentage of hits versus shots taken will be displayed. You will then be given a chance to play another game.



Screen photo taken from C-64 version.

CONTROL CAPSULE Torpedo Alley

KEY	FUNCTION
<i>Surface Mode:</i>	
Left arrow	Move periscope left.
Right arrow	Move periscope right.
[ENTER]	Dive.
[Space bar]	Fire torpedo.
<i>Submerge Mode:</i>	
Up arrow	Move sub up.
Right arrow	Move sub right.
Left arrow	Move sub left.
Down arrow	Move sub down.

The IBM PC and PCjr have a number of graphics-drawing commands which are used in games requiring a lot of animation. The **DRAW** command is capable of creating intricate designs on the screen with a minimum of overhead. (For complex animation though, this command has its limitations.) Once a shape is drawn with this command, you can store the shape in an array using the **GET** command. With the shape in an array, you can easily put it back on the screen anywhere you like with the **PUT** command. The default mode of the **PUT** command is **XOR**. This option will cause the shape to inverse the color of any pixel on the screen that comes in contact with a pixel in the shape. This allows the program to erase the shape by simply drawing at the same location again.

Torpedo Alley (IBM PC & IBM PCjr) Explanation of the Program

Line Nos.		Line Nos.	
100-220	Program header.	570-620	End-of-game routine.
230-320	Initialize program graphics and variables.	630-670	Scan keyboard.
330-370	Set up for start of a new game.	680-890	Submerged game routine.
380-420	Draw the periscope control room.	900-980	Move the islands and adjust ships speed.
430-470	Main control loop for periscope.	990-1120	Fire and move the torpedo.
480-510	Ship hit by torpedo.	1130-1200	Move ship through periscope.
520-560	Ship fires back and hits you.	1210-1250	Input routine and score-display routine.

For your key-in listings, see HCM PROGRAM LISTINGS Contents.

CONTROL CAPSULE Torpedo Alley

KEY	FUNCTION
<i>Surface Mode:</i>	
S	Move periscope left.
D	Move periscope right.
[ENTER]	Dive.
[Space bar]	Fire torpedo.
<i>Submerge Mode:</i>	
E	Move sub up.
S	Move sub right.
D	Move sub left.
X	Move sub down.

Animation on the TI-99/4A is easy with the use of sprites. With their automatic motion, you can have the program set them off on a course while it works on other tasks. You can even change the shape of a sprite while it's in mid-motion—this is how this program simulates the torpedo and its trail of white water getting smaller as it moves farther away. A loop which starts in line 660 reads the shapes from a **DATA** statement as they are needed. Because the sprites get their shape from redefining the shapes of normal characters, the **CHAR** command is called. This method is not as fast as first defining all of the shapes needed, and then reassigning a different character to the sprite. However, when a large number of graphics characters are required for either the animation or other parts of the program, this is the only method available.

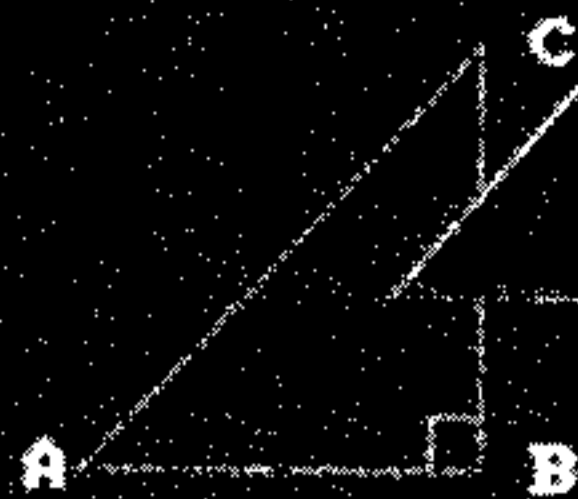
Torpedo Alley (TI-99/4A) Explanation of the Program

Line Nos.		Line Nos.	
100-190	Program header.	930-960	Main control routine—part 2.
200-260	Display title & option screen.	970-1020	Move sub on screen 2.
270-380	Display the main playing screen.	1030-1130	Update depth charges.
390-410	Main control routine—part 1.	1140-1200	Sub surfacing routine.
420-440	Place a ship in the periscope.	1210-1280	Ship shoots back at you.
450-500	Scan for input.	1290-1480	Program data for graphics.
510-600	Check ship and island position.	1490-1510	Explosion sound effects.
610-730	Fire torpedo at the ship.	1520-1550	Update ships motion.
740-800	End-of-game routine.	1560-1570	Subroutine to create the sonar sound.
810-860	Select one of four island shapes.		
870-920	Dive routine. Submerge below water.		

 Torpedo Alley requires TI Extended BASIC.

For your key-in listings, see HCM PROGRAM LISTINGS Contents.

HCM



AB = 6
BC = 10
LENGTH OF SEGMENT AC IS: ? ■

GEOMETRIX

by Wayne Koberstein
HCM Staff

*A sleek program to help you
"measure the Earth."*

How high is that tree? How long is that rafter? How much fence do I need to keep these animals enclosed? How many square feet in this oddball-shaped yard? These are the kinds of questions that long ago prompted humans to place a mental grid over their picture of the real world. They used this method of measuring the earth to build pyramids, navigate, and to create a whole abstract mathematical system based on simple two-dimensional shapes—a system the Greeks called geometry. But the practical reasons for this theoretical base are still with us today: We still need to "measure the Earth."

Geometrix is designed to administer problems dealing with the basic shapes of plane geometry. It is both an educational and a practical tool. Four different types of shapes are used by the program: rectangles, right triangles, parallelograms, and trapezoids. Each shape has its own unique set of mathematical relationships expressed by a given formula. In this text, we provide the basic formulas for rectangles and right triangles. Parallelograms and trapezoids are combinations of these first two shapes—so to solve these more complicated shape problems, you will use the skills that you've already learned from the simpler exercises. In working with each problem, you may use a small calculator for the arithmetic involved; but you will also

have to employ your powers of logic to arrive at a correct answer. The computer's job is to supply random problems and check for errors.

Although the shapes that the program displays appear to be abstract, they have a real practical value. These shapes occur throughout nature, and in man-made materials and objects. Trees and seashells, as well as buildings, cars, and telephones, are all examples of complex structures made up of simple geometric shapes. With *Geometrix*, you can discover how these basic shapes serve as building blocks for more complicated forms. And, conversely, you will learn how to see intricate objects and relationships in terms of their basic shape components. The sidebar "Geometrix at Home" provides some "real-world" applications for the shapes and formulas included in this program.

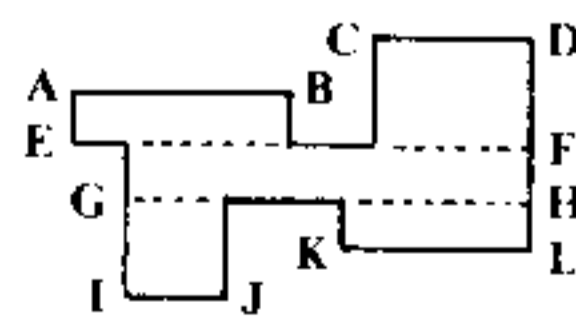
Getting With The Program

For each basic shape, there are three levels of difficulty. Each succeeding level supplies a more complex problem, often requiring you to solve several smaller problems in order to come up with a final answer. (An increase in difficulty also occurs as you move to the more complex shapes: parallelograms and trapezoids.) These exercises may at first seem almost *too simple*—but the lower-level problems have a purpose that will become

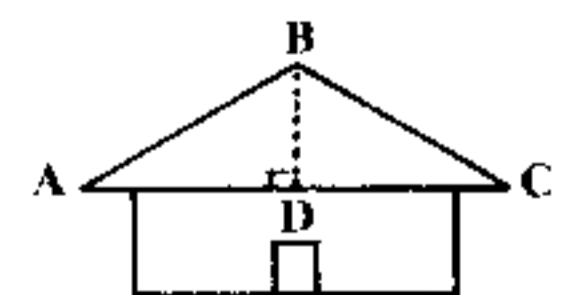
Geometrix At Home

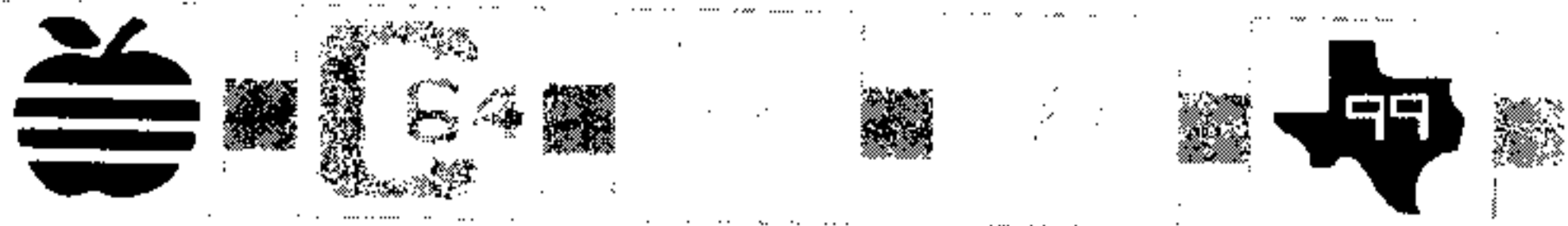
All of the basic shapes we investigate with this program have their applications in everyday life. Here are just a few examples:

1. Rectangles—Often when we deal with a rectangular surface, we need to find the area of that surface. Suppose you have an odd-shaped yard covered with grass (or wheat, etc.). As with our drawing, the boundary of your yard takes many 90-degree turns. How do you find the area of such a seemingly complicated shape? By referring to the Level 3 exercise for rectangles, you can see that one shape may consist of many component rectangles—as indicated by the dotted lines in this drawing. Simply figure the area of each rectangle and add up the total.



2. Right Triangles—In nature, triangles are fundamental—they abound everywhere. Although not all triangles are *right* triangles, any triangle can be analyzed as a right triangle by drawing a perpendicular line from its base to its apex—as represented in drawing 2 by the dotted line, BD. This line—often called the *height*—divides any triangle into two right triangles. A carpenter uses this feature in computing the length of rafters for a roof frame. If you see the "gable" end of a house as a triangle like the one shown here, the rafters would correspond to the segments AB and BC. Knowing the width of a given side (either AD or DC) and the height from roof-base to roof-top (BD) is sufficient to determine the rafter length.





apparent as you progress. These earlier exercises merely pound the stakes for the more complicated constructions.

The computer will check each answer, which must be rounded to the second decimal place. If you give three incorrect answers, the program supplies the correct solution. After completing a given exercise, you have a chance to enter your own variables—thus creating your own problem, which the computer will check for errors. Because each exercise uses a specific shape, each variable has a certain range of legal values. Entering an illegal value will cause illogical results—such as negative numbers—or other unexpected effects on the program. Part of the exercise entails figuring out what the legal range should be in advance of actually entering a value.

All of the basic shapes as they appear on the screen remain constant—they do not change their apparent dimensions as the stated values change. Each screen figure serves not as a literal representation of the dimensions given in a specific problem, but only as reference. This places even more emphasis on logic, rather than visual measurement or guess.

- 1) Rectangles
- 2) Right Triangles
- 3) Parallelograms
- 4) Trapezoids

Main Menu

The main menu will allow you to choose the type of shape you would like to work with.

After selecting a shape, you must select one of three difficulty levels (Level 1 is the easiest). The following is a breakdown of each shape, and the problems that each level will supply.

For clarity, let it be understood that two capital letters placed together represent a line segment between the points indicated by each letter. For example, AB represents a line segment between point A and point B. Also, the number values used in each problem are "generic" units; you could substitute any standard measuring unit and apply it to any problem.

1) Rectangles

Level 1

This problem presents a simple square. A square is defined as having four sides,

all being the same length, with the angle of the four corners being 90 degrees wide. The problem is to find the area of the square, with two sides known, using the basic formula:

$$\text{AREA} = \text{AB} * \text{BC} \text{ Given AB and BC}$$

Level 2

This problem is just like that of Level 1, except that rectangle sides do not have to be of equal length. Level 2 also uses the same basic formula and the same "givens" as Level 1.

Level 3

This level introduces a more complex shape: two rectangles put together to form an upside down L shape. Four of the possible eight line lengths in this shape are given. You must find the area of the shape by analyzing it in terms of its basic components.

2) Right Triangles

Level 1

Right triangles are defined as having only three sides, with one of the angles equal to 90 degrees. Right triangles have certain properties which make calculating unknowns simple. This first level is fairly easy—you are given the two sides adjacent to the 90-degree corner: AB and BC. You must find the area of the triangle, using the basic formula:

$$\text{AREA} = (\text{AB} * \text{BC}) / 2$$

Level 2

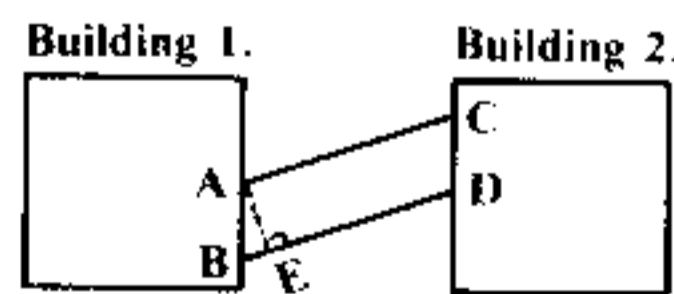
This problem uses the same shape as the one in Level 1, and asks that you find the length of the hypotenuse AC, which is the side opposite the 90-degree angle. To solve this problem, you can use the formula contained in the classic Pythagorean theorem on the relationship between the hypotenuse of a right triangle and its other two sides. Here we express this formula as:

$$\text{AC} = \sqrt{(\text{AB}^2 + \text{BC}^2)} \text{ Given AB and BC.}$$

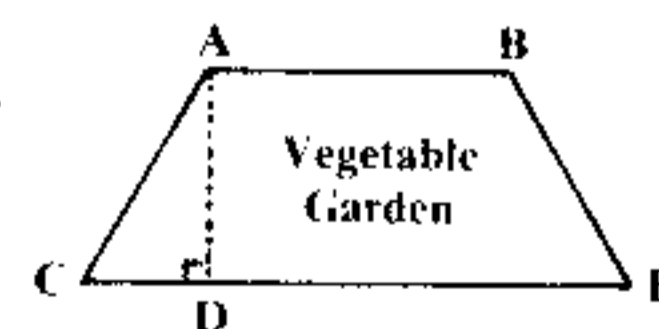
Level 3

This level is only slightly more complicated than Level 2; however, the shape has been changed to a vertically-symmetrical

3. Parallelograms—As long as we're up on the roof, let's look at how differently shaped roof surfaces require different tactics to, say, figure the amount of material required to cover them. Suppose our carpenter has to build a roof over a walkway between two buildings. These structures sit parallel to each other; but the doorways connected by the walkway are not directly across from one another. Looking directly down on the walkway, we see that its roof forms a parallelogram. (To keep it simple, we'll say this is a flat roof.) How much area, then, does this roof cover? If you know the width of the walkway (AE), and the length of one side (AC), it is relatively easy to calculate its total area.



others, occurs often in all kinds of construction. But, for variety, let's consider a garden that has to fit in a space that is narrower on one end than on the other—effectively, a trapezoid. How much area can our gardener cover with vegetables in this space? Referring to the Level 1 exercise, we need only to know the length of the narrow side (AB), the long side (CE), and the height (AD).



These examples are only suggestions of what can be done with basic geometric shapes in everyday circumstances. Perhaps you can imagine—or will encounter—other similar situations. You may even turn some of these examples around to exercise other aspects of what you've learned with *Geometrix*. For instance: What if you had to build a structure with trapezoidal sides? How would you calculate on paper the lengths of the bracing members in the side frames? Look at Trapezoid Levels 2 and 3 for some ideas on this.

HCM

triangle, which divides into two right triangles—each a mirror image of the other. Use the basic formula from Level 1 to solve the problem, given AB, BD, and AC.

3) Parallelograms

Level 1

A parallelogram has four sides. Opposite sides are always parallel with each other, and have the same length. Levels 1, 2, and 3 all require you to solve for area. To do this, you need to know the length of one side—in this case, the segment AC—and the distance to the opposite side—here, the “height” AE (a line 90 degrees from the known side to the opposite side). For Level 1, you are given AC and AE. Use the basic formula:

$$\text{AREA} = \text{AC} \times \text{AE}$$

Level 2

This problem is quite a bit more complex. It requires you to solve several problems to arrive at the correct answer. The unknown link in solving this puzzle is the height of the shape, segment AE. If you can find the length of that segment, you can use the formula from the first level to arrive at a solution. You are given AD, DF, and AB.

Level 3

This problem, like the exercise in Level 2, is more complex—requiring you to solve several problems before the final answer can be obtained. You are given EB, BF, and DE.

4) Trapezoids

Level 1

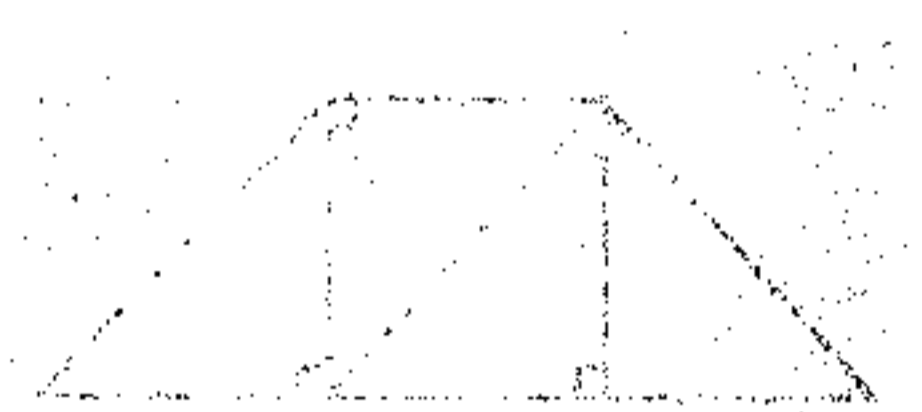
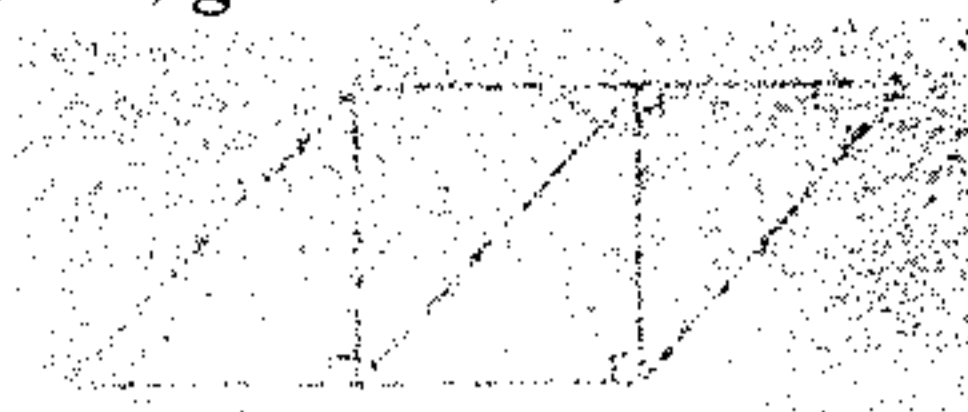
A trapezoid is defined as a shape with four sides, two of which are parallel. This shape poses more complex problems than the previous three. Because of the difficulties involved in solving such problems without getting into trigonometry, a few additional clues have been supplied for Levels 2 and 3. In Level 1, you must solve for the total area, given AB, AD, and CF.

Level 2

To solve this problem, you should know that the angles at points C and F are both 45 degrees. This still makes for a very challenging problem. Solve for segment AD, given AB and CF.

Level 3

This last problem is a real brain-twister. Again, it is necessary to supply additional information. Here we merely state that the two angles at C and F are equal to each other. Find segment AC, given AB, DB, and CF.



“... you will learn how to see intricate objects and relationships in terms of their basic shape components.”

you. All of this can be a worthwhile effort if the task at hand is very complicated. For small or limited jobs though, it may be easier to rely on the DRAW and XDRAW commands in conjunction with shape tables.

In this program we needed to recreate the letters A through H for display on the high-resolution screen. These characters are used to label the line segments of the geometric shapes. Lines 2110 and 2120 contain the shape table data. The first two values indicate how many shapes are in the table. Following this will be two bytes for each shape—these indicate where in the shape table the shape data starts. By knowing this, you can easily find the shape data for each of the eight shapes.

Two commands can access these shapes and place them on the high-resolution screen for you. The DRAW

command will place the indicated shape at the location you specify, replacing any graphics which may already be on the screen under the shape. The XDRAW command is most commonly used for animation because of its special

ability to reverse the color of the screen at every point where the shape is drawn. This simplifies animation because you can draw at the same location twice, reversing the screen colors where the shape draws again, and reverting them back to their original state before the first shape was drawn.

A number of short routines from line 1710 through 1960 draw the geometric shapes and display the letters which mark the line segments.




The Commodore 64's BASIC language can be difficult to work with, though in many ways it is very forgiving. The INPUT statement is one example. In a “clean” program, it should be so difficult for average users to make a fatal mistake (one that crashes the program or causes an error) that they couldn't make one if they tried. This is perhaps the most difficult aspect of programming—making sure that all “end-cases” are checked.

End cases are the limits within which the program must operate to perform properly. It could be as simple as making sure a user can't crash the program by pressing a key that was never meant to be pressed, or doing a complex edit check on a user's typed input in search of a legal response.

One end-case check which is often overlooked in BASIC programs is in the entry of numeric information. If you use a numeric variable with the INPUT statement, inputting anything other than a legal number will bring the cryptic message “REDO FROM START.” How does one avoid this?

Instead of using a numeric variable with the INPUT statement, use a string variable. When string variables are used, virtually anything can be entered. If you need to have a numeric value from the input, then you can convert the string characters into a numeric value. The VAL function does this for you, and it is much more flexible with numeric variables than the INPUT statement. The VAL function will scan the list of characters in a string and convert as many as it can into a number until it runs into an illegal character. This means that if you accidentally typed 123R, the VAL function would return the value 123. Any other numeric characters after the R would also be ignored—e.g., 123R456 would return a value of 123.



One of the biggest problems that Apple II programmers face is the inability to place standard characters on the hi-resolution screen. But, a number of alternatives are available to the programmer.

One method would be to create a machine-language routine (see “Apple Seedlings” in this issue) which assists in graphics development and manipulation. However, you must know machine language, or have access to a program which can manipulate graphics for



Have you ever noticed the little arrow symbol on the (ENTER) key of your PC or PCjr keyboard? You may also have noticed that quite a few programs use this symbol to represent that key rather than spelling out the word "RETURN" or "ENTER." Two built-in graphics characters resident on the system allow you to build this shape for use within text: ASCII characters 17 and 217. In a program which uses this symbol many times, it is sometimes useful to place these two characters into a string variable with a name which is easily understood:

```
RETRNS$ = CHR$(17) + CHR$(217)
```

This works very well as long as you're in Text mode or if you have a PCjr. But if you are using a PC in one of the graphics modes, this string may not work properly, depending on the color graphics adapter card that you are using. (Some graphics adapter cards do not support the built-in graphics character set; the PCjr does support it). If your program has that problem, you still can draw in the arrow shape as we did in *Geometrix*. It requires a little more effort and planning than the method just described, but the results are just as satisfying.

First, set up a string variable which contains the DRAWing commands:

```
ENTDRS$ = 'E3DG2DE2DGDUFU2ER6U3LD2LU2'
```

Before you can DRAW this shape, you need to determine where on the screen it should appear. There are a number of ways this could be done. One method involves adding another command to the beginning of the DRAW command string mentioned earlier. The BM command causes the pointer for the DRAW command to move to the coordinates specified without drawing a line. Another method (the one used in this program) is to simply use the PSET command to position the DRAWing cursor before DRAWing the shape.

One problem programmers still face in displaying graphics on the TI-99/4A is the lack of an expedient way to get graphics characters on the screen in large quantities. Innumerable algorithms have been designed to help this process—some quite complex, and some quite simple. In the case of this program, the simplest way is not only the most efficient, but also the quickest.

The one requirement for this method's success is that all of the graphics characters must be confined to those characters accessible from the keyboard; you will need to be able to type their unaltered counterpart into data statements. A FOR-NEXT loop can then be used to READ the data, one line of graphics at a time, and print it to the screen using the PRINT statement. If you are using Extended BASIC, you could speed up this process even more by using the DISPLAY AT statement.

To simplify the design of such a screen full of graphics, a piece of graph paper may be necessary. Then you can map out the position of each character on paper, so that when you enter the characters into the DATA statements, you will be able to count spaces and repetitions of characters (to ascertain their proper alignment with the rest of the screen). This may reduce your screen-development time by 50 percent or more.

The DATA statements which contain the screen format are in lines 4390 through 4470. In this program, we have left the upper-case (capital) letters alone so that they may be used as the labels for the line segments of the geometric shape. The lower-case letters and other characters have been redefined to create the geometric shapes. In these few lines of DATA are the formats for seven geometric shapes. To designate the end of the graphics for a screen, a null item is used in the data. A null item occurs when two commas are placed one after the other.

For your key-in listings, see HCM PROGRAM LISTINGS Contents.

HCM

Geometrix (Apple II Family) Explanation of the Program

Line Nos.	Explanation
100-200	Program header.
210-230	Relocate program.
240-340	Get options.
350-720	Rectangles.
730-1030	Right triangles.
1040-1360	Parallelograms.
1370-1700	Trapezoids.
1710-1960	Draw shapes.
1970-1990	End-of-problem routine.
2000-2100	Key-scan and input routines.
2110-2120	Graphics data.
2130	Text-messages data.

Geometrix (IBM PC and PCjr) Explanation of the Program

Line Nos.	Explanation
100-230	Program header.
240-320	Get options.
330-480	Rectangles.
490-620	Right triangles.
630-800	Parallelograms.
810-980	Trapezoids.
990-1000	Input routines for answers.
1010-1020	Check answer.
1030-1040	Next problem menu.
1050-1130	Draw shapes.
1140	Key-input routine.

Geometrix (C-64) Explanation of the Program

Line Nos.	Explanation
100-200	Program header.
210-360	Get options.
370-760	Rectangles.
770-1130	Right triangles.
1140-1510	Parallelograms.
1520-1940	Trapezoids.
1950-2080	Check and get the answer.
2090-2150	Next problem menu.
2160-2640	Draw shapes.
2650-2660	Time-delay loop.
2670-2690	Continue routine.

Geometrix (TI-99/4A) Explanation of the Program

Line Nos.	Explanation
100-200	Program header.
210-290	Initialize the program.
300-460	Get options.
470-1340	Rectangles.
1350-2140	Right triangles.
2150-2860	Parallelograms.
2870-3710	Trapezoids.
3720-3970	Misc. input routines.
3980-4020	Next problem menu.
4030-4060	Time delay and beep routine.
4070-4100	Single-key input routine.
4110-4470	Control and do shapes.



Here they are . . . the best of the one-line programs that we have received since printing the second "HCM One-Liners" column in *Home Computer Magazine* Vol. 5, No. 2. Although many interesting programs were submitted, we have selected what we felt were the best four (one for each brand of computer covered in our magazine) of those that arrived prior to this issue's press date. If you have not yet submitted your masterpiece, it is not too late! As long as we keep getting great One-Liners written in any computer language, we'll keep filling this page for you. Our prize winners this issue will each receive a check for \$50 for sharing their ideas with our readers.

Polar Coordinates [Applesoft BASIC on the Apple IIe, IIc]

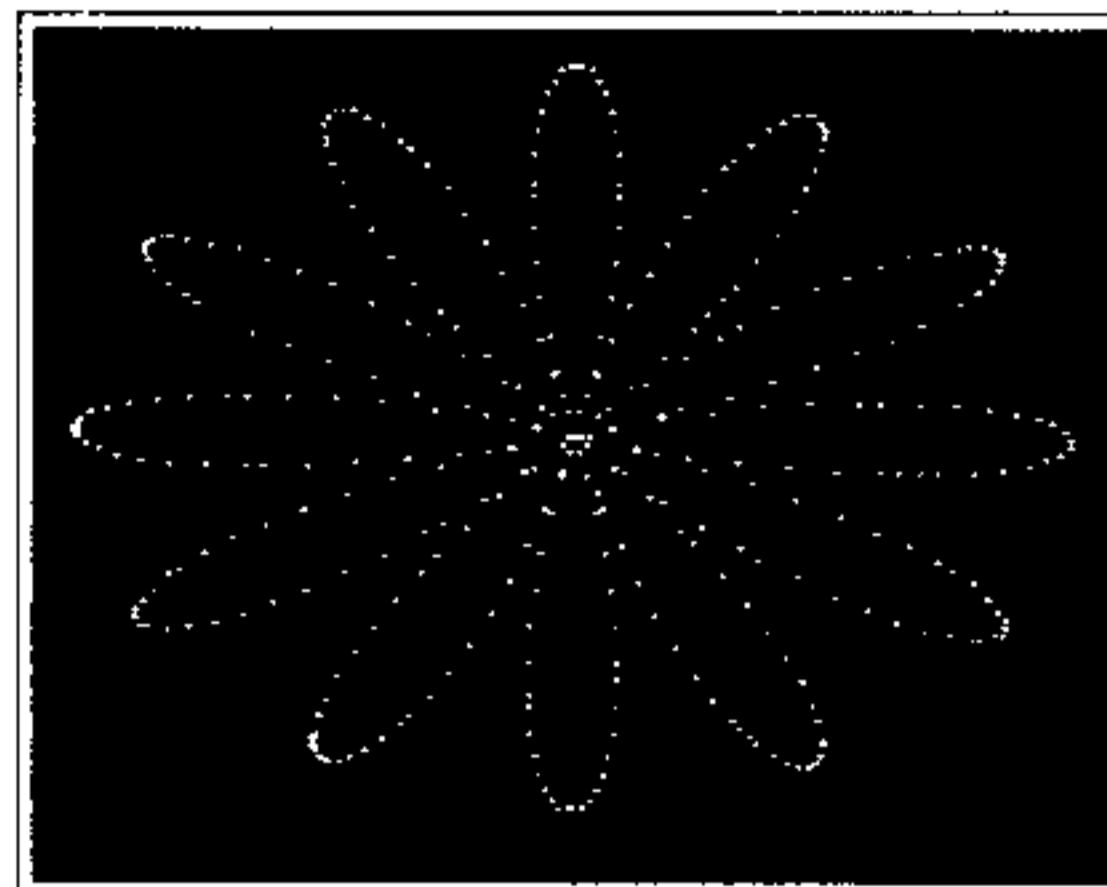
Dear Sir:

After reading your magazine, I decided to enter a one-liner of my own. This one-liner draws 10 different polar coordinate curves. They are referred to in some calculus books as leaved-roses. Each odd-numbered pattern shows that same number of leaves; each even-

numbered pattern shows double that even number of leaves. It takes about 8 minutes to finish all 10 curves.

James R. Klenke
Whitewater, KS 67154

```
1 TEXT 10 HOME 2 FOR B =
1 TO 3: C = 0: D = 360: COLOR =
* (INT (B / 2)) = B / C
2 AND Z > 5 FOR A = 1 TO 360
X = 139 * Y = 95 * Z = 95 * SIN
+ (A) * Z: HPLOT X, Y: N
NEXT A, B: TEXT HOME
```



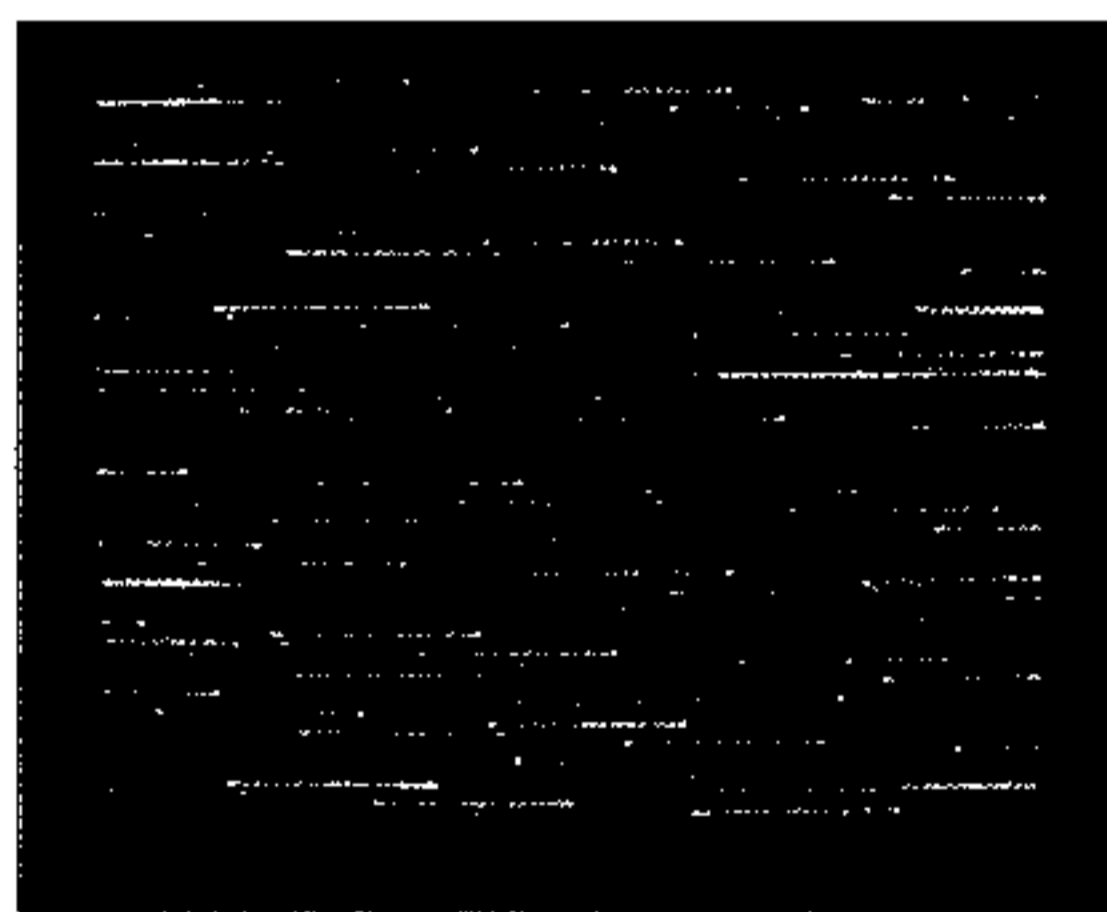
"It's Alive!" [Commodore 64 BASIC on the C-64]

Hi There HCM,

Here's a one-liner that will make your screen come alive. By pressing down various keys, you can create a variety of effects. The only way to exit from this frantic routine is to press the (RUN/STOP) and (RESTORE) keys simultaneously.

Thom Randall
Roseburg, OR 97470

```
1 ? "SHIFT CLR" : S = 828 :
PUSH 1, 33: POKES + 2, 208: P
OKES + 3, 76: POKES + 4, 60: P
OKES + 5, 3: SYS S
```



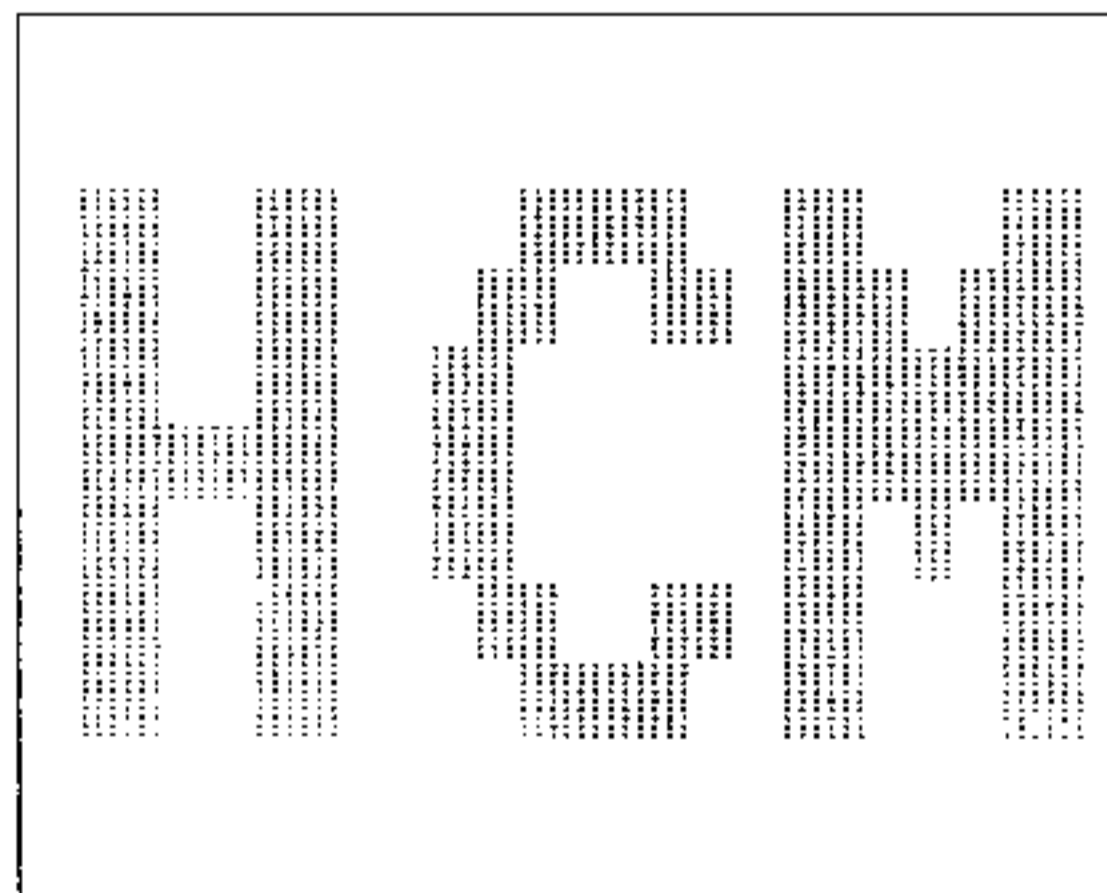
Message Banner [BASICA on the IBM PC, Cartridge BASIC on the IBM PCjr]

Dear Editors,

Included is my one-liner contest submission for the IBM PCjr (it also RUNS on the PC). As you can see, the program prints a banner on a line printer. Thanks for the great coverage of the PCjr! [NOTE: You can input up to 255

characters for your custom banner.]
David J. Bohlke
Manchester, IA 52057

```
1 SCREEN 1: COLOR 1: CLS:
INPUT <ENTER> BANNER: C
M$: FOR A = 1 TO LEN(M$): C
LS: PRINT MID$(M$, A, 1): F
OR C = 0 TO 7: FOR I = 1 TO F
3: FOR R = 7 TO 0 STEP -1:
A$ = STRING$(9, POINT(C, R)
+ 32): LPRINT A$: NEXT L
RINT: NEXT: NEXT: NEXT
```



Tabling Multiplication [TI Extended BASIC on the TI-99/4A]

Dear Editor,

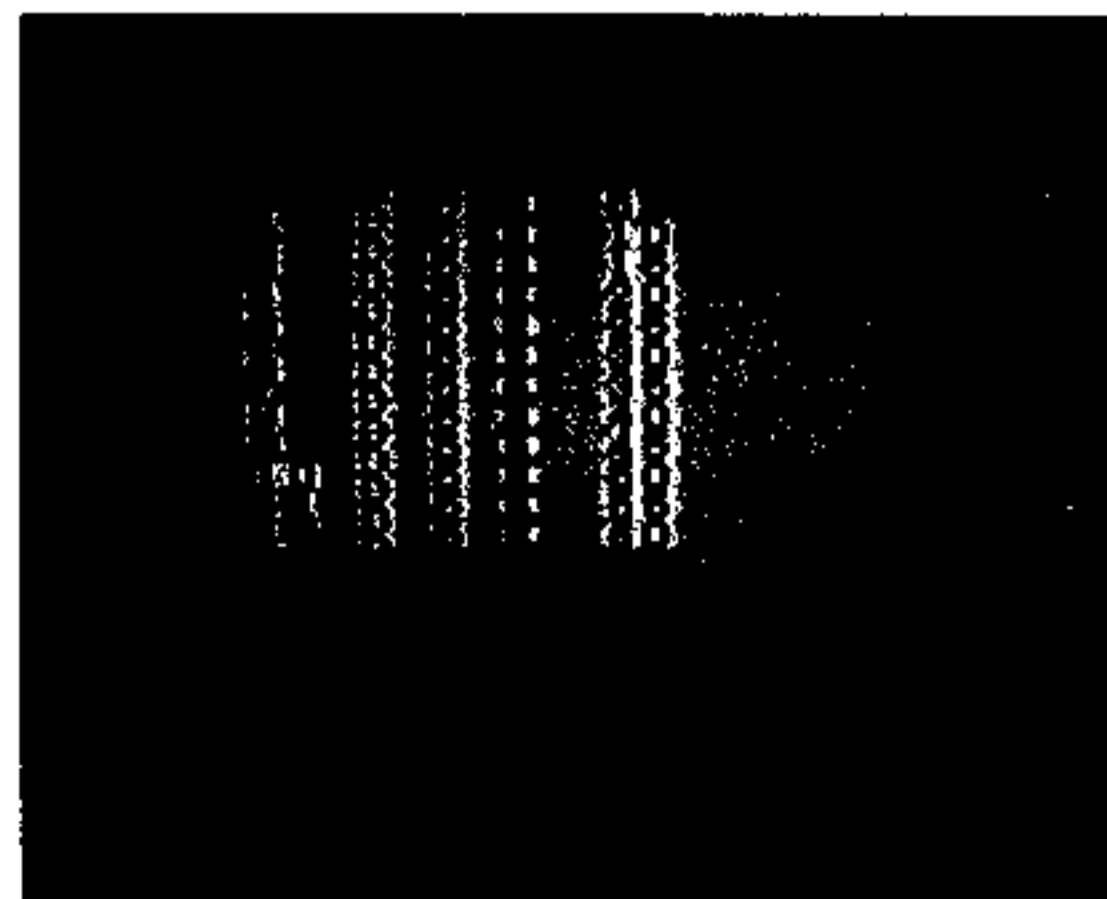
I read with interest your call for one-liners in *HCM* Vol. 4, No. 5. The ones in that issue were most interesting, and made me think.

This is my entry for your contest: When you RUN this program, input a number at the blinking cursor. A table

with this number and multipliers 1-12 will be displayed with the answers in columnar format.

Raleigh H. McQueen
Savannah, GA 31419

```
1 ACCEPT AT (1, 1): B: F
OR A = 1 TO 12: DISPLAY
AT (A + 3, 5): A: TAB (9): "X"
: TAB (10): B: TAB (12 + LEN(S
TR$(B))): "=": TAB (14 + LEN
(STR$(B))): A * B: NEXT
A: GOTO 1
```



All One-Liner submissions are subject to the same publishing criteria as Letters to the Editor (explained in the magazine's masthead, page 6). If you have written a great One-Liner in any language on any computer covered by *HCM*, send it addressed to: Letters to the Editor, 1500 Valley River Drive, Ste. 250, Eugene, OR 97401. You too may win a cash prize and be immortalized in print!


HCM Review Criteria

Each month, *Home Computer Magazine (HCM)* reviews products designed for the Apple II Family, Commodore 64 and VIC-20, IBM PC and PCjr, and Texas Instruments 99/4A computers. *HCM* reviews take a detailed look at the quality, utility, and value of commercially available packages for these machines. Because our publishing charter forbids accepting outside advertising, we strive to make the scope and content of our review pages shine with a unique blend of humanistic frankness and objectivity.

Not only will you find all relevant information for making a wise purchase decision, but in some special cases we also provide nuggets of compu-prestidigitation.* For example, we frequently include essential documentation not furnished by the manufacturer. Additionally, each issue of *HCM* tries to review at least one outstanding product—a "Diamond in the Rough"—which, because of company size, marketing clout, or for some other reason, has not received the attention it deserves.

At the beginning of each review, a review-at-a-glance box provides the user with an instant assessment of the product. Each item will be evaluated, where relevant, with the criteria below.

HCM Review



Name: Old Art
Program Type: Recycled Graphics
Machine: Apple II Family, C-64 & VIC-20, IBM PC & PCjr, TI-99/4A

Distributor: Hit 'n' RUN Software, Inc.
Price: \$99.99 (or trade for '72 Pinto)

System Requirements:
 Disk Drive, Joystick, Trash Can optional

Performance:
Engrossment:
Documentation:

*** Performance—**
 How well the product performs as intended, how well it takes advantage of a specific machine's capabilities, how well it responds to the user's commands, how effectively the graphics, sound effects, music, or speech are integrated with the software.

*** Engrossment—**
 Whether the game or activity has that intangible quality that holds players on the edge of their seats while the hours tick by unnoticed.

OR

*** Ease of Use—**
 The degree to which a user can interact with the product without outside help; the ease and effectiveness of error-handling features; whether the actual reading level of the activity is appropriate for the suggested audience.

OR

*** Ease of Set-up—**
 How well the product design facilitates easy installation.

*** Documentation—**
 The quality of the printed matter that comes with the product; whether the instructions are clear and comprehensive; whether the machine configuration requirements are spelled out. Information such as how to load a program, use the keyboard, and restart an activity contributes to the documentation rating, as do tips on performance peculiarities.

Products may also be evaluated in the following areas:

*** Flexibility—**
 Can the product be adapted to the specific needs of the users?

*** Cost/Benefit—**
 Is the product worth the user's investment in time and money?

*** Necessity—**
 Is the product a solution for which a problem already exists?

*** Originality—**
 Is it unique in concept, or simply a "me too" product?

*** Longevity—**
 The "Boredom Factor." Does the program sustain interest?

*** Rewards—**
 Are the audio-visual rewards motivating and appropriate?

*** Concept Presentation—**
 Are the concepts presented clearly, logically, and in depth?

*** Special Effects—**
 How does quality of sound and visual effects rate? Do they enhance or detract from the product or learning process?

Attention Software Authors & Peripheral Inventors:

*** WANT TO BE DISCOVERED? ***

Home Computer Magazine Wants To Give You A Chance!

We are looking for home computer products that have not received the attention they deserve. Each month, we will be singling out one such package for special review. If you have a unique commercial product of exceptional quality—but your advertising and promotion budget has

not allowed you to capture major media attention—we want to see it. We will consider reviewing any product that meets our high standards.

We are an Equal Opportunity Reviewer!

In order to qualify for possible review, your product must:

1. Currently be available for purchase to readers of this magazine.
2. Make a unique and important contribution to the home computer industry.
3. Be of outstanding merit, quality, and value.
4. Be consistent with the type of machines and products we normally cover.

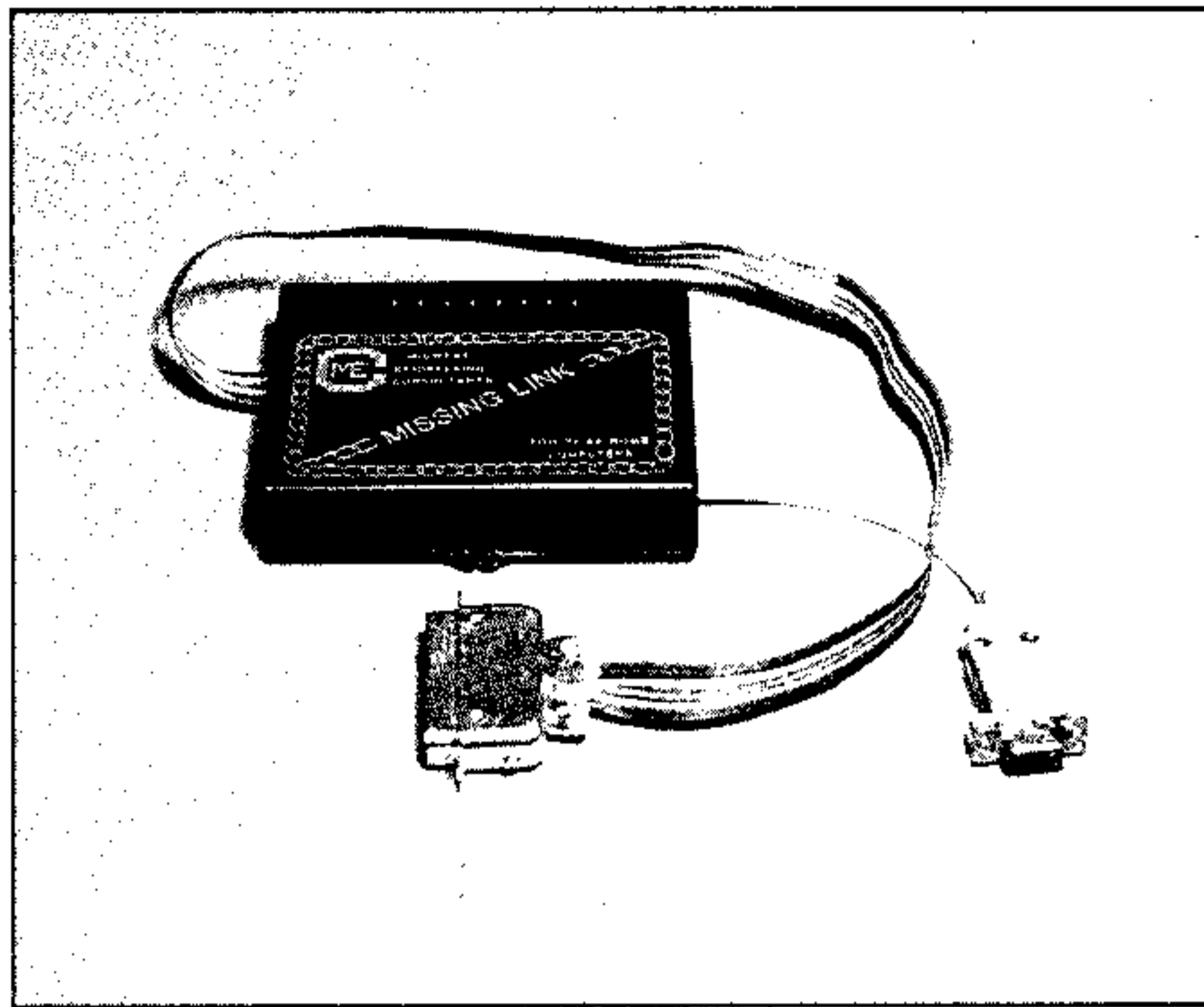
If you feel that your product qualifies, mail it to:

Home Computer Magazine
 Attn: Editorial Submissions
 1500 Valley River Drive, Suite 250
 Eugene, OR, 97401

We reserve the right *not* to reply to each inquiry, so please do *not* contact us except to request return of your product. If you want your product to be returned, please include sufficient return postage.

***Compu-prestidigitation**

(kóm•pū•prēs•teh•dī•jeh•tā•shŭn) —*n* 1. The magical quality of unexpected comprehension that results from presenting technical information about computers in a lively, entertaining, visually attractive and easy-to-understand format. 2. The magical tricks that make a computer sing, dance, and do all sorts of wonderfully useful things.



THE MISSING LINK

for the TI-99/4A

A Review by Patricia Swift

Parallel printers abound—but if you want to connect one to your TI-99/4A, you may be looking at some big bucks, unless . . .

In the chain of life, there are many missing links—missing, in that they are not yet found. Computer users encounter these all the time: "If only there were a thing to go from here to over there!" For TI-99/4A users in particular, the missing link has been one that will *inexpensively* connect their computer to a parallel printer.

If your printer has a parallel interface, you could spend a lot of time and money trying to attach it to your TI-99/4A. You could add Texas Instrument's Peripheral Expansion Box and RS-232 interface card (which has a parallel port), and then buy a cable somewhere. But *Missing Link* offers a low-cost alternative.

To use *Missing Link*, you'll need either one of two configurations: (1) a Mini Memory cartridge and a cassette recorder with cable, or (2) an Extended BASIC cartridge, 32K expansion memory, and a disk drive.

A Good Connection

The *Missing Link* hardware consists of a small plastic box with cables attached. The cables connect to the joystick port on the left of the 99/4A console and to the parallel connector on the printer. *Missing Link* works by sending characters out the joystick port, through the device to the printer. Actually, the "Missing Link Algorithms" send the characters. These algorithms are two assembly-language subroutines, which you can call from your own programs. The MSLINK subroutine sends a string of characters out the joystick port. For example, if you wanted to print the string A\$ from a BASIC program, you would say `CALL LINK("MSLINK",A$)`. The VPLINK subroutine is a screen dump program which is even easier to call from BASIC by using: `CALL LINK("VPLINK")`.

The software for the Mini Memory version comes on cassette tape. One side has the "Missing Link Algorithms," which are loaded into Mini Memory so that console BASIC programs can use them. The other side has a word-processing program written in console BASIC.

The software for the disk version includes the "Missing Link Algorithms" in assembly language, and a word processor in Extended BASIC. It also includes Extended BASIC programs for building sprites, for printing certain files via *Missing Link*, and for performing electrical engineering calculations. The 32K expansion memory

is required. All assembly language routines are stored on and accessed from the disk.

You will not be able to LIST your BASIC programs via the *Missing Link* in the usual way, because BASIC's LIST command doesn't use the joystick port. *Missing Link* offers some utilities for listing programs to the printer, but they are not as convenient to use as LIST. Documentation with the package described a Mini Memory program for LISTing programs which should have been on the tape, but I couldn't find it.

Setup: Cassette Version

The *Missing Link* package includes some very clear and detailed instructions for setting it up on a system with Mini Memory and cassette tape. I would add one more step: after you hook up *Missing Link*, go into console BASIC and create a short program (all REM's will do). Then SAVE CS1 to write the program to tape, and answer Y when asked to verify. If you get the DATA OK message, you're set to go. If not, you will have a chance to adjust things until you can record successfully.

After I loaded the "Missing Link Algorithms" into Mini Memory via Easy Bug, I was ready to test the VPLINK algorithm, the one that dumps the screen to the printer. I got an accurate printout of the screen, but all of the characters in the printout had been expanded 200 percent! Undaunted, I continued with the installation instructions by loading the word processing program. I used the word processor to make up a short text and print it out. Happily, my printer was able to print this perfectly—no expanded print problems.

I contacted the program's author about VPLINK, to ask why it was expanding my print while the word processor was acting normally. It happens that ASCII character 31 signals my Okidata printer to use expanded print, and that BASIC uses the same character for its "screen edge" characters at the beginning of its lines. If ASCII 31 means something to your printer, you may find that VPLINK is not useful as shipped.

This brings up an important point about *Missing Link*: you should be prepared to do some fine tuning to make things work perfectly with your printer. If you have a Gemini 10X printer, which is what Midwest Engineering uses, then *Missing Link* will probably work flawlessly right out of the box. But if you have a different printer,



Name: Missing Link
 Product Type: Parallel printer interface
 Machine: TI-99/4A
 Distributor: Midwest Engineering Consultants
 P.O. Box 159
 Vernon Hills, IL 60061
 Price: \$30 (cassette) \$34 (disk)
 System Requirements: Cassette version: Mini Memory, cassette recorder. Disk version: Extended BASIC, 32K memory expansion, Disk drive.

	Poor	Fair	Good	Excellent
Performance:	████████████████████			
Ease of Use:	██████████████████			
Ease of Setup:	██████████████			
Documentation:	██████			
Cost/Benefit:	████████████████████			

you may want to make some adjustments, and you may need a working knowledge of assembly language to do so.

Word Processing

The word processing program is the main event in both the cassette and disk versions. When you RUN the word processor, you see an introductory screen which explains the program and then waits for your input. At first I wasn't sure what input was expected, but I soon discovered that the program wanted me to set margins.

The next thing you see is the word processing menu, from which you select options by number. You can add new lines, change existing lines, insert lines, and delete lines. You can replace strings. You can save your text to tape or disk and read it back in later.

Because most printed lines are longer than the 28 columns you get on a BASIC screen, there is no way you'll ever be able to see your text on the screen in its final printed form. The word processor gives you two ways to print through *Missing Link*: either "formatted" or "unformatted." Formatted means that special operators for centering, paragraphs, and continuation are translated to achieve the desired results. In this mode, you can specify that the lines be numbered, or that the un-numbered text be printed single-spaced or double-spaced. You can also print unformatted text.

The cassette version of the word processor has one restriction because it's written in console BASIC: you can't have commas in your text when SAVEing and LOADing files. This is because a comma is a "field separator," which means something special to BASIC. In fact, the only way to put commas in text is to use another character as a placeholder. You could use semicolons as temporary placeholders for commas, and then use a special form of the Search and Replace option to replace all semicolons with commas. (Commas are legal in the disk version because it is written in Extended BASIC.)

If your printer has functions that you control by sending it special characters, you can make use of them with *Missing Link*'s word processor. For example, on my Okidata, ASCII 29 means compressed print. *Missing Link*'s documentation (and the *TI BASIC Manual*) shows that the keyboard combination [CTRL] = yields an ASCII 29. Thus, I was able to compress the print by putting a [CTRL] = right into my text. The *Missing Link* documentation lists the proper codes for a Gemini 10X. If you have a different printer, you will need to be familiar with its control codes.

The MSLINK algorithm subtracts 128 from any character with a value over 127 before sending it to the printer. This is not unreasonable, because standard ASCII characters are 7 bit values (127 maximum); but, it means that you will not be able to send control codes over 127 unless you modify the algorithm.

The Zero Bug

After printing my first document, I did what most people would do: I switched the printer off-line and ejected the page to tear it off. With the printer off-line (but with

the power still on), zeroes are fed back to the console. This can fill up the keyboard buffer and produce beeps of protest from the 99/4A. The zeroes (and beeping) stop when you switch the printer back on-line. This happens whenever the *Link* is attached, not just with the word processor. In some situations, you might want to remove the zeroes from the screen before proceeding, but that's the worst-case result. In other words, this problem is annoying but not dangerous. My best solution was to put a page-eject character ([CTRL] L in most cases) at the end of my documents.

Setup: Disk Version

The package includes a prominent "Precautionary Notice" for users who plan to use the *Missing Link* with TI's Peripheral Expansion Box (PEB) hooked up. This notice talks about possible differences in electrical potential between your printer and the PEB. *Missing Link* ties the two devices together electrically, and as such can damage

the console in rare cases. To avoid this, connect a wire between any screw on the PEB and a grounded screw on your printer.

No step-by-step documentation for setting up the Extended BASIC version is included. Simply mount the disk before you select Extended BASIC from the 99/4A's master menu, and you will get a list of the programs on the disk. You may select the program you want by number.

Conclusions

Missing Link is a good product which takes some time to learn to use. Its word processor is useful, and the product's low cost makes it very attractive. The package suffers, however, from sparse documentation—instructions that assume the user knows more than is reasonable to expect. (The cassette setup instructions are a notable exception to this statement.) Several documents are included, but they are not well organized. If you have a Gemini 10X or a "plain vanilla" printer, you can probably use *Missing Link* and its word processor as is. If you have a different printer, be prepared to spend some time customizing the software and/or learning to make your printer work correctly with the existing software. This is especially true if you want to make use of your printer's special features.

HCM

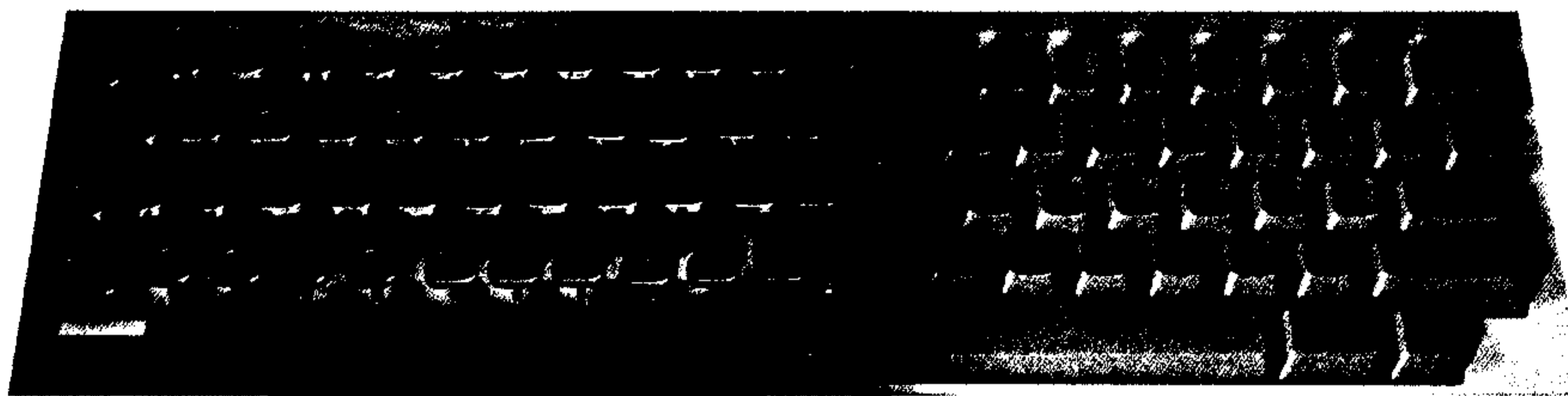
... you should be prepared to do some fine tuning to make things work perfectly with your printer.



COMPUTER-AGE TYPING:

A LOOK AT SOME KEY TYPING-INSTRUCTION PROGRAMS

by Dana M. Campbell
HCM Staff



With all of the programs available out there waiting to teach us how to type or help us type better and faster, how do we know what is a bargain, and what to avoid?

The more we learn about the human body, the more we realize how important exercise is for it. But although exercise may be generally regarded as healthy for the body and essential for its well-being, that doesn't mean that people always enjoy it, or even do it. Part of the problem may be attributed to the dreariness of traditional calisthenics, for along with the spread of novel, enjoyable ways to exercise (aerobics classes, jogging, health clubs, etc.) came droves of newly converted exercise nuts.

The same can be said of the importance of learning to type or improving existing typing skills. Now that computers are marching into the home, the school, and every office from the main floor up to the executive suite, typing ability has become a crucial survival skill in the technological age. Unfortunately, traditional typing classes and manuals generate about as much enthusiasm as the thought of drinking a glass of prune juice. Well fear not, modern one—with the exciting, ever-changing technology of the computer come brand new ways of learning typing skills. Today's available typing software spans the range from edge-of-your-seat, typing game excitement to more subdued fingering techniques. Here we take a look at several typing instruction programs in relation to a list of desirable typing program traits, as shown in Figure 1.

Games People Play

By offering a game option, it's possible for typing-tutor programs to provide a truly engrossing way to exercise typing skills. The more innovative game/learning programs make good use of the computer medium, stepping beyond the capabilities of traditional typing manuals. For instance, *MasterType* and *WizType* are almost entirely based on a game format, and they are certainly not of the drudge school of learning.

MasterType in particular offers a visually exciting way to learn and improve, as well as a great deal of flexibility. In this program, the user is a wizard who must defend his spaceship against an onslaught of words scattered around him in space. The words are defended by missiles, which advance at a speed equal to the user's words-per-minute (wpm) rate set in practice sessions. If a word is typed before its missile reaches the wizard, it and its missile are destroyed with a mighty zap by the old necromancer, only to be replaced by a new word. Similarly, *WizType*—which features the comic strip characters Wizard and Spirit from *The Wizard of Id*—levitates words or letters in the castle basement. As they are typed, the Wizard zaps them. If the words are typed too slow, Spirit turns into a dragon and torches the Wizard. Again, the words must be typed at a rate set by the user.

Space Invaders look-a-likes abound in the game options of *Keyboard Command*, *Touch Typing Tutor*, and *Typing Tutor III*. Though these 3 programs are more serious in format, they all offer variations on the *Space Invaders* theme—here letter combinations and increasingly difficult words and symbols drop out of the sky at faster and faster rates, and must be typed before hitting the baseline, cityscape, (fill-in the appropriate screen scene), etc. However, *Typing Instructor* takes a dive with its *Lobster Sea Adventure*.

You simply type a paragraph accurately and quickly enough to stay ahead of the lobster (which looks nothing like a lobster) before it "eats" you.

Generally, the game element seems to help decrease the boredom of drill work by making these programs *interesting*, and that's not just my opinion. *MasterType*, for example, has been on top of the educational-software charts for more than 2 years.

"Any computer typing-tutor program worth its salt as a learning medium should be able to take the information, process it, and use it to help you in future lessons . . ."

Can You Put Your Finger On It?

Whether lifting weights or trying to type, knowing the proper way to execute the exercise when you first begin will save you from picking up harmful bad habits later on. *Touch Typing Tutor* and *Typing Instructor* begin by detailing the proper typing posture and hand position and carefully continue on. *Type 'N' Write* merely highlights the proper keys to press on a screen reproduction of the keyboard, and it does this only when in the first of 8 modes.

Worse yet, *Typing Coach* assigns each home-row finger a color, and uses colors on the screen keyboard to clue you in to which keys to press with each finger. This essentially provides a crutch that you come to rely on, hindering the process of learning which letters to associate with each finger. It's adding an unnecessary extra step: First you must remember that the forefingers are red—later you learn that red stands for the F and J keys.

Brief written and visual guides on proper keyboard fingering accompany the other typing programs at the start of each new lesson. But because the depth and presentation of beginning finger-placement instruction varies so much between programs, you should view a demonstration of several of these programs before buying to determine what is appropriate for you (or your child, or whoever the intended user is).

As Long As It's Accurate . . .

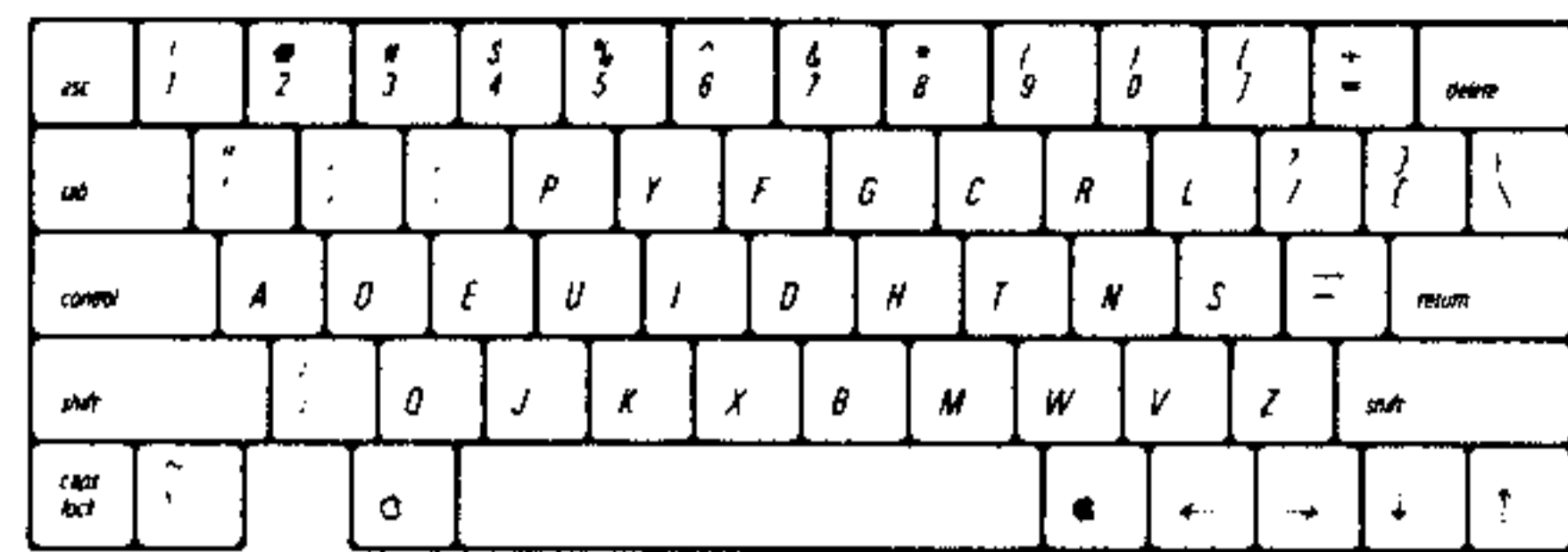
The phrase "Do it until you get it right," usually spoken by angry teachers and parents, seems to be the over-riding philosophy threaded through these programs. Apparently, most typing-program developers believe that given enough repetitious exercise, anyone can improve their typing accuracy. It's a sad theory, and I hate to advocate anything that promotes drudgery, but it seems to be true. For instance, *Typing Coach* has 28 lessons, each introducing 2 new keys. You type various letter sequences that include those two keys—and any others that you learned in previous lessons—for 10 lines. You may continue to repeat the exercise as long as you want for each lesson. Believe me, when you finish a lesson, you *know* those keys.

Type 'N' Write is an attractive alternative for kids, with colorful icons of children at play guiding you toward keyboard success. Here again you type letter sequences, continuing to repeat the same line until infinity or until you complete it with fewer than 5 errors, whichever comes first. It may be tedious, but it does work.

Zippering Along

Speed building on the track or on a keyboard is synonymous with racing—either against a competitor, or against your own time. In a typing context, wpm tests for comparison against your own previously set rate are the most common way used in typing software to build speed. Some programs simply time you as you type during the regular lessons, while others have separate, specific speed tests and drills. The latter are the ones indicated with a YES answer in Figure 1. *Typing Tutor III* provides several timed test options, covering just words, just numbers, the full keyboard, and a "standard" speed test of up to 7 lines of words to type. These are good options to look for when shopping for speed-building programs.

Touch Typing Tutor—the only program that we could find still being distributed for the TI-99/4A—is ponderous to use, especially for speed building. Part of the problem stems from the 99/4A's keyboard layout. The fact that you must reach for the (FCTN) key to produce such punctuation as a question mark, and apos-



Dvorak: A Keyboard Alternative

If you are just learning to type and you have an Apple IIc, you might consider trying the Dvorak keyboard layout, as opposed to the QWERTY layout common to most typewriters and computers. The original keyboard design that we still use today (with the letters Q, W, E, R, T, and Y appearing on the second row, left side) was arranged to deliberately slow down typists so that the mechanical keys on the old-time typewriters would not tangle and jam up. On a Dvorak keyboard, the 10 home-row keys (A, O, E, U, I, D, H, T, N, and S) comprise 70 percent of all letters typed, according to extensive studies on the subject. Thus, even those people who are already accomplished typists can learn this new keyboard and greatly increase their typing speed.

On the U.S. version of the Apple IIc, the (Keyboard) switch above the keyboard toggles between the Dvorak and the QWERTY layouts. By carefully popping off the keys on the IIc and arranging them in the Dvorak configuration shown in the IIc manual, you can load up *MasterType* or *WizType* and choose their Dvorak options to learn a whole new modern way of typing. The lessons are constructed in the same way as they are for the QWERTY keyboard, except that the keys are learned in a different order, with both versions starting with their respective home rows.

troupe necessarily slows you down. If you are already nimble on a typewriter keyboard, with its standard layout of punctuation keys, you will certainly want to adapt to your TI machine before you start timing yourself. But even without the keyboard differences, this program provides little impetus to increase typing speed.

Don't Forget The Numbers . . .

A quality exercise program would not exclusively focus on the torso and forget the extremities, and neither would a good typing program. All of the programs listed in Figure 1 provide an opportunity to practice skills on the top row of numbers and the punctuation keys. Number drills are, in most cases, separated from the letter drills. Although people who simply want to brush up on those top-row keys may like the separation, others who appreciate the practice that numbers and letters thrown together provides will probably miss the option to do so.

The More Records, The Better

Not to be overlooked by homo-sapien typing tutors is whether these programs are capable of supporting scores, lessons, and progress records for more than one user. *MasterType* and a few others keep track of high scores for their games, but *Keyboard Command* seems specially suited for classroom use. Its Student Management File (a separate data-base program on the disk) is accessible only by a password, and it will hold the records of up to 100 students. Once in the file, the teacher, parent, or user can preselect the lesson number (from 1 to 19 covering all keys), the skill concentration (accuracy, speed, or both), and whether the keyboard should be shown on screen during the lesson. A user may also edit, add, delete, or scroll through and view









student records, which consist of a student's name, last lesson worked on, average drill accuracy and speed, whether the drill was completed, and the record number. The record list may be alphabetized and/or printed out, along with a graph of each student's performance on each key, if desired. It's quite a handy option and an additional way of putting the old computer to use that is often overlooked.

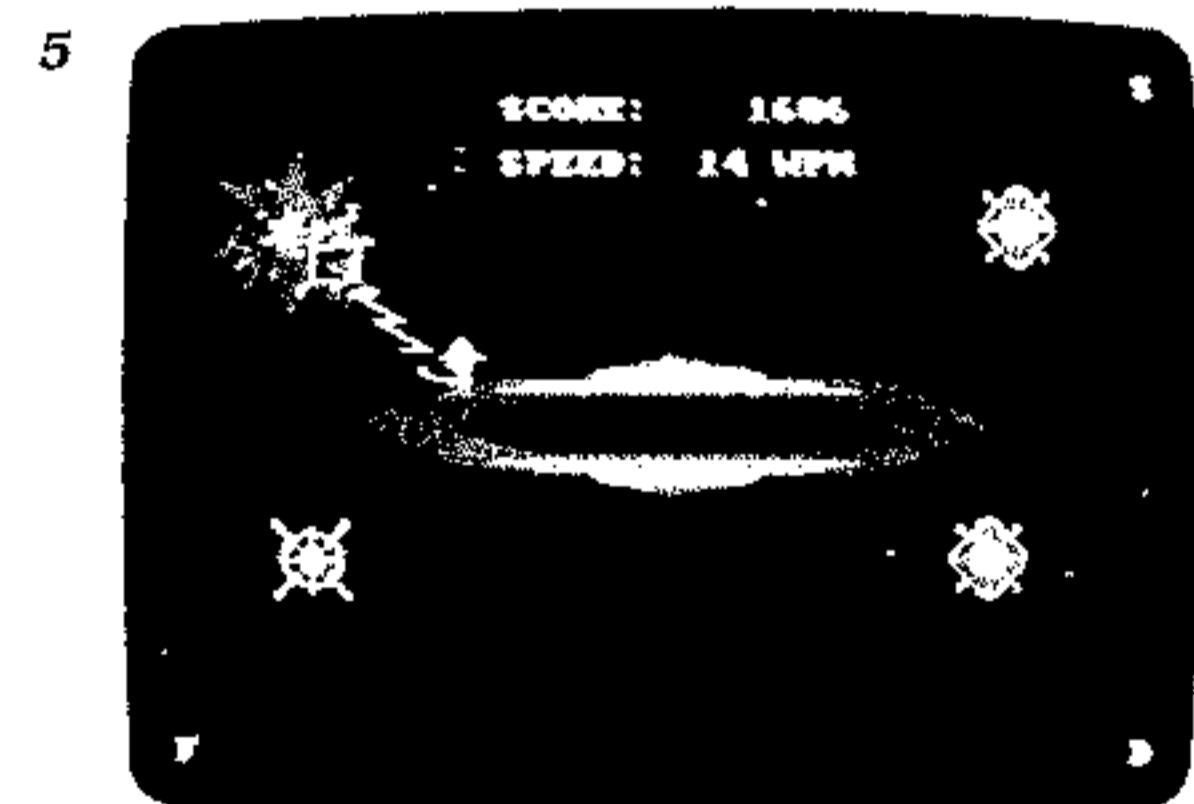
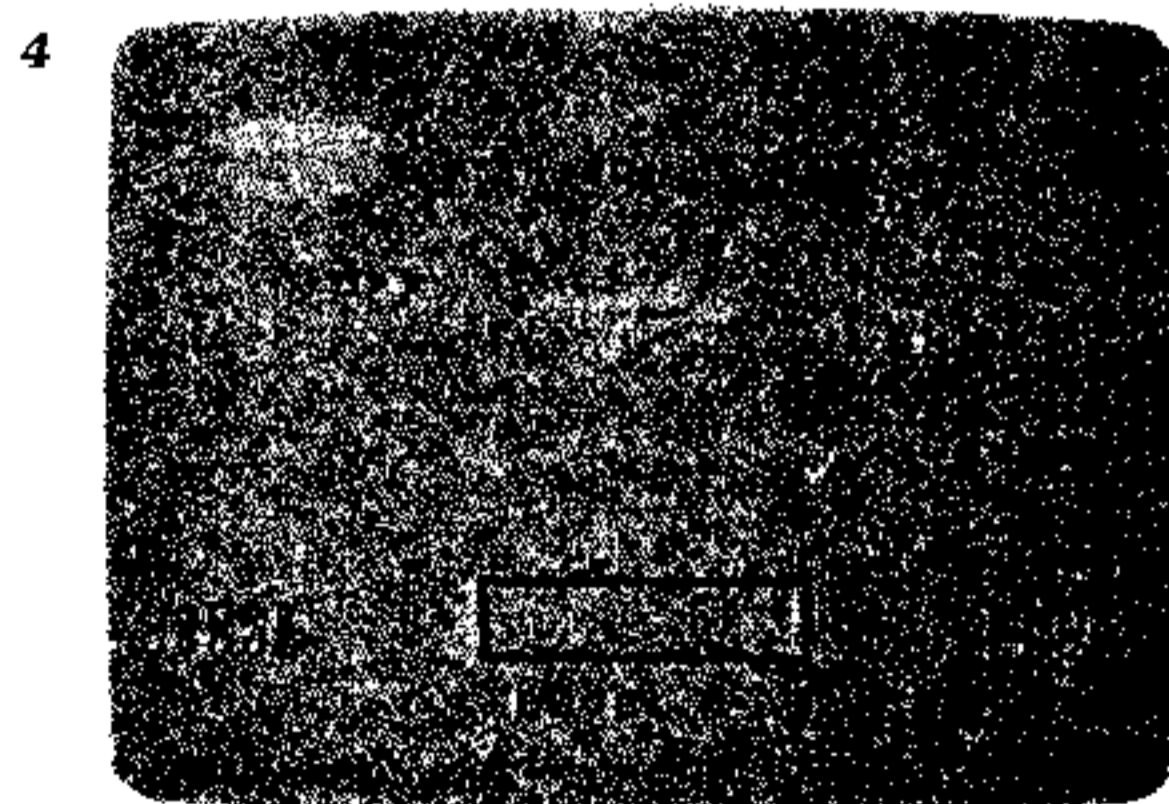
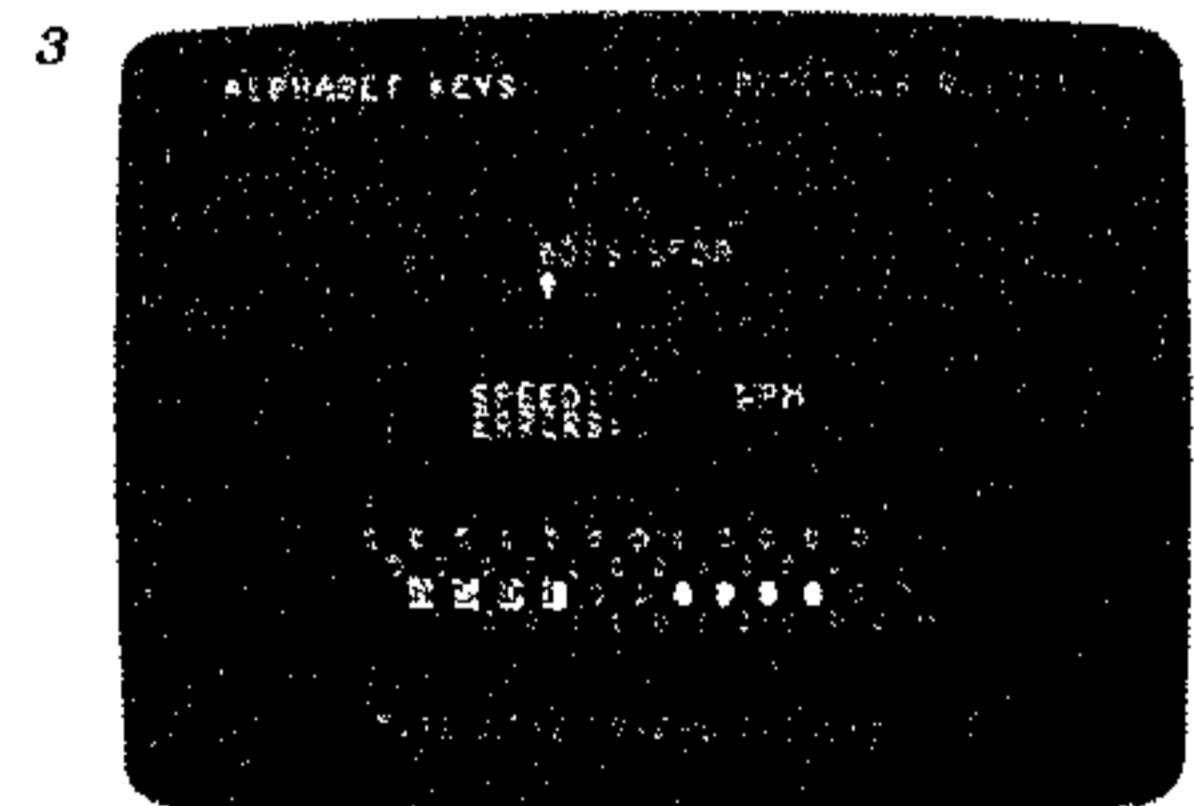
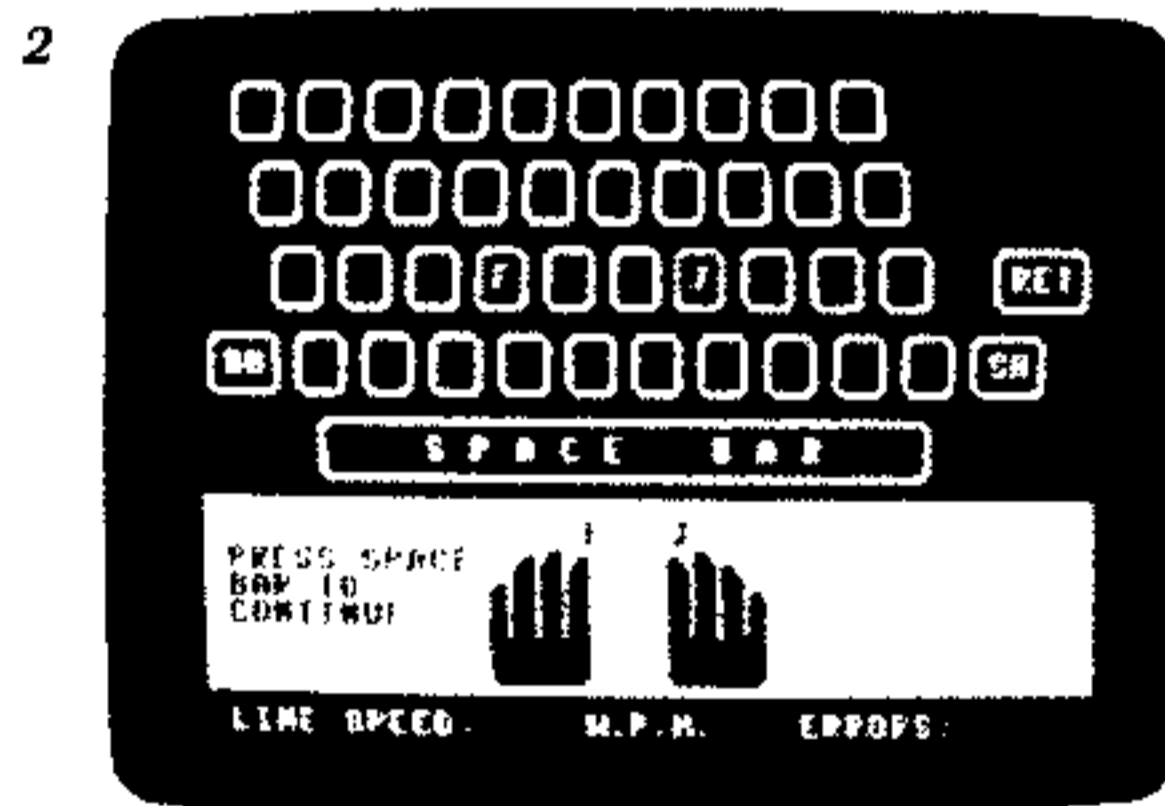
Create Your Own Lessons

The latest in home video and computer health programs have options that let you adjust the program to suit your own needs, and *MasterType*, *WizType*, and *Typing Tutor III* are no different in this respect. *MasterType* allows you to create a lesson file of 40 words, each no longer than 10 characters. They can be words often used and specific to a certain profession, spelling

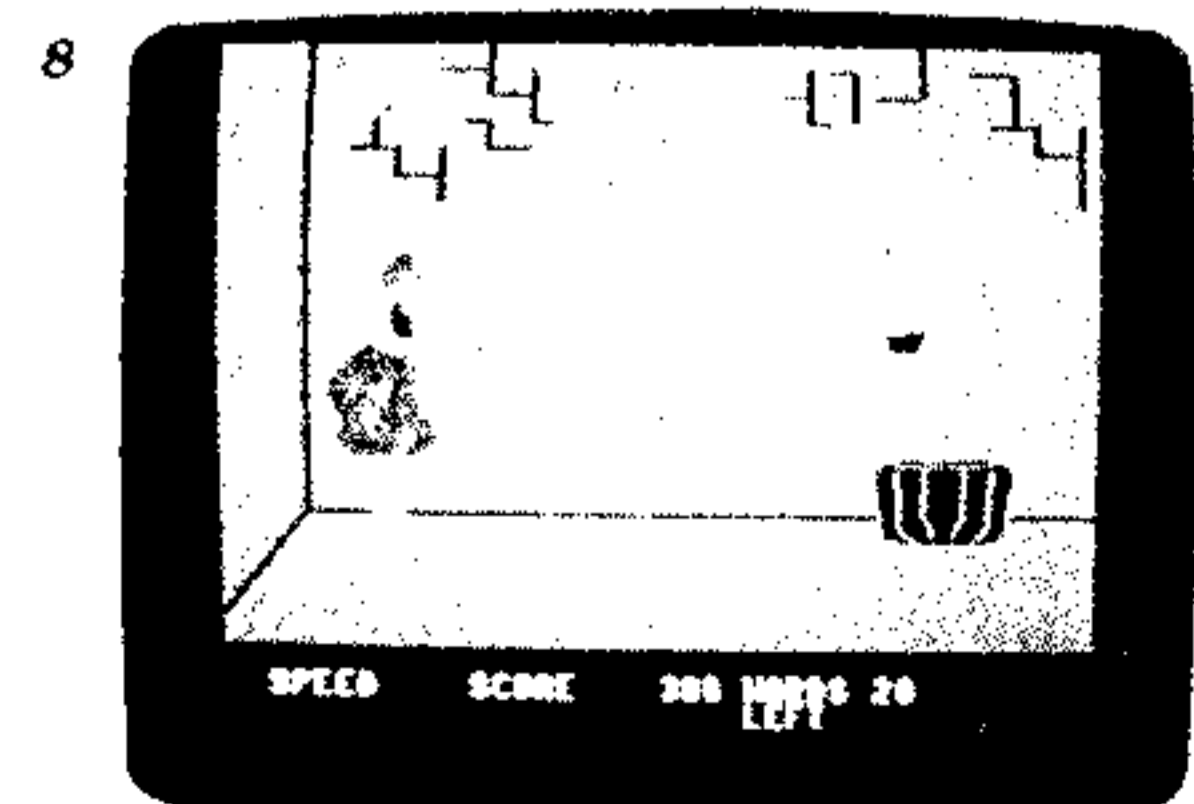
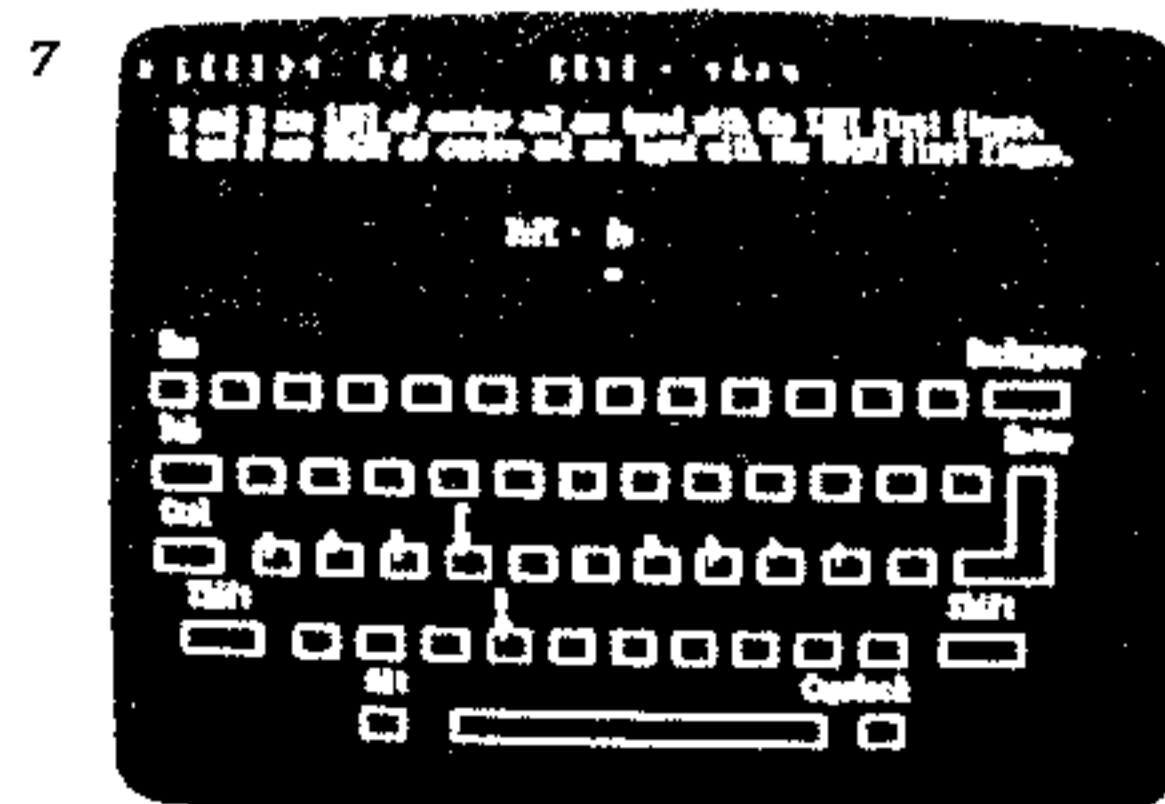
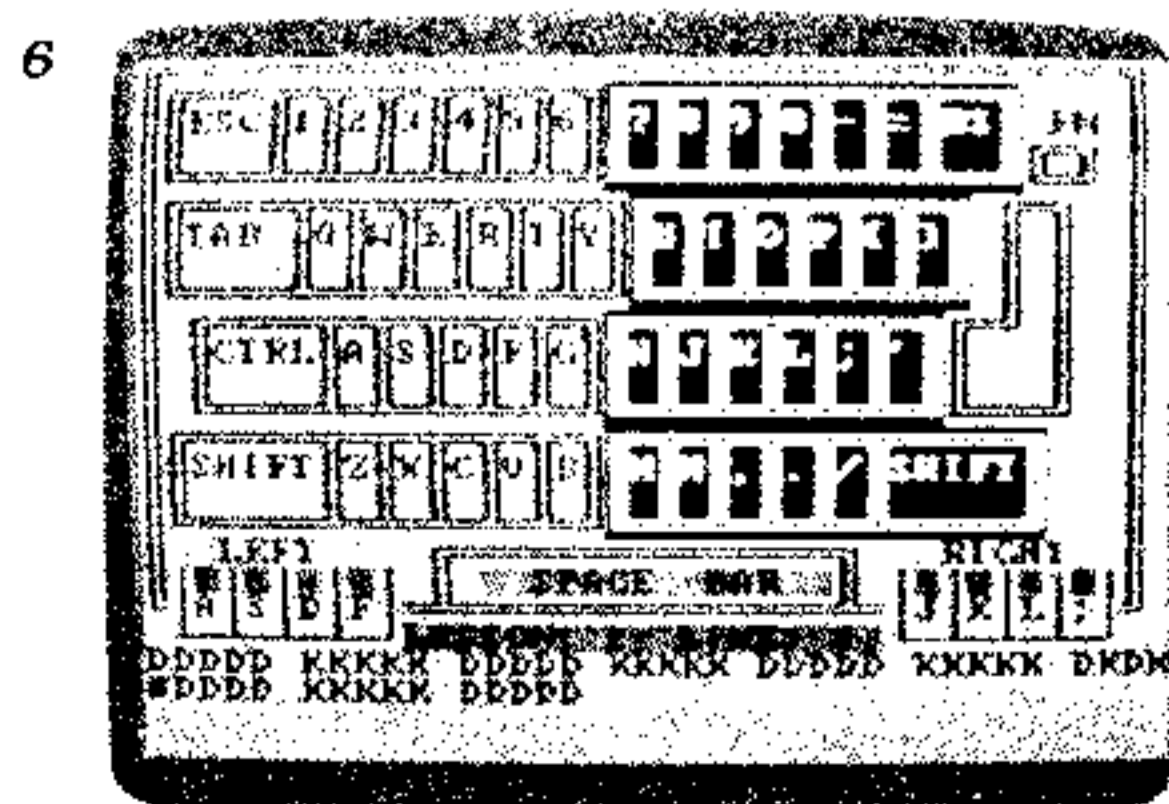
FIGURE 1

Some Popular Typing Programs and their Features

	Key Command 	Type 'N' Write 	Typing Instructor 	Touch Typing Tutor 	MasterType 	Typing Coach 	Typing Tutor III 	WizType 
Game Option	YES		YES	YES	YES		YES	YES
Teaches Finger Placement	YES	YES	YES	YES	YES	YES	YES	YES
Builds Accuracy	YES	YES	YES	YES	YES	YES	YES	YES
Builds Speed	YES		YES	YES	YES		YES	YES
Numbers & Symbols	YES	YES	YES	YES	YES	YES	YES	YES
Multiple Records	YES						YES	YES
Lesson Creation					YES		YES	YES
Analysis Incorporated			YES	YES	YES		YES	
Progress Displayed	YES					YES	YES	
Sentences & Paragraphs			YES	YES	YES		YES	YES
Dvorak Keyboard					YES			YES



1. Key Command.
2. Type 'N' Write.
3. Typing Tutor III.
4. Touch Typing Tutor.
5. MasterType.
6. Typing Coach.
7. Typing Instructor.
8. WizType.



words for grade schoolers, or any other word type. *WizType* allows you to do the same, only it's capable of supporting a list of 240 words no longer than 8 characters each. You can also enter your own text for practice in Paragraph mode.

Typing Tutor III lets you write your own Standard Typing Test text of no more than 7 lines, but only on its IBM PC version. It's a shame this option isn't also offered on the Commodore and Apple versions. With these, the same copy appears again and again—so when your speed continues to dramatically improve, it is probably due to your familiarity with the material. In any case, a lesson-creation option certainly increases any typing program's flexibility.

Functional Feedback

Any computer typing-tutor program worth its salt as a learning medium should be able to take the information input—your typing strengths and weaknesses—process it, and use it to help you in future lessons and tests within the same program. Unfortunately, it appears that only 3 programs here do this. Following each lesson in *Touch Typing Tutor*, the keys on which more practice is needed are displayed, along with the numbers of the lessons where they appear. Another practice option lets you work on these problem keys a little more.

MasterType repeatedly displays letters and words that you miss or are slow to type, and *Typing Tutor III* has something called TRM—Time Response Monitoring. This program monitors the time it takes you to type each letter and then uses that information to create new lessons tailored to your needs.

At-A-Glance Progress Reports

I believe that the ability to directly compare past and present performance in any endeavor does provide some motivation for steady improvement. Even though graphs and other visual representations of a person's performance usually are not integral for a program's use, they provide an instant picture of your progress and problems. *Typing Coach* keeps a chart of your old and current wpm speed and number of errors for each one of its 28 lessons. *Keyboard Command* allows you to print out or simply display a bar graph of your percentage of accuracy and speed for every single key on the keyboard. Finally, *Typing Tutor III* keeps a list

of your "fast" and "slow" keys, and also graphs your accuracy.

Sentences: Putting It All Together

You can learn to do an unlimited number of exercises, but unless you perform an adequate number of each one in your routine, you will be doing only a superficial workout and won't build much stamina. So it is with the discipline of typing. Unfortunately, only one of the programs reviewed here displays more than a word or a few sentences to type at a time. This is fine when initially learning the keys. But in the everyday world, most information comes in bulk form, usually in paragraphs. *WizType* is the only program that provides extensive text samples to practice your speed and skill, and what samples they are! The text choices include pieces from *Tale of Two Cities*, *Hamlet*, *Mother Goose*, Typing Trivia, the Declaration of Independence, the Gettysburg Address, *Cremation of Sam McGee*, and Facts About the Wizard of Id. You can choose to have Bung—the *Wizard of Id*'s drunk court jester—pace you as he jumps on his pogo stick; or, you can try typing at your leisure. However, nowhere does the program keep track of the number of errors you make when typing a drill or paragraph.

Perhaps the next generation of typing programs will take this extra step and provide lengthier samples to provide more experience in dealing with consistency, rhythm, and fatigue.

"By offering a game option, typing-tutor programs provide an engrossing way to practice typing skills that makes good use of the computer medium, stepping beyond the capabilities of traditional typing manuals."

In Summary

Perhaps the greatest advantage that typing-tutor programs offer is the ability to immediately calculate your words per minute and number of errors. And those programs that offer a game element obviously provide a stimulus for increasing speed that no typewriter and manual can offer. Some programs instantly appeal to both youngsters and adults, while others take a drier, more regimented approach to typing instruction and work just as well. All of them are incredibly easy to use, with the barest of documentation required because everything is menu-driven, and, in most cases, it is easy to get into and out of various modes. The "key" to finding the *best* program *most* suited to your needs is to first determine what those needs are, and visit a software dealer to actually try out and compare back-to-back those that are available for your machine. Then sit down and give your fingers a real workout! **HCM**

Typing Instruction Software

Keyboard Command, for the Apple II Family, retails for \$34.95 and is available from Computations, P.O. Box 502, Troy, MI, 48099.

Type 'N' Write, for the Commodore 64, retails for \$24.95 and is distributed by Human Engineered Software, 150 North Hill, Brisbane, CA 94005.

Typing Instructor, for the IBM PC and PCjr, retails for \$49.95 and produced by Individual Software, 1163-1 Chess Dr., Foster City, CA 94404.

Touch Typing Tutor, for the TI-99/4A, retails for \$19.95 and is distributed by Triton Software Co., P.O. Box 8123, San Francisco, CA 94128.

MasterType, for the Apple II Family, Commodore 64, and IBM PC and PCjr retails for \$39.95 and is available from Scarborough Systems, Inc., 25 North Broadway, Tarrytown, New York, 10591.

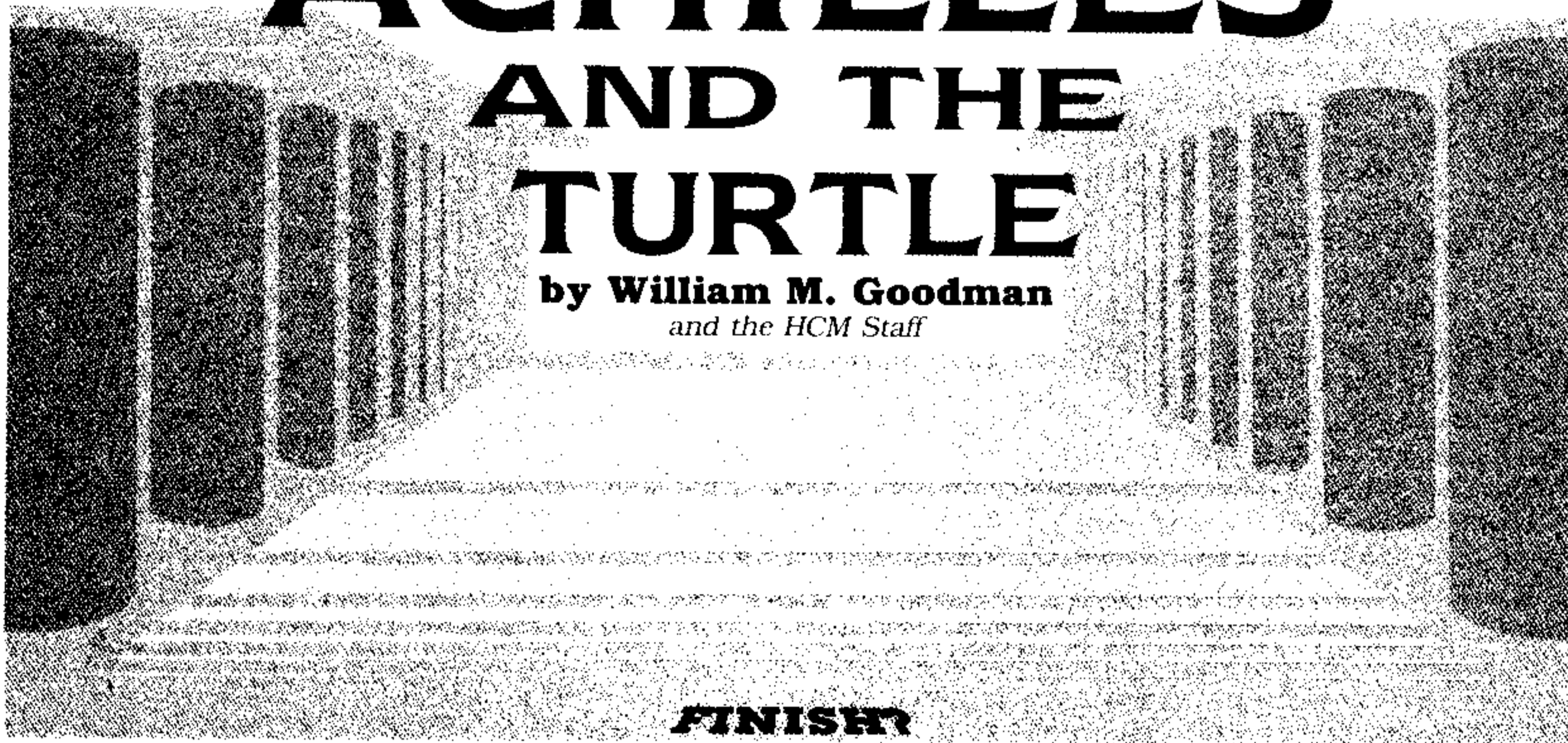
Typing Coach, for the Apple II, II+, and IIe, Commodore 64, and IBM PC and PCjr retails for \$24.99 and is distributed by K-tel Software, Inc., 11311 K-tel Dr., Minnetonka, MN., 55343.

Typing Tutor III, for the Apple II Family, Commodore 64, and IBM PC retails for \$49.95 and is distributed by Simon & Schuster Electronic Publishing, 1230 Avenue of the Americas, New York, NY, 10020.

WizType, for the Apple II Family, Commodore 64, and IBM PCjr retails for \$34.95 and is available from Sierra On-Line, Inc., Coarsegold, CA 93614.

ACHILLES AND THE TURTLE

by William M. Goodman
and the HCM Staff



*No matter how much one may learn about a subject,
one never knows everything about it.
The LOGO turtle shows a Greek philosopher why models
don't always lead to accurate conclusions.*

When one makes a model of something, even the best modeler has to fill in the gaps with assumptions, guesses, and just plain random choices. This is perfectly normal. But problems arise later if one assumes (incorrectly) that because a feature appears on the model, it is a part of the original object as well.

This article is based on an ancient model devised several centuries before Christ by the Greek philosopher Zeno. Zeno "proved" that *motion is impossible*—thus if we think that things can and do actually move, we are merely being deluded. Doubtless you are asking: "With what model could he possibly hope to support *that view?*" As it turns out, he relied on several images; but the one to be used here takes the form of a race between the fleet-footed warrior Achilles and a tortoise (an ancestor of our beloved LOGO Turtle). This race, suggested Zeno, is a *good model* for motion; so if it turns out that the race envisioned cannot be completed, then neither can motion occur—despite all appearances. Unfortunately for Zeno, this is perhaps the first documented case where someone mistook the features of his model—imperfections and all—for real-live properties of the object it was designed to represent.

Zeno's Model

Zeno's model was basically constructed like this: Give the turtle (or tortoise, if you prefer) a head start at place T1, and place Achilles behind him at A (see Figure 1). Then begin the race.

When Achilles reaches T1, the turtle will have advanced to T2. Then, once Achilles gets to T2, the turtle will have already made his way to T3. But when Achilles finally advances in turn to T3, the turtle will have by now arrived at T4 . . . and so on. In short (Zeno concluded), we are deluded if we think that Achilles can ever move to overtake the turtle, because this model *shows* that catching up is simply out of the question—the turtle will always be "a step ahead."

Now I don't know about you, but I for one found Zeno's conclusion to be a bit extreme! So, I decided to use LOGO

to model the race between Achilles and the turtle, to see what *really* happens. This spawned the procedures called *RACE1*, controlled by *FIRSTTRACE*. (For convenience, I will begin referring to Achilles as A., and to the turtle as T.)

The race begins when *FIRSTTRACE* calls the procedure *SHORTTRACE*, where the illustration of A. and T. running is produced by simple animation. Although there is a single sprite for each of A. and T. (Sprite 1 for A., Sprite 2 for T.), each sprite alternates in carrying one of the two distinct shapes for each runner. (See Figure 2 for the shapes and characters.) After each runner takes a "step," the procedure *WINYET?* checks to see whether A. has won yet. When Achilles finally *does* catch up and win, *WINNER* causes him to jump for joy while the turtle slinks away.

Was Zeno Wrong?

"But if Achilles *wins*," you ask, "does this not *prove* that Zeno was wrong after all?" Not quite, I'm afraid. Look at some of the features of *Race1* that do not exactly apply to the "real" race:

1. The relative heights of Achilles and the turtle are not proportional to their real heights.
2. The LOGO colors of A. and T. are not the same as the actual colors (at least, this is unlikely).
3. The two-stage animation of their running styles is a very crude simulation of their actual running styles.
4. The relative speeds of the LOGO A. and T. are not proportional to their real speeds.
5. The running routine in *SHORTTRACE* causes Achilles to move in "quantum leaps" of two or three "turtle-paces" at a time. Moreover, A. does this at moments when T. is motionless. But the "real" Achilles has to somehow "pass through" all of the distance that comprises each step, and must do this when the turtle is also in motion.

Of these distinctions between *Race1* and the real race, the first three are of the "harmless" variety—they make no significant difference in the accuracy of the model

to the real race. The fourth distinction is potentially more significant: If the purpose of the model is to illustrate something about *motion*, is it not important how we choose to represent *speed*? Maybe yes, maybe no. This is just the sort of place where the objectivity of the model can get clouded.

It is the fifth distinction between Zeno's race and *Race1*, however, that is most critical in this example.

Zeno is really asking: "how can a moving Achilles ever catch up to a moving turtle?" Yet our program always moves A. and T. *one at a time*—and always allows A. a leap of several turtle-paces *regardless of where the turtle is positioned*.

So of course A. is bound to win! But this is hardly a model for the same race that Zeno himself had in mind. For him, even if the turtle is only one pace ahead of Achilles, it is still doubtful that A. can catch up to T., provided T. keeps moving. In short, when *Race1* mechanically adds a few extra steps to Achilles' position (hence letting him "get ahead"), it is not acting in a manner which *models* the race that Zeno described. Thus (and this is the key point), the winner of *Race1* does not necessarily correspond to who should have won in Zeno's race!

A LOGO Model For Zeno's Theory

A better model for Zeno's race is found in the *Race2* listing. To appreciate this program, imagine that the race between A. and T. is being filmed by a new type of "3-lens video camera." One lens focuses only on Achilles, while the second focuses only on the turtle (so that these runners appear on the screen always at the same size). The third lens focuses only on the *space between* A. and T. The upshot of this supposed "new technology" is that the space between A. and T. can now be blown up in scale without enlarging the images of these runners at the same time.

The sequence of *Race2* is as follows: First (using tiles) the procedure **MARKERS** causes a row of suitably-scaled flags to appear alongside the race path. As in *Race1*, A. and T. are given their starting positions, with T. having an initial head start. This time the A. sprite is given an initial speed of 8 in the `:ASPEED` variable.

GETSETGO first calls **TURT** to advance the turtle two steps with animation. The procedure **ACHILLES** then updates Achilles' speed and calls upon **RACE** to animate a step for A. **WITHIN26?** next checks to see whether A. has advanced to within 26 "screen-units" of catching T. ("Screen-units" are just a measure of the apparent distance between two points on the video screen image.) If Achilles is not yet that close to the turtle, **GETSETGO** starts over.

Once A. *does* come within 26 screen-units of the turtle, however, **BLOWUP** changes the *scale* of the "video image" of the imagined third lens by a factor of 4. Hence, suppose that the real Achilles and

turtle are separated by, say, five feet, and that this distance is initially represented on the screen by a distance of 26 screen units. Then, after the procedure **BLOWUP** has been run, this same separation would now be represented by $26 \times 4 = 104$ screen units—which is only to say that the distance between A. and T. would appear to have been magnified four times.

To enhance this appearance of magnification, the procedure **MARKERS** is used again to redraw the flags along the race path so that they look farther apart. Also, `:ASPEED` is increased so that A. appears to move faster. Thus the pattern is: **GETSETGO** until A. appears close to the turtle (by the current scale of magnification), then **BLOWUP** the scale by a factor of 4, and **GETSETGO** again. On the screen, a printed reminder always keeps the viewer up-to-date on the current scale being used by the third lens.

Sure enough, when *this* model is used, Achilles *never does* catch up to the turtle. (Try it!) That is, the model captures just that point of view that Zeno intended. Does this mean that Zeno was right? I don't know. If you think that *Race2* is an acceptable model for a real race, then I guess Zeno was right—so always place your bets on the person with the head start! If not, maybe you can suggest a better procedure for representing a real-life race. [Let us know of your enhancements in a "Letter to the Editor"—Ed.]

HCM

For your key-in listings, see HCM PROGRAM LISTINGS CONTENTS.

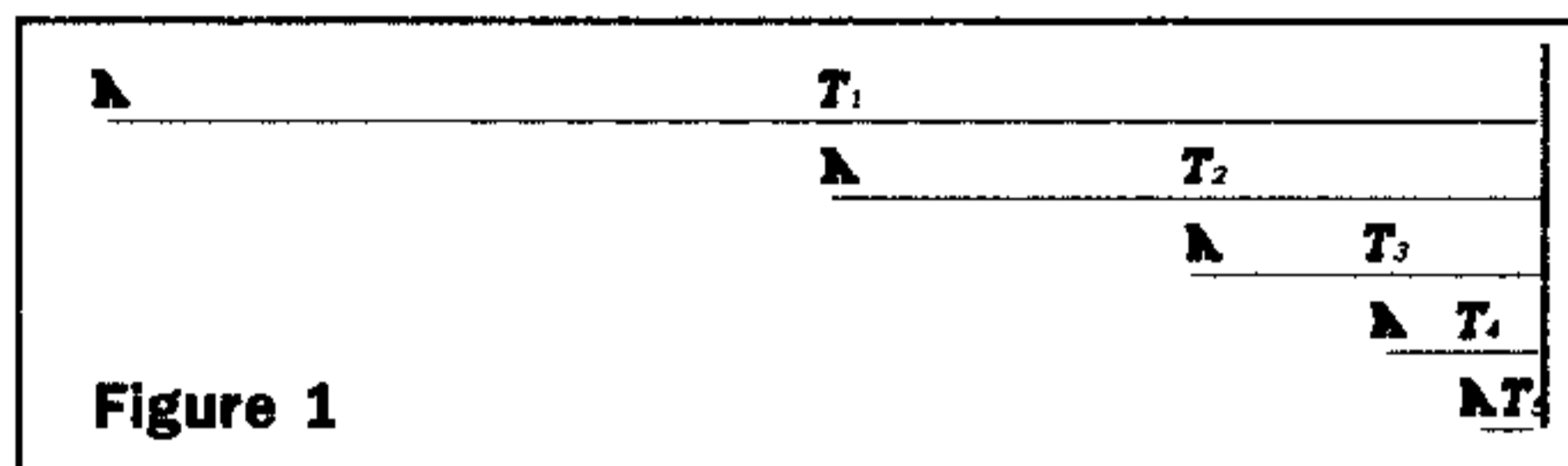
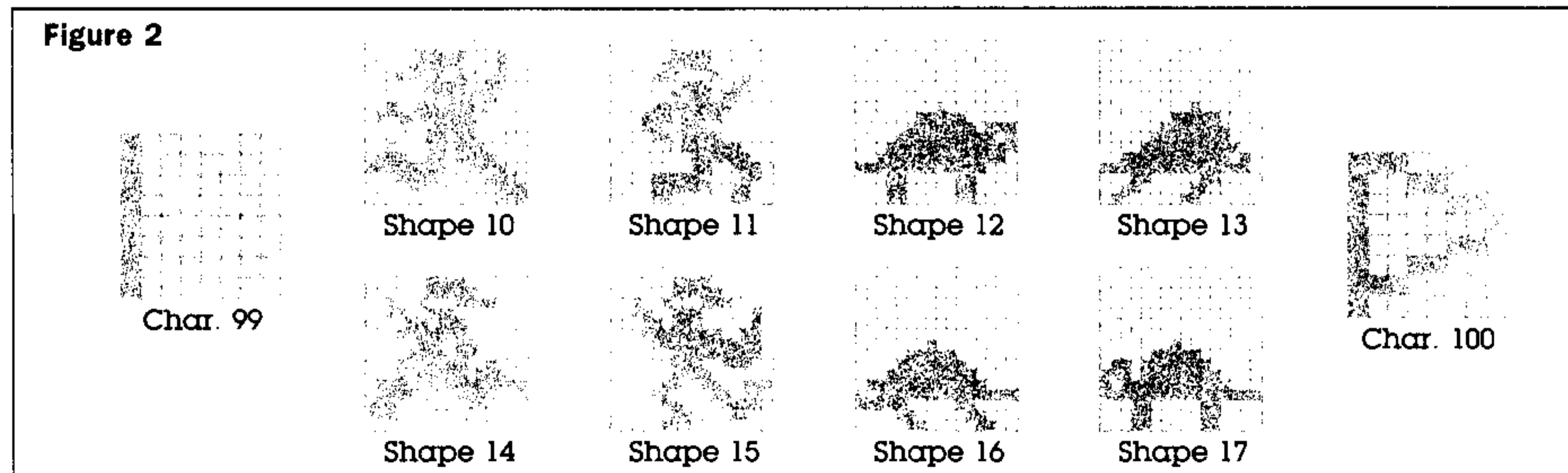


Figure 1

**"Now I don't know about you,
but I for one found
Zeno's conclusion to be
a bit extreme!"**



INDUSTRY WATCH

SURPLUS TI-99/4A PARTS APPEARING IN RADIO SHACKS

Tandy seems to have snapped up a good supply of surplus parts for the TI-99/4A—parts that were sold when TI left the home computer market. Available now through the Radio Shack catalog are video modulators with a cable assembly (just \$4.95), power supply boards (\$4.95), and genuine tan 99/4A keyboards (only \$2.95). Imagine replacing half a computer for only \$13 . . .

THE FALL AND RISE OF IBM'S JUNIOR

Although the dust around IBM's March announcement to discontinue production of its PCjr has settled, speculation over the impact of its withdrawal from the market continues to stir things up. Many industry analysts believe that IBM pulled PCjr to make way for the introduction of a "PC II" and the corresponding price cuts on the existing line that would effectively bring the price of the current PC closer to Junior's original price. The PCjr departure also leaves Apple relatively free to clinch its hold on the school market, but it may benefit only temporarily in the home market. Once PCjr production stops, IBM may offer it at fire-sale prices to clear out its existing inventories (reportedly to total another 150,000-300,000 units), thus undercutting the Apple II Family computers. These price cuts (and IBM's promise to fully support the machine with software and parts) are good news for PCjr's installed and potential user-base. In addition, several third-party peripheral vendors have stated their intentions to feed off the machine for at least another year. One firm, Racore Corp., has even introduced its "ATcessory," adding a 1.2 megabyte floppy disk drive to the PCjr, and a local area network that is said to link up to 16 PCjrs, PCs, XTs, and ATs. Smart money is betting on Big Blue re-entering the home market with a lower priced machine in fourth quarter.

FAMOUS FACES ENTER COMPUTER HARDWARE & SOFTWARE ARENA

General Electric, Boeing Computer Services, and Encyclopedia Britannica, all heavyweight players in electronics and information, are now taking their first steps in the personal computer field. Banking on its name and reputation for quality in appliances, General Electric has decided to enter the peripheral market. The company recently introduced a thermal printer, a modem, and two monitors. Rather than offering the industry's traditional 90-day warranty, GE is offering 2-year warranties. In another arena, Boeing Computer Services has released a three-dimensional spreadsheet called *Boeing Calc* that is supposedly similar to Lotus' 1-2-3. It will run on micros and mainframes. Finally, Encyclopedia Britannica has acquired MSA's Educational Business Unit, which consists of DesignWare and Edu-Ware Services. DesignWare will sell its own products plus the Edu-Ware line, and will reportedly operate as part of Britannica Learning. Now, if GE, Boeing, and Britannica ever join forces, we may actually get to see a computerized dishwasher that writes term papers and runs *Flight Simulator*.

COMMODORE FACES TRIUMPHS AND PROBLEMS WITH AMIGA

Commodore International Ltd. is working hard at customer relations by expanding its service and support capabilities, while enjoying the unexpected success of its Commodore 16 in Mexico. Behind the scenes, however, the company is reportedly struggling with unhappy Amiga managers and shareholders, and a delayed production schedule for its Macintosh-like Amiga computer. Commodore is also expected to post substantial losses in its third and fourth quarters. Even so, Commodore's regional service centers are scheduled to increase from 650 to 1,000 by May 1. A toll-free hotline, and an information service for its users via CompuServe are also being set up. In Mexico, where the market lags behind that of the U.S. by 2 to 3 years, the little \$170 Commodore 16 has made its way into thousands of homes after having flopped in the U.S. Meanwhile, Commodore's stock has tumbled by about 50% since the company acquired Amiga Corp., leaving Amiga shareholders angrily holding the bag. To smooth things over, Commodore has reportedly promised to award more than \$7 million in bonuses if production on the crucial Commodore Amiga computer meets its summer schedule—a timetable calculated to offset the parent's cash problems with early sales.

TO ST, OR NOT TO ST — THAT IS THE QUESTION

While Atari scrambles to raise enough cash to remain solvent and develop new products, decisions are being made regarding what may be its saving grace—its ST series of computers. Two underlying operating systems are currently being tested in ST prototypes. The more sophisticated of the systems would allow GEM, part of the overall operating system, to take advantage of the system's capability to set up hierarchical files on a hard disk, and provide color. The ST line will imitate Apple's Macintosh (which has no color), but will be marketed at prices low enough to cause both IBM and Apple some concern in the consumer market. If the ST series can, in fact, be shipped in sufficient quantity this year—without any major product flaws—it stands a good chance of keeping the phoenix-like Atari Corp. flying high. In a recent development, however, Atari's announcement of withdrawal from exhibiting at June CES casts serious doubt upon their ability to deliver.

PRICE WARS HEAT UP AGAIN ON NEW FRONTS

The introduction of Atari's and Commodore's new machines this summer is expected to ignite another price war on the computer battleground—but others have already fired the first shots. Desk-organizer programs, which entice users with resident utilities such as appointment calendars, notepads, calculators, etc., have recently experienced cuts ranging from 30 to 50 percent. Prices are hovering in the \$50 to \$100 range, and have not yet stabilized. Dot-matrix and daisy-wheel printers also have been rapidly tumbling in price, while increasing in performance and features. Keyboard macros are also following suit: Software Research Technologies has declared war on Borland International by dropping the price of its keyboard-redefinition program, *Smartkey*, to \$49. *Superkey*, Borland's similar program, retails for \$69. The only sure victor of these price wars will be the consumer.

A TURNING POINT FOR BIG RED?

Apple Computer has been experiencing a number of growing pains lately with the departure of key engineers and executives, poor morale in its Apple II division, and new-product worries. As Apple management began operating more and more like a Fortune 500 company, it opted to wedge its "favorite fruit," the Macintosh, into similar corporate environments—while paying less and less attention to its "cash cow" Apple II Family development. As a result, many disgruntled engineers and middle managers from the less-glamorous Apple II division, feeling slighted and ignored, left for greener pastures elsewhere. Since co-founder Steven Wozniak sold more than 3 million shares of his Apple stock and joined the exodus, morale reportedly eroded even further. In addition, Apple's long-awaited flat-panel LCD screen for the IIc was universally panned for its small, hard-to-read display. In spite of the problems, Apple has released an enhancement kit for the IIe that provides (among other things) full compatibility with the IIc, improved mouse response, and the ability to enter BASIC commands in upper and lower case. Big Red is also said to be working on a "Turbo Mac," expected to support 2 megabytes of RAM; and, by 1986, a next-generation, 16-bit Apple II machine.

RUMORS SUPPORT NOTION ABOUT TI-99/8 CLONE

Well-substantiated rumors persist that a TI-99/8 "clone" is being beta-tested now, and that it will make an appearance at the Summer Consumer Electronics Show in Chicago. The East-Coast manufacturer is said to be negotiating with Texas Instruments to use some of the 99/4A's proprietary operating system. The new machine reportedly uses the 9995 chip, and will come with 128K RAM expandable to 512K. It will have 16K VDP RAM expandable to 64K. All expansion will be through the TI Peripheral Expansion Box. The video display is expected to support composite and RGB monitors, and may offer a variety of high-resolution options, including 80 columns. Look for a \$300 price range for a minimum configuration that can run existing 99/4A software and make use of existing peripherals.

Offer Expires June 30th 1985

SPECIAL LIMITED-TIME GET-ACQUAINTED OFFER

SAVE \$14.50[†] Over Single-Issue Price



Enter or Renew a 10-issue* ON DISK™ or ON TAPE™ Program Subscription For Only \$45 and Receive Your Choice of One Gift Shown Below

† Program subscription price increase goes into effect after June 30, 1985

*10 issues will be added on to any remaining issues you might still be entitled to because of a current program subscription or media given as premiums for subscribing to Home Computer Magazine.

For Apple II Family, C-64, IBM PC & PCjr Users

ON DISK Revue™ Volume 1

- 10 Programs On A Ready-To-Run Disk
- Exciting Activities For The Entire Family—At An Affordable Price
- Entertainment, Productivity & Education Plus Valuable Programming Secrets

Valuable Proof-of-Purchase Coupon For Gift Redemption



An Attractive Library-Storage Slipcase

CONTENTS

1. Boolean Brain
2. Electronic Home Secretary
3. Savings Planner
4. Snap-Calc
5. Tablut
6. Wild Kingdom
7. Musical Mystery Words
8. Market Madness
9. Tower of Hanoi
10. Missile Math

64-page Full-Color Book



Disk Stored in Re-usable Protective Pouch

A \$14.95 Premium Value!

For TI-99/4A Users



Your Choice of One Gift

4 Tapes or 4 Disks from 1 Group Below:

Group A December, 1982 to March, 1983	Group B April, 1983 to July, 1983	Group C August, 1983 to November, 1983
---	---	--

OR

6 Magazines from 1 Group Below:

Group 1 December, 1982 to May, 1983	Group 2 June, 1983 to November, 1983
---	--

OR

Best of 99'er—The 360-Page "Classic" Reference Book.

An \$11.95 to \$19.95* Premium Value!

* 4 Tapes or Disks, \$14.95; 6 Magazines, \$11.95; Best of 99'er, \$19.95

Yes, please enter my 10-issue program subscription immediately. My computer type is:

Apple II Family IBM PC IBM PCjr C-64 (disk) C-64 (tape)*

* NOTE: C-64 tape subscribers will receive as their premium, the disk-based ON DISK Revue Volume 1.

Please Print Name _____

Address _____

City _____ State _____ Zip _____

Check or Money Order Enclosed VISA MasterCard Date Expires _____

MUST BE IN U.S. FUNDS DRAWN ON A U.S. BANK

Account No. _____

Tel. No. _____ Signature _____

Enclose payment or credit card information & mail with completed form to:

Emerald Valley Publishing Co.
P.O. Box 70288 • Eugene, OR 97401

Or use our TOLL-FREE Order Line for VISA/MasterCard orders only:
1-800-828-2212 (Minimum \$10 Order)
In Oregon, Alaska, Hawaii Tel. (503) 485-8796

Please allow 4-6 weeks for your first issue.

Satisfaction Guaranteed—or the unfilled portion of your subscription will be refunded, less the cost of any premiums you have received. Prices Subject To Change Without Notice. Defective media gladly exchanged. NO REFUNDS on shipped media. PSV1530485

—TI-OWNERS ONLY—

Select Your Premium Below:
(Mark only one (1) box for your choice)

Either

99'er Home Computer Magazine[†]

	ON DISK	ON TAPE
Group A	<input type="checkbox"/>	<input type="checkbox"/>
Group B	<input type="checkbox"/>	<input type="checkbox"/>
Group C	<input type="checkbox"/>	<input type="checkbox"/>

OR

6 Issues of 99'er Home Computer Magazine[†]

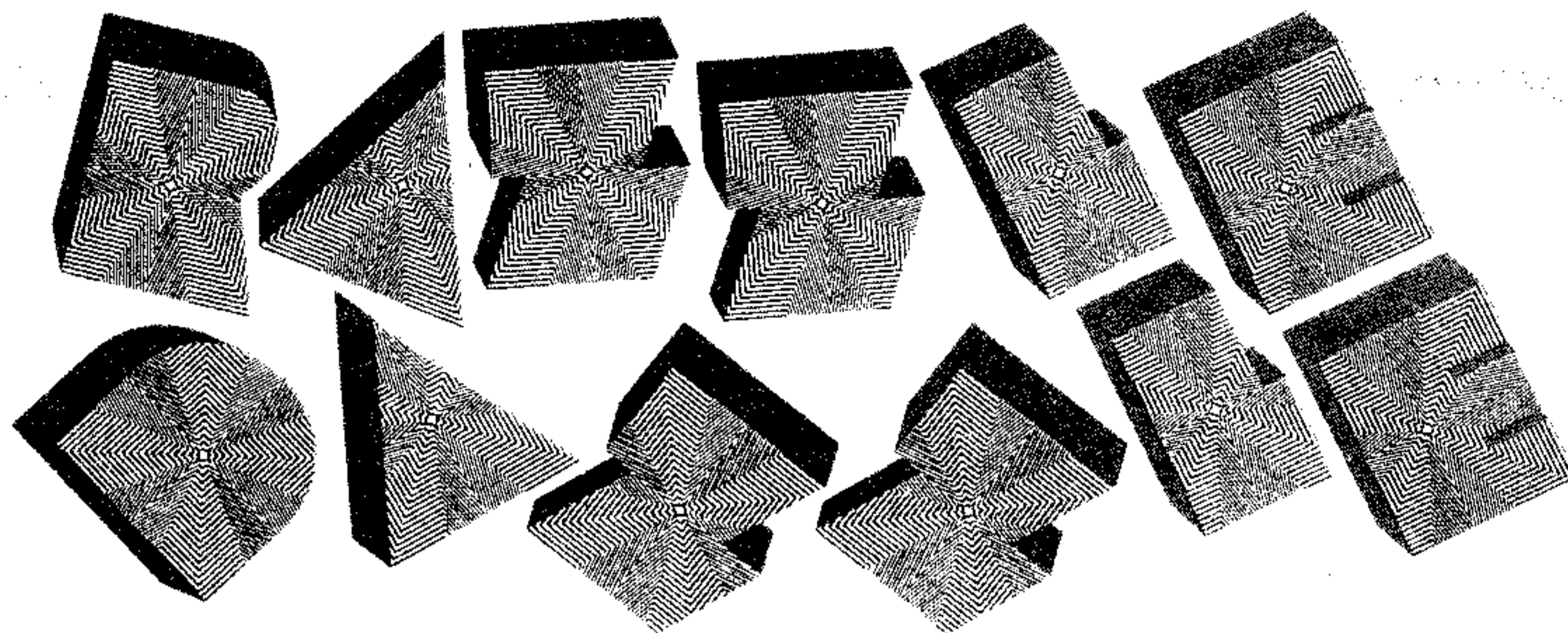
- Group 1
 Group 2

OR

Best of 99'er

[†]For editorial and program information, refer to the inside back cover of this magazine.

Canada add \$5 for software subscription. Software subscription not available in other countries at this time.



by **William K. Balthrop**
HCM Staff

**Want to create beautiful 3-D scenery without mountains of code?
 This *SPRITEly* tutorial shows you how . . .**

Ever wonder how programmers get the computer to generate three-dimensional graphics in a game? Does it take mountains of computer code to, say, make mountains on the screen? Does it have to be written in machine language? The answer to both of the last two questions is *no*. TI Extended BASIC can do all of this, and more.

With the use of sprites, you can create an almost unlimited number of special effects. For the most part, sprites have been used in computer games as the focus of attention: the alien spaceship, your own tank, flying asteroids . . . But sprites can play a big role in helping to create a more realistic background, as well.

By now you have seen many games in which the background moves in order to create the illusion of motion, while the focus of attention—the object under your control—is almost stationary. In order to do this with normal characters on a large scale, you would probably still need to resort to machine language. However, by using sprites, you can create this effect on a limited scale.

Simulating horizons which move past as you travel down the highway is really quite simple. In fact, you can add several "layers" to the horizon that all move at different speeds. Try looking out the side window of your moving car while sighting past your finger and watching the view outside go by. Objects farther away from you will seem to pass slower than closer objects. This effect is easily transferable to sprites by giving those sprites "farther back" on the horizon slower speeds.

These are only a few of the tricks that sprites are capable of performing. To reinforce the three-dimensional effect, you can also use the fact that "lower-numbered" sprites pass in front of "higher-numbered" sprites. (When creating a sprite in TI Extended BASIC, you assign it a number between 1 and 28.)

Only one drawback hampers the use of sprites—the maximum number of visible sprites on any horizontal row (row meaning individual pixel rows) of the screen is four. If there are more than four sprites on any screen row, those four sprites with the lowest sprite numbers will remain visible, while sprites with higher sprite

numbers will become invisible. Under certain circumstances, this can benefit special effects creation, but it does not help us here where we need to be able to create an entire horizon across the screen.

The best solution is to enlarge the sprites to their maximum magnification of 4 by invoking `CALL MAGNIFY(4)`. This creates a sprite that is four characters high by four characters wide. With four of these magnified sprites you can cover 16 characters across the screen. It is thus possible to cover one-half of the screen's width with these four sprites to create our horizon. Then we fill-in the gaps between each sprite where the background shows through by placing normal characters of a certain color behind like-colored sprites. When the sprites move, it's as if an entire section of the screen is moving.



The Program

In *Horizon*, we actually use two animation tricks. We use sprites as mentioned above, but we also use normal character animation to a limited extent. This is because a problem resides in getting the dashed lines on the highway to scroll by along with the horizon. There are too many dashed lines to do this with

our limited number of sprites. Thus, we will settle for a less elegant kind of animation in trade for the increased feeling of motion that the program will evoke.

The sprites will be set to a magnification of four. Because sprites get their shape from the redefinable characters of the TI computer, we must define the graphics shapes of four individual characters for the overall shape of each sprite. Taking this into consideration, we will define the shapes of eight characters to create two sprite shapes (four character shapes per sprite shape). These two sprite shapes will be used for all of the mountains in the background. Fortunately, we can save some overhead in the program because 2 or more sprites can share the same four-character shape definition. Sprites 1, 3, 5, and 7 all use the same shape. The shape is defined in the four ASCII characters 96, 97, 98, and 99 to form one sprite shape. Sprites 2, 4, 6, and 8 use character shapes 100, 101, 102, and 103 to define their sprite shape. The four characters used in a sprite shape must be *contiguous*. In other words,



they must all be in a four-character group (e.g., 64, 65, 66, and 67), and must start with a character whose ASCII value is evenly divisible by 4.

We will also need to define sprite shapes for the car, which will remain stationary on the screen. The last 2 sprite shapes we need are for the road intersection and the billboard, both of which "move" alongside the road.

Line 200 contains a FOR-NEXT loop which reads the graphics information for each character from DATA statements and then assigns that information to the proper character. The variable A\$ is used later to help animate the lines in the road with standard character animation. Line 210 assigns color to the four character groups we will be using. Notice that only one CALL COLOR statement is required. You can assign as many color groups as you like, up to the legal line-length limit.

Line 220 will paint part of the background with solid colors. This sets the stage for the sprites. Earlier we stated that because the sprites can only cover one-half of the screen, we need to place background characters of the same color behind them.

"By now you have seen many games in which the background moves in order to create the illusion of motion . . . by using sprites, you can create this effect on a limited scale."

Line 230 and 240 will paint a dashed line down the road. The line is actually a solid line of four different characters repeated again and again. Only one character at a time, however, will contain a shape that can be seen on the screen. The other characters are assigned a shape with no pixels turned on—so they are invisible.

Line 250 sets the magnification factor for the sprites to 4. As discussed earlier, this will cause the sprites to reach their maximum size.

Lines 260, 270 and 280 each consist of only one statement: CALL SPRITE. However, these three lines are responsible for the initialization, placement, and speed of all 11 sprites. Line 260 will assign sprite numbers 5, 6, 7, and 8. These sprites will each use one of the two mountain shapes, and will all be placed on the screen to appear as the mountains closest to the viewer. Line 270 places the next four mountains on the screen, farther away from the viewer and moving more slowly. Finally, line 280 places the car, intersection, and billboard on the road.

The final character animation takes place in lines 290 through 350. The variable X will continue counting from 1 to 4 as long as you let the program run. This controls the location to which the ON GOSUB statement in line 310 will branch. Each of the statements in lines 320 to 350 cause the shapes used for the dashed line on the highway to shift. The shape is cleared from one character and added to another.

HCM

Horizon (TI-99/4A) Explanation of the Program

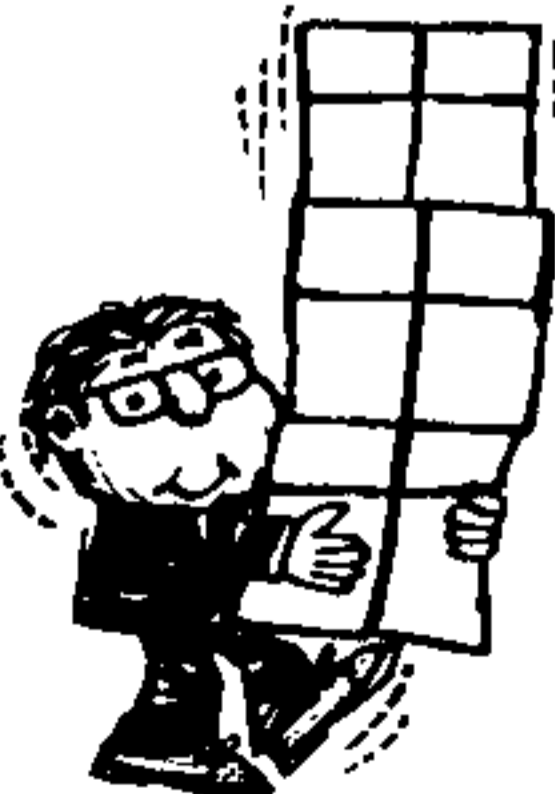
Line Nos.	Description
100-190	Program header.
200-210	Initialize program graphics and color.
220-240	Display static character graphics.
250-280	Place sprites on the screen.
290-350	Animate the dashed line on the highway.
360-380	Character-graphics shape data.

XB Horizon requires TI Extended BASIC.

```

100  ** * H O R I Z O N * * *
110  * * * * *
120  * * * * *
130  I I C O P Y R I G H T 1 9 8 5
140  I E M E R A L D V A L L E Y P U B L I S H I N G C O .
150  I B Y W I L L I A M K . B A L T H R O P
160  I H O M E C O M P U T E R M A G A Z I N E
170  I V E R S I O N 5 . 3 . 1
180  I T I E X T E N D E D B A S I C
190  I
200  C A L L C L E A R : : F O R Z = 9 6 T O 1 1 6 : : R E
C A D A $ : : C A L L C H A R ( Z , A $ ) : : N E X T Z :
: A $ = " 0 0 0 0 0 0 0 0 F F "
210  C A L L C O L O R ( 1 1 , 1 6 , 2 , 1 2 , 2 , 2 , 1 3 , 1 3 , 1 3 ,
1 4 , 4 , 4 )
220  C A L L H C H A R ( 1 3 , 1 , 1 3 6 , 6 4 ) : : C A L L H C H A
R ( 1 5 , 1 , 1 2 8 , 1 6 0 ) : : C A L L H C H A R ( 2 0 , 1 , 1
2 0 , 9 6 ) : : C A L L H C H A R ( 2 3 , 1 , 1 2 8 , 6 4 )
230  X = 1 : : F O R Z = 1 T O 3 2 : : X = X + 1 : :
I F X > 1 1 5 T H E N X = 1 1 2
240  C A L L H C H A R ( 2 1 , Z , X ) : : N E X T Z
250  C A L L M A G N I F Y ( 4 )
260  C A L L S P R I T E ( # 5 , 9 6 , 1 3 , 8 3 , 1 , 0 , - 6 , # 6 , 1
0 0 , 1 3 , 8 3 , 5 5 , 0 , - 6 , # 7 , 9 6 , 1 3 , 8 3 , 1 2 0 , 0 ,
- 6 , # 8 , 1 0 0 , 1 3 , 8 3 , 2 0 0 , 0 , - 6 , # 9 , 9 6 , 1 3 , 8 3 ,
2 2 8 , 0 , - 2 , # 2 ,
1 0 0 , 4 , 6 7 , 3 5 , 0 , - 2 , # 3 , 9 6 , 4 , 6 7 , 1 0 0 , 0 ,
2 , # 4 , 1 0 0 , 4 , 6 7 , 2 0 0 , 0 , - 2 )
270  C A L L S P R I T E ( # 1 , 9 6 , 4 , 6 7 , 2 2 8 , 0 , - 2 , # 2 ,
1 0 0 , 4 , 6 7 , 3 5 , 0 , - 2 , # 3 , 9 6 , 4 , 6 7 , 1 0 0 , 0 ,
2 , # 4 , 1 0 0 , 4 , 6 7 , 2 0 0 , 0 , - 2 )
280  C A L L S P R I T E ( # 9 , 1 0 8 , 2 , 1 6 1 , 2 4 0 , 0 , - 1 0 ,
# 1 0 , 1 0 4 , 1 5 , 1 4 3 , 1 2 0 , # 1 1 , 1 1 6 , 5 , 1 2 1 , 2 0
0 , 0 , - 1 0 )
290  X = 0
300  X = X + 1 : : I F X = 5 T H E N X = 1
310  O N X G O S U B 3 2 0 , 3 3 0 , 3 4 0 , 3 5 0 : : G O T O
3 0 0
320  C A L L C H A R ( 1 1 2 , " 0 " , 1 1 5 , A $ ) : : R E T U R N
330  C A L L C H A R ( 1 1 5 , " 0 " , 1 1 4 , A $ ) : : R E T U R N
340  C A L L C H A R ( 1 1 4 , " 0 " , 1 1 3 , A $ ) : : R E T U R N
350  C A L L C H A R ( 1 1 3 , " 0 " , 1 1 2 , A $ ) : : R E T U R N
360  D A T A 0 , 0 0 0 0 0 0 7 1 F 3 F 7 F F F , 0 , 0 0 0 0 C 0 F 8 F
C F E F F F F F , 0 , 0 7 0 F 1 F 3 F 3 F 7 F F F F , 0 , 0 0 C 0 F 0
F 8 F C F E F F F F
370  D A T A 0 , 0 0 0 7 0 A 7 A 7 F F F 7 0 2 0 , 0 , 0 0 8 0 4 0 3 E F
E F E 1 C 0 8 , 0 , 1 E 1 E 1 F 1 F 1 E 1 E 1 F 1 F , 0 , F 0 F 0 F 0
F 0 F 0 F 0 F 0 F 0 , 0 0 0 0 0 0 0 0 F F
380  D A T A 0 , 0 , 0 , E 0 F C F F F F F F F F F F 9 F 8 3 8 0 8 0 8 0
0 0 0 0 0 0 0 0 0 0 0 0 C 0 F C F F F F F F F F F 1 F 0 3 0 1 0 1 0
1 0 1 0 1

```



MOVING?

Don't Miss Out On Any Issues Of

HOME COMPUTER

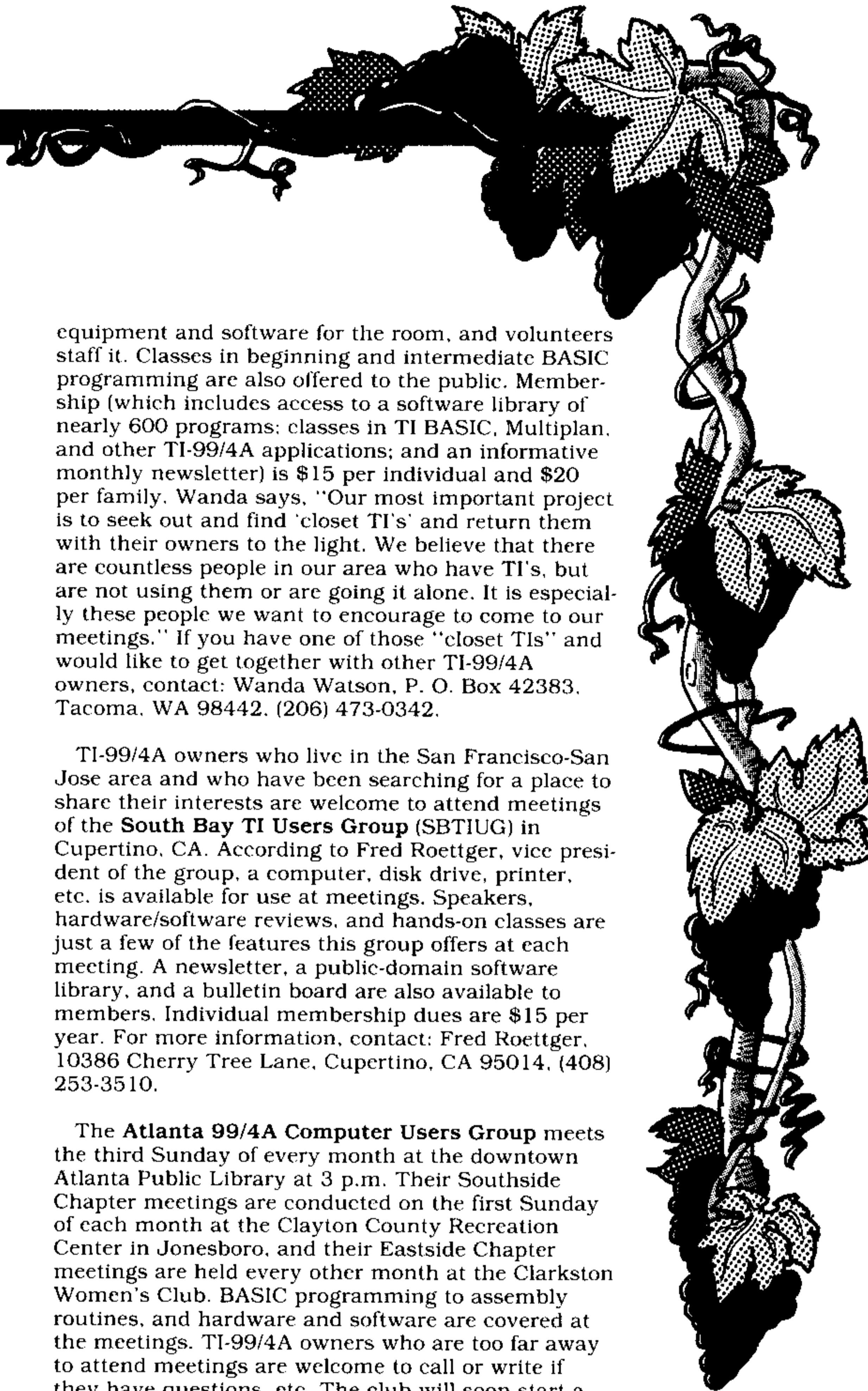
magazine

Send us a Change-of-Address Card (available at any Post Office) 6-8 weeks prior to the move.

Be sure to include both the old & new address, plus the alphanumeric code above your name on the mailing label.

Home Computer Magazine
P.O. Box 70288
Eugene, OR 97401

Please send this information to:



busy receiving calls from all over the country, as well as from overseas. Speaking of overseas, this club also has members from Germany and Italy. Dues are \$10 per year, which includes an excellent newsletter and access to the club library of 42 double-sided disks of public-domain software. For more information, contact: Ben Blackstock, 385 Collins Road NE, #201, Cedar Rapids, IA 52402, (319) 393-5416.

Formed on November 28, 1984, the **Mohawk IBM PCjr Users Group** in Albany, New York is growing at a steady pace every month and now has 30 active members, according to spokesperson Gary Hermansen. Most of the membership is interested in business applications such as word processing and data bases for the PCjr. This group puts out a monthly newsletter and has a public-domain library. If you are a PCjr owner looking for others of like species, contact: Gary Hermansen, P. O. Box 12-305, Albany, NY 12212, (518) 869-9787.

Group Grapevine stepped into its own backyard and spoke to Jim Cox of the **Eugene IBM PC Club**, Eugene, Oregon. The Eugene club is not limited to PC owners, but welcomes PCjr and IBM-compatible owners as well. Meetings, on the third Thursday of each month, consist of a half-hour of questions and answers, a panel discussion, a speaker, and a demonstration of software or hardware. Dues are \$20 per year, and include access to their disk library and a subscription to their monthly newsletter. For more information, contact: Jim Cox, P. O. Box 5070, Eugene, OR 97405.



South, Mobile, and Alabama User's Group (SMAUG) of Grand Bay, Alabama meets on the second and fourth Tuesdays of every month at 7 p.m. in the University of South Alabama Engineering Department building to provide technical assistance to owners of the TI-99/4 and 4A computers. By pooling their expertise they can assist in solving programming problems, and will share ideas, demonstrate equipment, and discuss new software for and uses of the TI home computer. Membership fees are \$10 a year for individuals and \$15 for families. Benefits include a newsletter and access to their public-domain library. If you are a TI-99/4A owner living in this area and would like the fellowship of people with a like interest, contact: Daniel L. Parrott, Route 2, Box 169-B, Grand Bay, AL 36541, (205) 865-6686.

The president of the **Tacoma Eastside 99ers Users Group**, Wanda Watson, recently wrote to Group Grapevine with information regarding their group. Beginning with just 5 members one year ago, they have grown to nearly 50 members today. Besides offering support to members, it is the goal of the group to promote computer literacy in the community. In October 1984 they opened a free, public-access computer room. Group members donated or loaned

equipment and software for the room, and volunteers staff it. Classes in beginning and intermediate BASIC programming are also offered to the public. Membership (which includes access to a software library of nearly 600 programs; classes in TI BASIC, Multiplan, and other TI-99/4A applications; and an informative monthly newsletter) is \$15 per individual and \$20 per family. Wanda says, "Our most important project is to seek out and find 'closet TI's' and return them with their owners to the light. We believe that there are countless people in our area who have TI's, but are not using them or are going it alone. It is especially these people we want to encourage to come to our meetings." If you have one of those "closet TIs" and would like to get together with other TI-99/4A owners, contact: Wanda Watson, P. O. Box 42383, Tacoma, WA 98442, (206) 473-0342.

TI-99/4A owners who live in the San Francisco-San Jose area and who have been searching for a place to share their interests are welcome to attend meetings of the **South Bay TI Users Group** (SBTIUG) in Cupertino, CA. According to Fred Roettger, vice president of the group, a computer, disk drive, printer, etc. is available for use at meetings. Speakers, hardware/software reviews, and hands-on classes are just a few of the features this group offers at each meeting. A newsletter, a public-domain software library, and a bulletin board are also available to members. Individual membership dues are \$15 per year. For more information, contact: Fred Roettger, 10386 Cherry Tree Lane, Cupertino, CA 95014, (408) 253-3510.

The **Atlanta 99/4A Computer Users Group** meets the third Sunday of every month at the downtown Atlanta Public Library at 3 p.m. Their Southside Chapter meetings are conducted on the first Sunday of each month at the Clayton County Recreation Center in Jonesboro, and their Eastside Chapter meetings are held every other month at the Clarkston Women's Club. BASIC programming to assembly routines, and hardware and software are covered at the meetings. TI-99/4A owners who are too far away to attend meetings are welcome to call or write if they have questions, etc. The club will soon start a Lending Library where books that the club has purchased can be checked out by members at the monthly meetings. For more information, contact: Marshall Gordon, P. O. Box 19841, Atlanta, GA 30325.

Information just blew in from **The Windy City 99 Club** of Chicago, Illinois. These TI-99/4A owners meet the third Thursday of every month at 7:30 p.m. and offer as well as a user's library featuring business, educational, and game software. If you would like to learn more about this club (established in 1982), write: Michael Mickelsen, 640 N. LaSalle Street, Suite 280, Chicago, IL 60610.

HCM

Speeding Up A BASIC Program

PART ONE

by John P. Russo
and the HCM Staff

Program execution speed is a popular topic of interest for many microcomputer users. The process of measuring the speed of one program or system against another is called "benchmarking." It involves running and timing two functionally similar pieces of software and calculating their differences in execution time. It is a valuable tool in comparing the performance of different machines, but otherwise it has little practical value.

In this article, we'll discuss ways of actually improving program performance, and use benchmarking to compare machines *against themselves*. We will present a number of rules which can be applied to programs in almost any version of BASIC. However, the times given and the examples discussed here will deal with Apple II (Applesoft) BASIC, IBM PC DOS 2.0 BASIC (Disk), Commodore 64 BASIC, and TI-99/4A Extended BASIC. We chose TI Extended BASIC rather than console BASIC so that maximum compatibility could be maintained between the different BASICs.

Let us state at the beginning that program speed is not always critical, and other factors—such as readability or memory space—may be more important. However, there are situations where fast execution is the most important concern. For example, speed is often critical in game programs. Also, certain communications applications require speed so that data won't be lost.

Before getting into the specifics of speeding up programs, let's consider what happens when a user types RUN when a simple program, such as *Program 1*, is in your computer's memory:

Program 1

```
10 LET N = 0
15 REM *** BEGINNING OF LOOP
20 LET N = N + 1
25 IF INT (N/7) = N/7 THEN PRINT N;
30 REM THIS LOOP CHECKS EACH NUMBER THAT IS LESS
35 REM THAN 1000 TO SEE IF IT IS DIVISIBLE BY 7
40 REM IF SO, THE NUMBER IS PRINTED.
45 IF N < 1000 THEN GOTO 20
50 REM *** END OF LOOP
```

Line 10 is, of course, written in BASIC, a language which many humans understand, but which is not the native tongue (machine language) of a microcomputer. Before a computer can execute the instruction on line 10, it must be provided with an equivalent instruction written in its native tongue. Rather than investigate all the details of interpreted BASIC, we can view the execution of a BASIC statement as a two-step process, namely:

1. Translate the BASIC statement into machine language.
2. Execute the machine-language statement.

The major disadvantage of interpreted versions of BASIC is that step 1 consumes so much time. Some lines, such as those within loops, must be translated many times during the execution of a program.

Let's take a closer look at *Program 1*. The program doesn't do anything very exciting—it merely prints all numbers which are less than 1000 and (evenly) divisible by 7. It is a short program—excluding remarks it has only 4 lines. Despite this program's brevity, we will find that there are many techniques which can be applied to this program to speed it up.

It seems reasonable that our efforts to decrease execution time should center on loops, because they are the primary time-consumers in a running program. This suggests that if we wish to speed up *Program 1*, we should look more closely at the loop in lines 20 to 45. The REM statements within this loop may have caught your eye. Most of us acknowledge (perhaps grudgingly) that program remarks are important. However, when they're placed within loops, they can lengthen execution time by a surprising amount. If remarks must be included in the working version of your program and speed is essential, then make sure the remarks are not within loop bodies. Behold: we have our first "speed-up" rule:

RULE 1: Remove REMark statements from program loops.

By applying this rule to our program, we get *Program 1.1*. The new program runs significantly faster than the old one on three machines, and on the IBM PC it actually runs 25 percent faster. The REM statements on line 15 and 45 have not been removed, because their removal would not have saved a significant amount of time. Note also that although variables in BASIC are initialized to 0 at run time, deleting the apparently unnecessary line 10 would not yield any worthwhile time savings.

Program 1.1

```
10 LET N = 0
15 REM *** BEGINNING OF LOOP
20 LET N = N + 1
30 IF INT (N/7) = N/7 THEN PRINT N;
40 IF N < 1000 THEN 20
45 REM *** END OF LOOP
```

In general, how much time can be saved by removing REMarks? Naturally, any reduction in time depends on the number of remarks in the loop and the number of times the loop is executed. But savings also depend on the *length* of the REM statements. Figure 1 illustrates the time required to "execute" short (3 characters) and long (70 characters) REM statements.

In many versions of BASIC, loops can be constructed in just two ways: with GOTO statements, and with



**Want to add more snap
to your BASIC programs?
This two-part tutorial tells you
not only HOW, but HOW MUCH
execution time you can save
by following our "rules of thumb."**

FOR-NEXT statements. (IBM disk BASIC for the PC and cartridge BASIC for the PCjr also provide WHILE loops.) In *Program 1* we used a GOTO statement, but clearly, a FOR-NEXT loop could have been used in the manner listed here:

```
PROGRAM 1.2
20 FOR N = 1 TO 1000
30 IF INT (N/7) = N/7 THEN PRINT N;
40 NEXT N
```

Notice that this program is a bit shorter than *Program 1.1*. It also runs substantially faster. In fact, on the Apple and Commodore machines, this program executes about 36 percent faster. On the TI and IBM computers, the improvement is not as great (25 percent faster on the former, and 19 percent on the latter), but there is still a noticeable difference. Of course, improving program speed sometimes results in a longer and/or less-readable program. However, replacing GOTO loops with FOR loops actually yields shorter, more readable programs. We then have our second rule:

RULE 2: Whenever possible, use FOR-NEXT loops instead of loops constructed with GOTO statements.

To get a clearer picture of the time differences, it is helpful to compare the execution times of "empty" loops. This measures the time overhead of the loop structure itself. *Loop 1* and *Loop 2* below are the empty loops taken from *Program 1.1* and *Program 1.2*. Figure 2 compares their execution times.

<pre>Loop 1 10 N = 0 15 N = N + 1 20 IF N < 1000 GOTO 15</pre>	<pre>Loop 2 10 FOR N = 1 TO 1000 20 NEXT N</pre>
---	--

Using a conditional GOTO when branching is usually faster than using an unconditional GOTO. For example, the next loop uses an unconditional branch in line 30 and requires slightly more execution time. Incidentally, the WHILE loop available in IBM BASIC also runs slightly slower than *Loop 1*, so RULE 2 applies to WHILE loops as well as to loops built with GOTO statements.

```
Loop 3
10 N = N + 1
20 IF N > 1000 GOTO 40
30 GOTO 10
40 REM PROGRAM CONTINUES ON THIS LINE
```

We can speed up execution even more if we omit the N of the NEXT N, to get the loop 4. (TI-99/4A users note: this option is not possible in TI's BASICs.) When the

In this article, we will assume that you wish to write more efficient programs in an interpreted version of BASIC. An important alternative to interpreted BASIC is *compiled BASIC*. A BASIC compiler translates the entire BASIC code into executable machine language *before* the program is RUN, and creates what is called a *run-time module*. If a compiled version of BASIC is used, then a program line translation is carried out only once (during the compilation of the program) so at run-time, no further translation is required. It follows that compiled programs run much more quickly than interpreted programs.

Even faster execution speeds are possible by writing efficient assembly-language programs. For many persons, however, the methods of increasing speed that we present in this article will have to suffice—because they do not have access to a compiled language, and either do not know or do not wish to use an assembly language.

NEXT statement references the variable, the computers take longer to execute the loop.

```
Loop 4
10 FOR N = 1 TO 1000
20 NEXT
```

By not specifying the N variable in *Loop 4*, we eliminate the computer's need to search for a variable, and decrease execution time. The savings are sufficient to justify stating our next rule:

RULE 3: FOR-NEXT loops execute more quickly if the control variable is not stipulated in the NEXT statement.

If we apply RULE 3 to *Program 1.2*, we get the following program and a 2- to 3-percent speed increase. In a while, we'll see that if we apply this rule to a loop within a more complex program, the time savings could be much greater.

```
Program 1.3
20 FOR N = 1 TO 1000
30 IF INT (N/7) = N/7 THEN PRINT N;
40 NEXT
```

We applied 3 rules, and now have a program which is only three lines long. However, two more improvements can be made! The first of these emerges

FIGURE 1
Time (milliseconds) Required to "Execute" One REM Statement.

	Apple II	IBM	C-64	TI-XB
Short REM	0.3	0.3	0.3	.5
Long REM	2.2	2.8	2.5	.7

FIGURE 2
Comparing "Empty" Loops

	Apple II	IBM	C-64	TI-XB
Loop 1	9.2	5.0	10.0	12.0
Loop 2	1.2	1.3	1.3	4.7

from the observation that the division in line 30 is performed twice. Perhaps the program will run more quickly if we can find a way to do it just once. But that is easy—just save the result of the first computation! When we do this, we obtain *Program 1.4*.

```

Program 1.4
20 FOR N = 1 TO 1000
25 LET CHECK = N/7
30 IF INT (CHECK) = CHECK THEN PRINT N;
40 NEXT

```

On the IBM and TI machines, the speed increase is negligible, but on the other two machines it is a full 10 percent. However, the program is now longer and has an additional variable—we had to trade memory space for time. This exchange of space for time is often required in order to speed up program execution. Deciding whether to make this trade is not always easy, but in any case, we have our next rule:

RULE 4: If the same computation is performed more than once, execution time can sometimes be shortened by saving the result of the first computation and using this saved result, instead of repeating the computation.

By applying RULE 4 to *Program 1.3*, we were able to eliminate one computation, $N/7$, at the expense of adding one assignment statement. Because division is a relatively time-consuming operation on Commodore and Apple computers, we obtained a significant increase in execution speed for those machines. On the IBM PC and TI-99/4A, division computing is more efficient, and our replacement doesn't save much time. RULE 4 works best when it eliminates more than two computations. The replacement of a faster operation, such as "+," should be done with care—some experimentation may be necessary. Some insights along these lines can be obtained by studying Figure 3, which gives the time costs of some common BASIC functions.

FIGURE 3
Time (milliseconds) for Some Common Functions

	Apple II	IBM	C-64	TI-XB
=	1.2	1.6	1.3	3.9
+	1.0	1.0	1.1	2.7
-	1.0	1.1	1.1	2.9
*	3.1	1.3	3.2	3.0
/	3.2	2.2	3.3	6.2
^	46.0	14.7	46.6	14.3
ABS	0.5	0.4	0.6	0.6
ASC	0.9	0.7	0.9	3.4
CHR\$	1.3	1.2	1.4	10.2
COS	24.3	23.3	25.3	72.0
INT	1.3	0.9	1.4	1.2
LEN	1.0	0.8	1.1	2.9
LEFT\$	2.7	1.1	2.9	21.7*
MID\$	4.4	1.7	4.6	*
PEEK	1.3	0.5	4.7	12.8
SQR	49.1	6.9	47.5	62.9
STR\$	7.2	1.9	7.3	14.2
VAL	7.2	17.8	7.6	14.7

Note: "=" is assignment, "^" is exponentiation.
* TI Extended BASIC does not have a MID\$ or LEFT\$ function—but the SEG\$ command is identical to the MID\$ in function and was used here.

The times one obtains in making up such a table are dependent on the operands used. For example, it takes longer to compute "1.23*3.48" than it does to compute "2*3." Another factor which affects the times is whether the operands are constants or variables. The times given above are based on "average" computations, with both operands being variables. For example, the addition problem timed was $A + B$, where A held the value 3, and B held the value 2.45. Recall the program yielded by RULE 4 above (*Program 1.4*).

In many versions of BASIC, time can be saved by restricting the use of constants such as 7 or 3.14159. To accomplish this, the constants are assigned to variables such as S or PI . Then, each time the constant is needed, the appropriate variable is referenced. This is the idea behind RULE 5, which is the first rule that does not work on all 4 machines (it does not work in IBM BASIC).

RULE 5: If a constant is used many times in a program, then store the constant as a variable and use the variable instead.

If we apply this rule to the program above, we get:

```

Program 1.5
10 LET S = 7
20 FOR N = 1 TO 1000
25 LET CHECK = N/S
30 IF INT (CHECK) = CHECK THEN PRINT N;
40 NEXT

```

The speed improvement in this case, even for the Apple and Commodore machines, is only 2 percent. However, the savings from this rule can be substantial. For example, if we compare the execution times of the two segments below (see Figure 4), we find that RULE 5 is quite effective on the Apple or Commodore machines, and to a lesser extent on the 99/4A. Both segments compute the area of a circle 1000 times.

Before RULE 5	After RULE 5
30 FOR N = 1 TO 1000	10 LET PI = 3.14159
40 LET AREA = 3.14159*14*14	20 LET R = 14
50 NEXT	30 FOR N = 1 TO 1000
	40 LET AREA = PI*R*R
	50 NEXT

A slightly different application of RULE 5 involves the replacement of a frequently used constant (such as CHR(13)$) by a variable (like $RETURNS$$). After one makes the assignment, $RETURNS$ = CHR(13) , it is then much faster to access the variable $RETURNS$$ than it is to compute CHR(13)$. This application of RULE 5 works even on the IBM PC and PCjr.

At least one additional way to improve our program exists. It involves a rule which is of a slightly different nature than those previously given. It can yield spectacular time savings, but may be difficult or impossible to apply.

FIGURE 4
Execution Time Before and After RULE 5

	Apple II	IBM	C-64	TI-XB
Before	18.4	5.1	28.4	16.7
After	6.5	5.6	6.6	13.5

RULE 6: If the usual "speed-up" methods still do not give the level of performance needed, then try to devise a better algorithm.

In the case of our present programming task, it is possible to devise a much better algorithm. The key observation here is that the numbers which are divisible by 7 are precisely those numbers which are multiples of 7—i.e., the numbers 7, 14, 21 . . . We can find these numbers by addition and avoid testing the numbers that we know are not multiples. Nearly any version of an algorithm based on this new idea will be significantly faster than our best previous effort. The next program, for example, runs at least 8 seconds faster than *Program 1.1*!

```

Program 1.6
10 LET S = 7
15 REM *** START OF LOOP
20 PRINT S;
30 LET S = S + 7
40 IF S < 1000 THEN 20
45 REM *** END OF LOOP

```

In view of the rules given above, it is clear that *Program 1.6* is not the fastest implementation of the new algorithm. It was provided to point out that even a non-optimal version of a superior algorithm is usually much better than any version of an inferior algorithm. The next program is also based on the new algorithm, but it uses a FOR loop with the STEP option, which makes it considerably faster than the GOTO loop above:

```

Program 1.7
10 FOR S = 7 TO 1000 STEP 7
20 PRINT S;
30 NEXT

```

It is surprising that as many as 6 run-time reduction rules can be applied to a program as simple as *Program 1*. Figure 5 summarizes the time improvements achieved by applying these 6 rules.

FIGURE 5
Execution Times After Application of RULES 1-6.

Rule	Apple II	IBM	C-64	TI-XB
Before rules	22.9	21.1	24.2	39.1
1	19.9	15.7	21.0	37.1
2	12.5	11.5	13.4	29.3
3	12.3	11.1	13.1	..*
4	11.1	11.0	11.8	28.5
5	10.9	11.1	11.7	28.5
6	1.7	2.6	2.1	7.7

*Not applicable to TI BASIC or Extended BASIC

FIGURE 6
Execution Times for RULE 7

	Apple II	IBM	C-64	TI-XB
Before	53.1	7.3	53.6	22.4
After	6.6	5.6	6.6	16.4

FIGURE 7
Execution Times for RULE 8

	Apple II	IBM	C-64	TI-XB
Before	6.0	4.9	6.1	14.8
After	5.7	4.5	5.8	14.4

RULE 7: Replace exponentiations, such as A² and A³ with the corresponding multiplications, A*A and A*A*A.

This idea may have occurred to you when you noticed in Figure 3 that exponentiation was the most expensive operation time-wise. The programs below illustrate the potential time savings. Both programs compute the area of a circle with a radius of 14 in a loop with 1000 repetitions. See Figure 6 for the rather marked improvement in execution time when RULE 7 is applied.

<pre> Before RULE 7 10 LET PI = 3.14159 20 LET R = 14 30 FOR N = 1 TO 1000 40 LET AREA = PI*R^2 50 NEXT </pre>	<pre> After RULE 7 10 LET PI = 3.14159 20 LET R = 14 30 FOR N = 1 TO 1000 40 LET AREA = PI*R*R 50 NEXT </pre>
--	---

RULE 8: Remove unnecessary parentheses.

By gaining a clear understanding of operation precedence, it is often possible to remove parentheses which serve no purpose. For example, compare the execution time of the segments below:

<pre> Before RULE 8 10 LET A = 2 20 LET B = 3 30 FOR N = 1 TO 1000 40 LET C = (A*B) - N 50 NEXT </pre>	<pre> After RULE 8 10 LET A = 2 20 LET B = 3 30 FOR N = 1 TO 1000 40 LET C = A*B - N 50 NEXT </pre>
--	---

In this case (see Figure 7), a 5 to 8 percent improvement is achieved, merely by knowing that in the absence of parentheses, multiplications are done before subtractions. There are some machine-specific precedence details, but in general, the order of evaluation is:

- 1) Functions, such as ABS, INT, etc.
- 2) Exponentiation
- 3) Multiplication and division
- 4) Addition and subtraction
- 5) Relational operators, such as < =

Keep these rules in mind whenever you program, and you'll find that many seconds can be shaved from a program's execution time. Stay tuned for a follow-up in the next issue as we will detail more rules about conditionals and subroutines to aid you in *Speeding Up a BASIC Program*. HCM

Thinking of subscribing?
Remember these time-worn truths:



"A watched pot never boils."

"Patience is a virtue."

"Good things come to those who wait."

"Allow 6-8 weeks for delivery of your first issue."

HOME COMPUTER™
magazine

Moon Patrol

A Review by Steve Nelson
HCM Staff

Don't miss this career opportunity with the Lunar City Police Department.

Applications now accepted . . .

I remember when I first saw the Lunar City Police Department's ad for replacement gunners on the moon. I was a three-year veteran of the L.A. Police Department (inventory control) and was bored with my unexciting job. When I joined the ranks, I expected excitement, danger, challenges—anything but being stuck behind a desk all day while a jerk like Lieutenant Weedleman threatened me with disciplinary action just because I violated departmental policy number 7438.8 section 574.2 paragraph 16—of the dress code (I had a button missing on my shirt). I lied about my combat experience and caught a freighter the next day. I wasn't sure what I was getting myself into, but I figured that I had nothing to lose—except maybe my life."

Moon Patrol, by Atarisoft, is an arcade style game that puts you in the driver's seat of a Lunar City patrol car. Your assignment is to patrol Sector 9, a dangerous beat if there ever was one.

Modern Conveniences

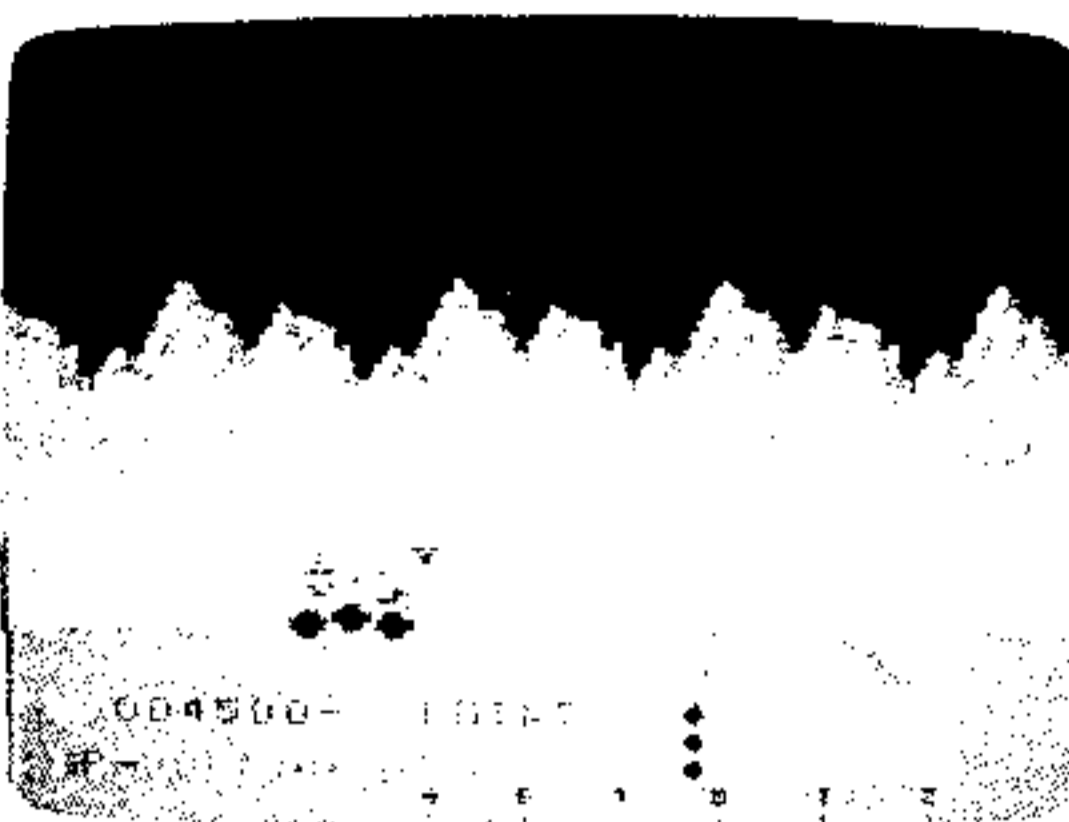
Driving a patrol car on the Moon is no picnic, and driving one through Sector 9 is even worse—it's the home of the meanest, nastiest, most ornery thugs in the known galaxy. Fortunately, your patrol car is equipped with all of the latest modern conveniences: anti-gravity buttons, laser cannons, and six-wheel drive to get you over the rough spots—and it can sure get rough out there. Huge craters, moon rocks, patrol-car-eating space plants, rolling boulders, land mines, tanks, and even flying saucers make Sector 9 a real challenge.

Players may choose from a beginner's level for green recruits, and a championship level for experienced cops. Each course is divided into 26 subsections marked by points A-Z; each of these subsections are grouped together into 5 larger sections. Getting through each region requires all of your driving skills. Both levels are similar in appearance, and are quite difficult as you get near the end of either one. Points are scored each time you jump over a crater, blast or jump over a moon rock, and shoot down enemy saucers and tanks.

Bombs Away

From the moment you begin your patrol, you are attacked by UFO's piloted by space thugs who indiscriminately drop bombs all over the sector. There are tons of them, and only one of you—so evasive action and a quick trigger finger are necessary. You must shoot down the attacking space ships and dodge the bombs.

HCM Review



Name:	Moon Patrol
Program Type:	Arcade game
Machine:	TI-99/4A, C-64, Apple, IBM PC
Distributor:	Atarisoft 1265 Borregus Ave Sunnyvale, CA 94066
Price:	Cartridge (TI & C-64) or Disk (Apple & IBM) Prices vary under \$35
System Requirements:	TI-99/4A & C-64 console only. Apple II, II+, IIe or IIc with disk drive. IBM PC with 128K RAM, color graphics board
Performance:	Poor Fair Good Excellent
Apple, IBM	████████████████████
C-64*	████████████████████
TI-99/4A	████████████████████
Engrossment:	████████████████████
Documentation:	████████████████████

* Review copy unavailable from publisher.

all the while remaining on the lookout for space craters and moon rocks. Using your patrol car's anti-gravity button (on the joystick), you can jump over meteor craters and other obstacles. Once you make it through the first section, you may be awarded bonus points if your time is fast enough. Then you begin the next section. Each time you advance, the pace of the attacks and the level of difficulty increases.

The game's pace is very fast, and you must maintain absolute control over your patrol car or you will crash into a crater or a moon rock. You can control the car using the keyboard or joysticks. I recommend joysticks because using the keyboard is somewhat awkward—there are 5 keys required to control the car.

The graphics are decent, even though my patrol car's wheels appeared to be designed with Leggo blocks. As you are patrolling Sector 9, three "layers" of distant mountains scroll by at different speeds, giving you the illusion of depth, although you can really move in only

"Driving your patrol car through Sector 9 is extremely dangerous—it's the home of the meanest, nastiest, most ornery thugs in the known galaxy."

two dimensions. The attacking space ships are almost too easy to destroy, but there are so many of them that it is still difficult to get through a section without losing several patrol cars. Occasionally, my laser bullet would blast right through an attacking ship with no effect—but I'd just get it on the next pass.

The sound effects are better than average, with a crazy tune that never did get on my nerves as much as a

lot of video-game music does. The documentation adequately provides you with the basics of how the game is played (you don't need to know much).

Moon Patrol has several features that make it more enjoyable than most run-of-the-mill video games. The game keeps track of the fastest time on a given sector (in memory only), as well as your elapsed time in the sector just completed, and the average time it takes to complete the block—giving you an indication of how your performance shapes up. The game shows one of its friendliest features when you lose a patrol car: You don't have to start all over again at the beginning—you simply begin again at the point where you lost your car.

I tested versions of *Moon Patrol* for the TI-99/4A, the Apple II, and the IBM PC and they were essentially the same, except for the differences in graphics and sound effects—the TI version was the clear winner here.

Moon Patrol stacks up as a first rate imitation of the real arcade game. It provides you with a difficult mission, good graphics and sound, and special features that tend to draw you right into the game. Whether you are just looking for a new game with which to pass the time, or, if you are a maniacal video gamer, *Moon Patrol* promises to make life a little more exciting for you.

HCM

HOME COMPUTER™

product news

Each month we publish items of interest and news of recently or soon-to-be released computer products. Our publication of information from manufacturers of computers, peripherals, software, and accessories is not to be construed as product endorsement. Prices quoted are the manufacturers' suggested retail prices and are subject to change.

Send press releases to:

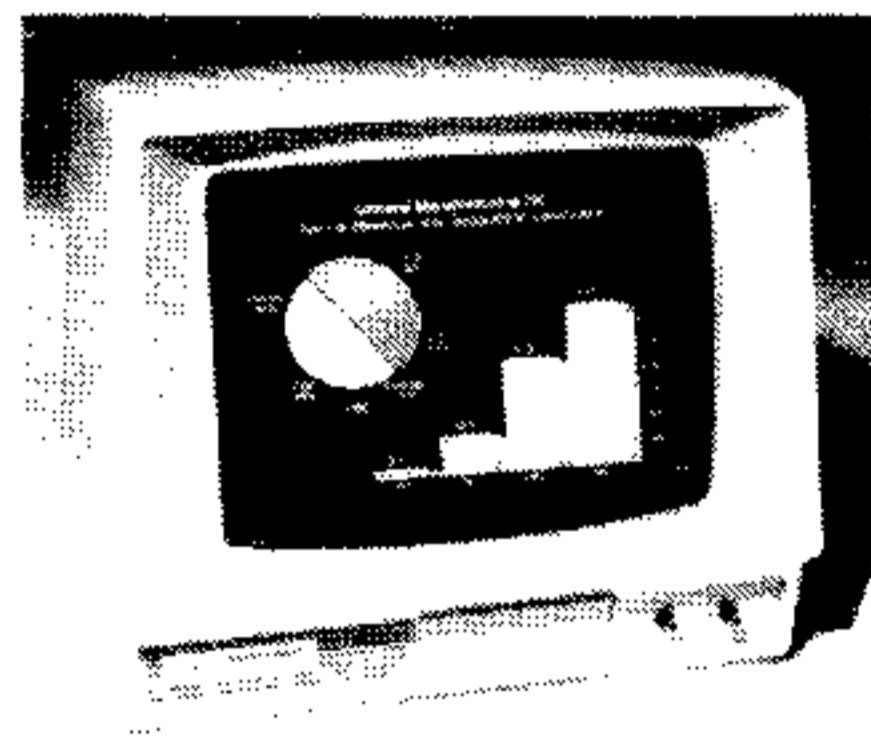
Product News Editor
Home Computer Magazine
1500 Valley River Drive., Suite 250
Eugene, OR 97401



Persistence Pays Off

Phosphor Reduces Screen Flicker

Tecmar, Inc. has introduced a "long persistence," high resolution color monitor for use with IBM PC and PCjr systems. The 640 x 480 resolution and long persistence phosphor is designed to reduce screen flicker. Tecmar's Color Monitor features 16 colors, and costs \$795.



Tecmar, Inc.
6225 Cochran Rd.
Solon, OH 44139-3377
(216) 349-0600



Graphx From the Land Down Under

High-Powered Graphics for the 99/4A

Graphx of Sidney, Australia has announced a new graphic design program which bears the company name. Graphx provides MacPaint-like features, such as free-hand drawing, a Zoom option, an easy-to-use color palette, shape-filling, typewriter mode, and flexible cut-and-paste. Straight lines, circles, and ellipses are also provided for. Drawings—in whole or in part—can be stored on a clipboard, on disk, or even used in assembly language programs as background,

etc. Whole collections of often-used shapes, special alphabets and the like can be stored in this fashion and used later. The Clipboard can even be used to create animation. An additional Printer Utility System allows letterhead printing, and giant-sized Poster Prints. Versions are available for Mini-Memory, Editor Assembler, or Extended Basic modules. 32K Expansion and disk drive required. The developer's price is \$50.

Graphx
P.O. Box C568
Clarence Street
Sydney, N.S.W., Australia 2000



Brainfood for Apples

Improve Your Memory by Computer

Brainworks has changed its name to Brainpower and released three new software products. Thinkfast, for both the Macintosh and the Apple II Family of computers, is a program designed to strengthen one's memory for numeric and verbal information (left brain), and abstract

graphic/visual material (right brain). TeleChess, for the Apple II Family, offers real-time animated graphics permitting chess play by telephone. Torpedo Run, for the IBM PC, is a simulation game presenting eight historical World War II submarine missions.

Brainpower Inc.
24009 Ventura Blvd.
Calabasas, CA. 91302



A Light Pen For Kids

Simplifying the Educational Process

MicroEd introduces a light-pen reading series designed to provide learning-disabled and young students with beginning word skills normally taught in kindergarten through third grade. Initially available for the Commodore 64, the programs use 800 full-

screen drawings, and can be purchased in four packages consisting of 14 disks. Packages One and Two have 3 disks and retail for \$74.95 each. Packages Three and Four have 4 disks, and retail for \$99.95 each.

MicroEd Incorporated
P.O. Box 444005
Eden Prairie, MN 55344
1-800-MicroEd

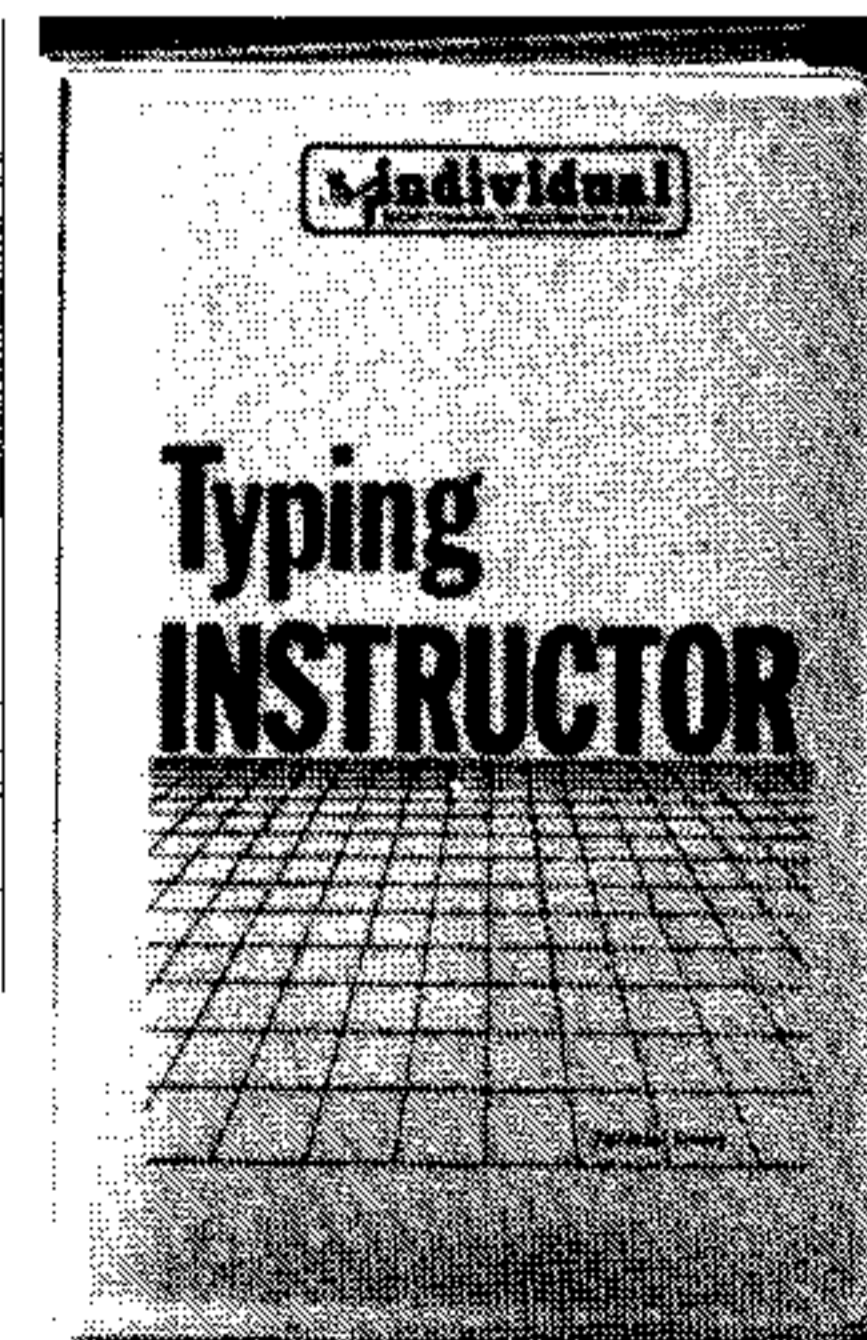


Three For The IBM PC

New Tutorials on Computer Concepts

Individual Software Inc. has introduced three new tutorials for the IBM PC and PCjr. They are: a new version of Professor DOS, a program designed to teach the use of DOS 1 and 2; Typing Instructor, which offers interactive training, drill and practice, a fast action game, and word processing concepts; and Professor Pixel, which teaches you graphics, sound, and animation.

Individual Software Inc.
1163-I Chess Dr.
Foster City, CA 94404
(415) 341-6116



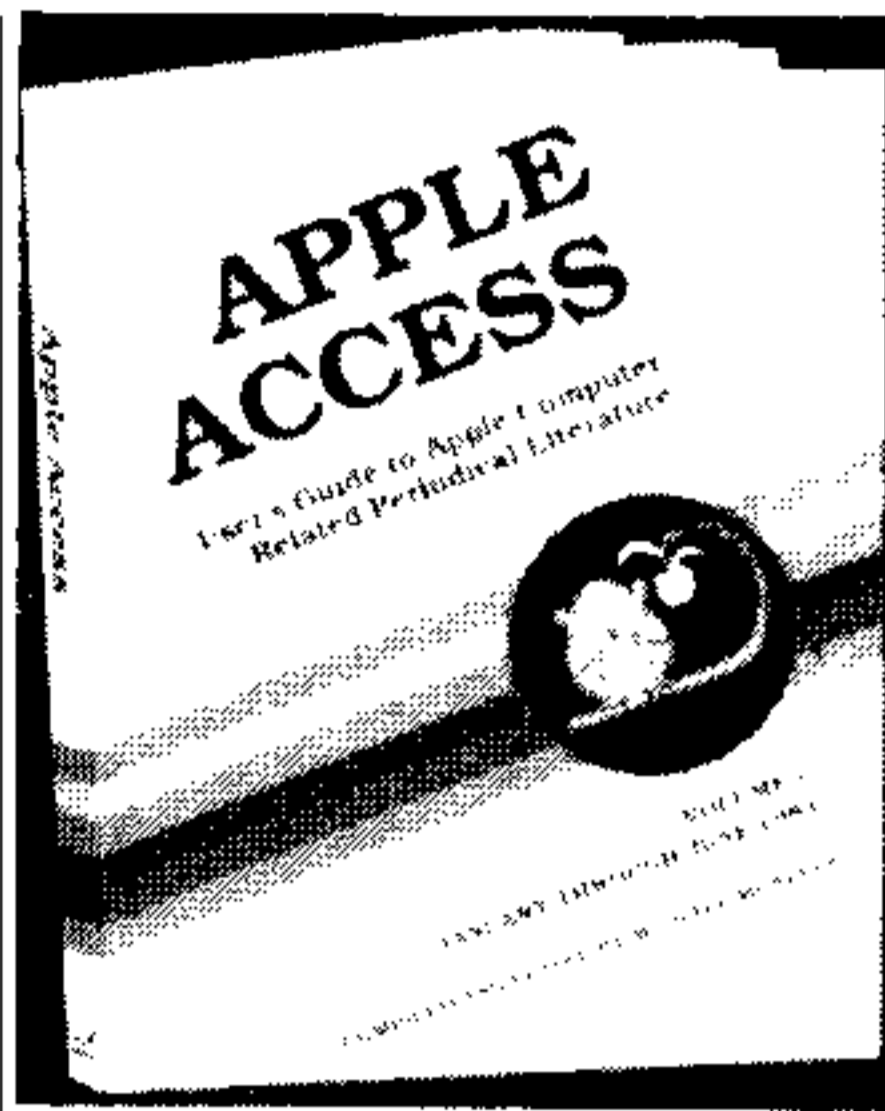
HOME COMPUTER

product news

Everything About Apples

A User's Guide to Apple Periodicals

Stony Point Publications has announced the release of a new book, *Apple Access*—a user's guide to Apple-computer-related periodical literature. *Apple Access* is the first volume in a series of semi-annual guides directed at documenting most of what is written, and what will be written about Apple computers. Each book will contain more than 9000 entries. In addition, there will be two catch-up volumes covering the period from 1977 through 1982, and another for 1983. The books will retail for \$19.95.



Stony Point Publications
Box 4467
Petaluma, CA. 94953



A New Speaker

Speech Synthesizer with a Loud Voice

Votrax, Inc. has upgraded their line of Type-'N-Talk speech-to-text synthesizers by adding a built-in speaker system. Type-'N-Talk will still retain an external speaker jack. The Type-

'N-Talk synthesizer will operate on any personal computer that has an RS-232 serial port. The complete system retails for \$265.

Votrax, Inc.
1394 Rankin
Troy, MI 48083
(313) 588-2050



A Game Just Like the Movies

And A Mystery That Teaches Geography

Two new games have been released by Broderbund Software. *Where In The World Is Carmen Sandiego?* is an adventure mystery game that teaches users about world geography with the help of a 1985 World Almanac and Book of Facts that accompanies the program. The mystery scenario varies in every game because there are 10 different villains and 30 different countries. It requires an Apple II Family computer with at least

64K, and it retails for \$39.95. *Karateka* was designed to emulate an action-packed movie. It has a story line, characters, animation, and sound, some of which was recorded and transferred to disk. In the game, a young karate master must use his karate skills to fight an evil warlord to rescue his bride-to-be. *Karateka* is available for Apple II Family (\$34.95) and Commodore 64 (\$29.95) computers.

Broderbund Software
17 Paul Dr.
San Rafael, CA 94903-2101
(415) 479-1170

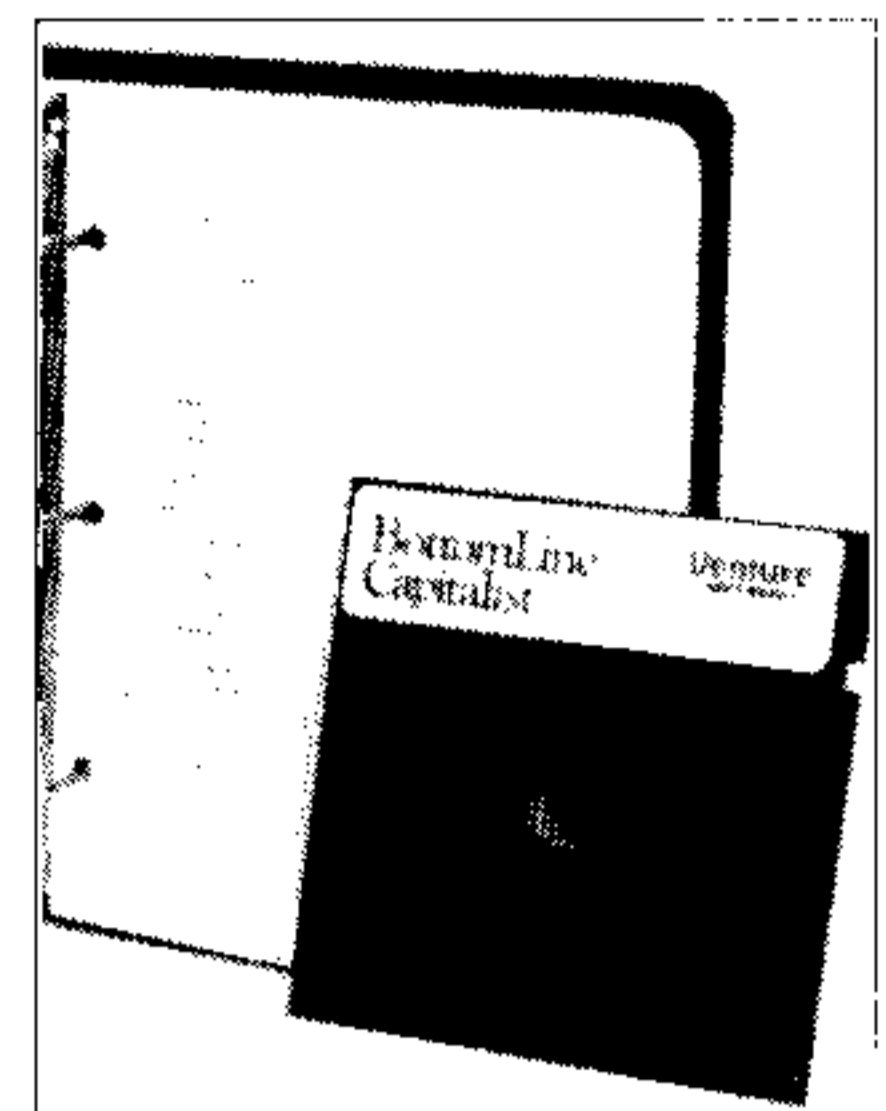


Capitalism: the Bottomline

Market Strategy Taught By Game

Bottomline Capitalist, a business-simulation program for IBM PC and Apple computers, challenges players or teams to compete as entrepreneurs in the marketplace. Through 40 quarters of business activity, players make decisions on budgeting, raising capital, marketing, developing managers and staff, raising capital, and more. The goal is to foster growth of the business enterprise, increase profitability, and maintain or gain market-share. *Bottomline Capitalist* retails for \$129.95.

Venture Software
16200 Ventura Blvd.
Encino, CA 91436
(818) 986-4110



A Symphonix of Words

A Real Speech Synthesizer for Junior

SynPhonix jr is a speech synthesizer board by Artic Technologies that plugs into the internal modem slot of the IBM PCjr. It includes an on-board speaker and an external speaker jack. In addition, the *Sonix jr* Speech Operating System and the *Text-to-Speech jr* translation program come with

SynPhonix jr. This support software allows users to generate vocabulary from phonemes or strings of English text. Unlimited speech, music, and sound effects may then be incorporated into user-written programs with BASIC statements. The *SynPhonix jr* retails for \$175.

Artic Technologies
2234 Star Court
Auburn Heights, MI 48057
(313) 852-8344



Teenagers in Space

14 Year-Old Writes TI Game in Forth

Spacestation Pheta is a complex, graphics-filled game available for the TI-99/4A from T & T Software, a small company headed by 14 year-old Tad Woods. Written in Wycove Forth, this new game features 80 different playing screens. A built-in editor allows the user to design and add an unlimited number of original screens. Players explore a mysterious abandoned space station with a limited supply

of oxygen, encountering strange objects that can be used to survive. High scores can be saved on disk, and the editor also allows flexible playing speeds and starting points. A disk controller and drive, 32K Memory Expansion, and one of 4 modules—Extended Basic, Mini Memory, Editor Assembler, or TI-Writer—are required. *Spacestation Pheta* is available directly from the company. Price is \$14.95.

T & T Software
109 Tee Circle
Salem, VA 24153



Pull-Down Menu Magic

Four New Kids On the TI Block

Asgard Software announces the release of four new programs for the TI-99/4A computer. The products are: an easy-to-use schedule maker with pull-down menus and 20 different menu options for a variety of tasks in the home or business; a stamp-manager program for

philatelists, also featuring pull-down menus and a printer option; a music synthesizer designed with the musical novice in mind, that allows users to create and merge compositions; and an arcade-style game, Balloon War. All four programs require Extended BASIC.

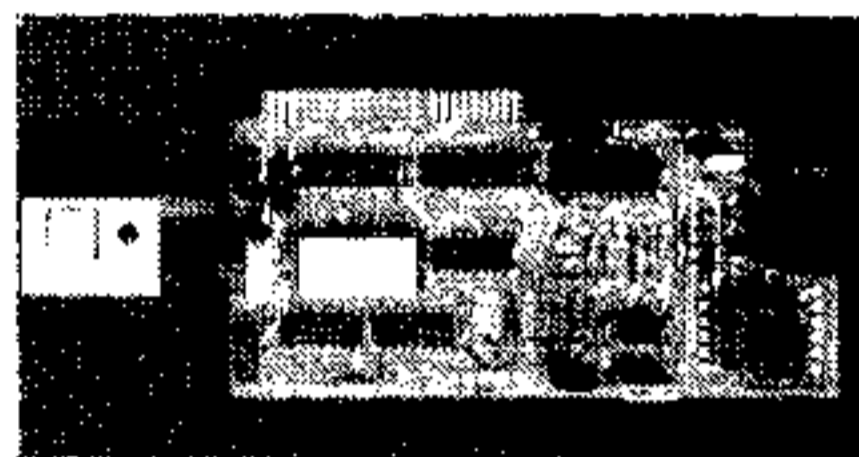
Asgard Software
P.O. Box 10306
Rockville, MD 20850



More Modems Make Market

Data Exchange Systems For Apple II

Telstar Communications has released its Networker Modem along with its Zoom/Modem IIe for Apple II, II+, and IIe systems. The Networker Starter Set (\$99) includes a plug-in 300-baud modem board and software to store conversations to disk in DOS or ProDOS format. Networker with advanced software (\$149) adds an advanced text editor, and the ability to transfer Apple text or ASCII files to other computers over the phone. The



Zoom/Modem IIe (\$149) has all the features of the Networker, plus auto-dial, auto-answer, monitor speaker, and Hayes Micro-modem IIe compatibility. The Zoom/Modem IIe is also available with advanced software for \$199.

Telstar Communications
145 Lincoln St.
Boston, MA 02111
1-800-344-3311



Invest to Learn. Learn to Invest

College Subjects, Computer-Taught

Useful Software has packaged a number of programs of interest to college students and investors on-to disks for the Commodore 64. The College Pak contains more than 20 programs for computer-aided instruction in college-level math, physics, chemistry, engineering, language,

history, and medicine. The Investors Pak contains more than 20 business and investment programs in Real Estate, Mortgages, Bonds, Loans, Syndications, Leases, Shelters, and other investment analysis subjects. The College Pak and The Investors Pak are \$25 each.

Useful Software Co.
P.O. Box 54-H
Scarsdale NY 10583
(914) 633-8427



Keeping Track and Keeping Fit

Two Products Aid Weight Loss

Journal and Belly Buddy for Weight Loss are the two latest releases from New Found Software for Apple II Family machines. Journal allows users to daily record thoughts and ideas as in a diary, or as an organizer. It includes a perpetual calendar and an appointment calendar. Journal is password but not copy-

protected for personal use. It costs \$15.95. Belly Buddy for Weight Loss computes calories, carbohydrate grams, protein, and 13 other nutrients for 425 food items. Users have the option to add more, and can track their food intake, set weight goals, and keep a 5-year weigh-in history. It costs \$28.95.

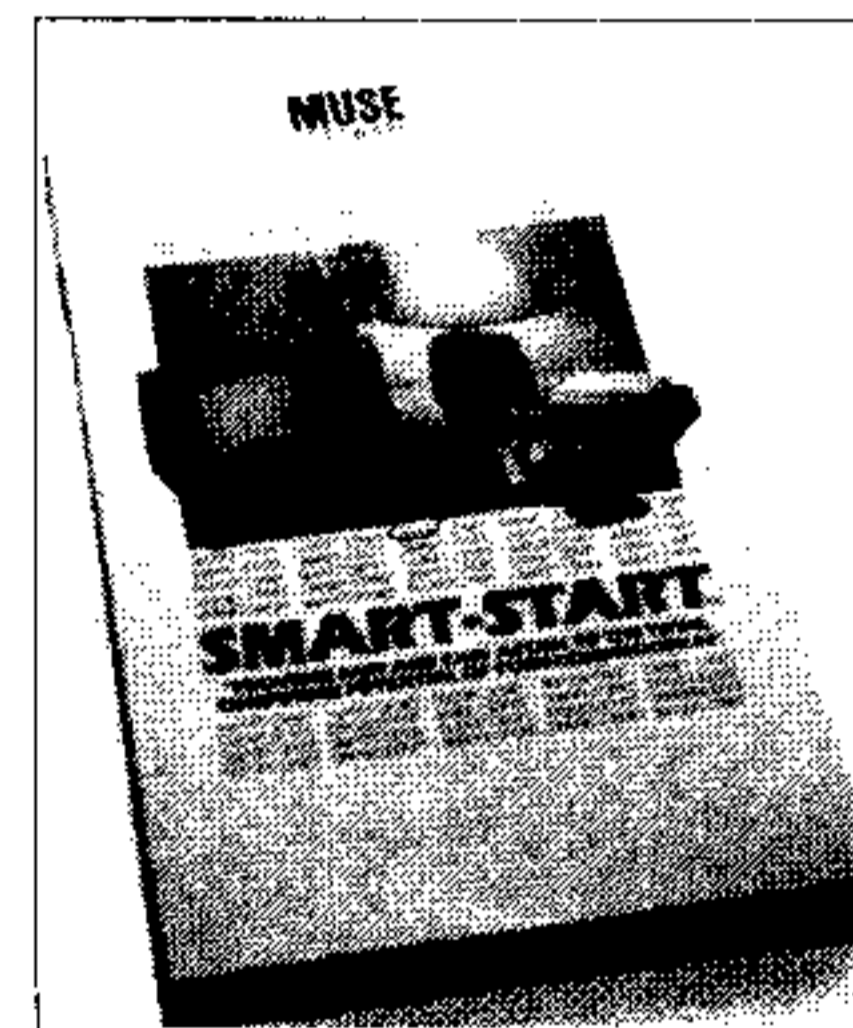
New Found Software
9040 Aileen Dr.
Mentor, OH 44060



Starting Smart, Staying Smart

Utility Speeds Up C-64

Muse Software has announced the release of Smart Start, a program for the Commodore 64 that generates BASIC code automatically. It is designed for use with a disk drive to make writing, saving, loading, and running programs as simple as moving the cursor and pushing a button. Smart Start can be used immediately by computer neophytes to create and save pictures, music and sound effects programs, and more. It retails for \$39.95.



Muse Software
347 N. Charles St.
Baltimore, MD 21201
(301) 659-7212



Expense Accounting

Program Tracks Outflow of Money >

Sapana Micro Software has announced the release of its Expense-Track II, a personal and small-business accounting program to store and retrieve expense information, and print checks. It runs on the IBM PC and PCjr. The program accommodates 255 main

expense categories and 255 subcategories, and can sort by month, day, year, and category. A calculator mode is included for numerical values, and summary reports can be produced in bar graph and table form. Expense-Track II is priced at \$69.95.

Sapana Micro Software
1305 S. Rouse
Pittsburg, KS 66762
(316) 231-5023



More Help For River City

Musicman Composes on IBM, Apple

A new program from Zepher Services allows the user to compose and play music on the IBM PC/XT/PCjr or Apple II Family computers. With Musicman, you can compose music on screen using standard musical methods. Compositions can be saved on disk for later replay or revision. Sample music is also provided with the program. A musical staff of 5 lines appears on the screen

where you can place whole, half, quarter, eighth, or sixteenth notes, or rests. Then, you can set a desired key and any accidental sharps or flats. Notes can be dotted for extended duration. Once the composition is done, you can change the tempo or pitch to test the variations. Legato can also be varied. Musicman is available for \$29.95 (plus \$2.00 shipping) directly from the publisher.

Zepher Services
306 South Homewood Ave.
Pittsburgh, PA 15208
(412) 247-5915



C-64 BASIC Made Easy

Step-by-Step Guide to Programming

Howard W. Sams & Co. is releasing a new addition to their popular 8-to-80 series entitled Commodore 64 for Kids 8 to 80. The book is designed to work hands-on to create short BASIC programs in Commodore BASIC. Commodore 64 for Kids 8 to 80 is fully illustrated, and includes exercises and review material useful to both beginning and experienced programmers. The content of the book is based on concepts used at National Computer Camps by the authors.



Howard W. Sams & Co.
4300 W. 62nd St.
Indianapolis, IN 46268
(317) 298-5400



Financial Planning the TI Way

Key-in Business Application Programs

The Chilton Company has released the book Financial Analysis on TI Computers. The book contains 18 chapters on various topics of financial interest. Each chapter contains a brief explanation of the analysis technique, example applica-

tions, and program listings. All programs are written in TI BASIC. Specific topics include financial ratio analysis, cost-volume-profit determination, inventory-level analysis, and many more. The book is also available for the IBM PC.

Chilton Book Company
Radnor, PA
1-800-345-1214



When it Rains—Have Fun

10 Programs Especially for Kids

Thorn EMI Computer Software Inc. has announced the first in its new Computer Clubhouse Series of software products available for the Apple II Family of computers and the Commodore 64. Rainy Day Fun programs are designed for children aged 5 through 9, and are educational in nature. Ten programs on the disk use icons to lead children through the menus. Rainy Day Fun retails for \$39.95.

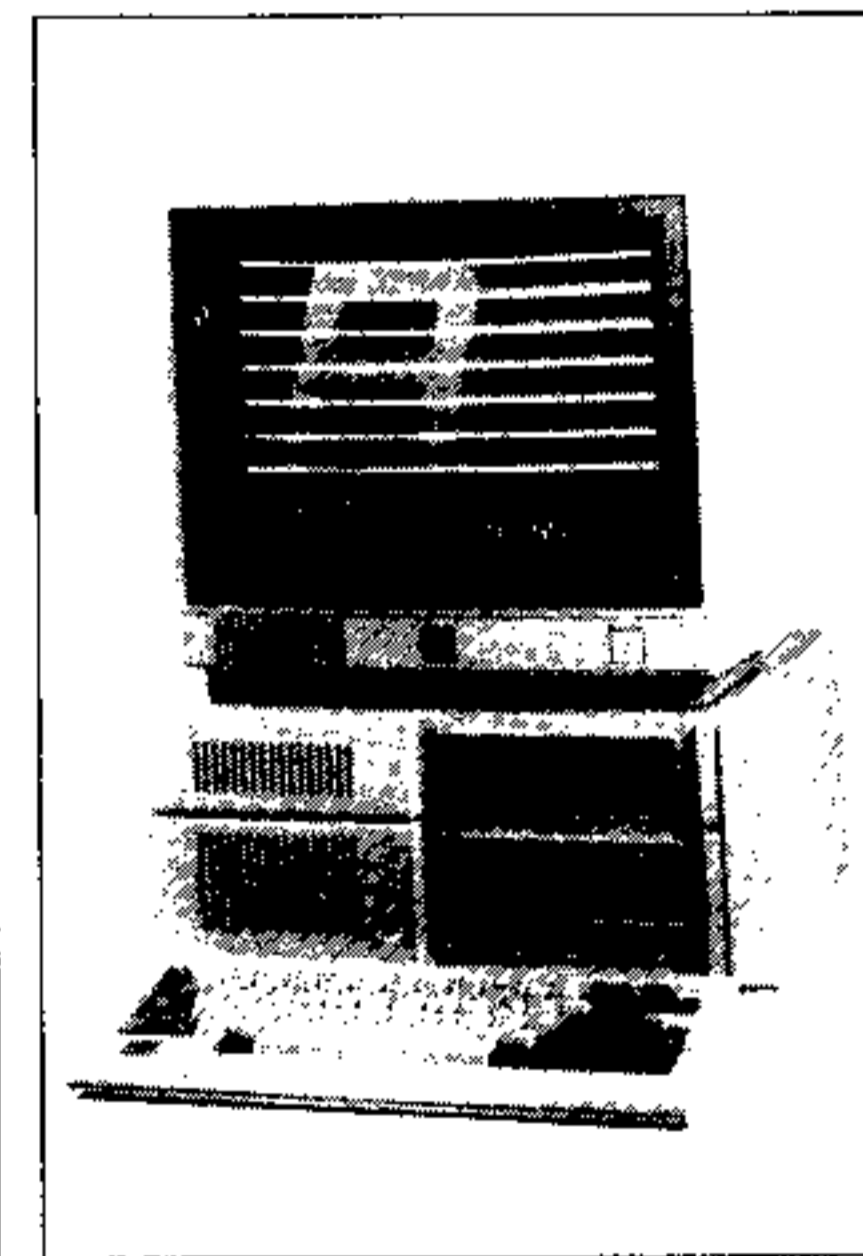
Thorn EMI
Computer Software Inc.
3187C Airway Ave.
Costa Mesa, CA 92626



Little Brother's Growing Up

PCjr Gets More Memory

Racore Corp. has introduced two new products for the IBM PCjr that give it many capabilities of the IBM PC and PC AT and XT. The new products are a 1.2MB floppy drive called the Racore ATcessory, and a network product, Racore-Net, enabling the PCjr to communicate with the IBM PC AT. The floppy disk drive includes a parallel printer port, a clock calendar, a direct memory access drive controller, and a PC/PCjr mode switch. ATcessory makes the PCjr even more PC-compatible, allowing it to run Framework, Wordstar 2000, and other software previously not usable on the PCjr. Racore-Net allows up to 16 PCjrs, PCs, XTs, and



ATs to be linked together. ATcessory sells for \$895, and Racore-Net retails for \$200.

Racore Corp.
10 Victor Square
Scotts Valley, CA 95066
(408) 438-7255



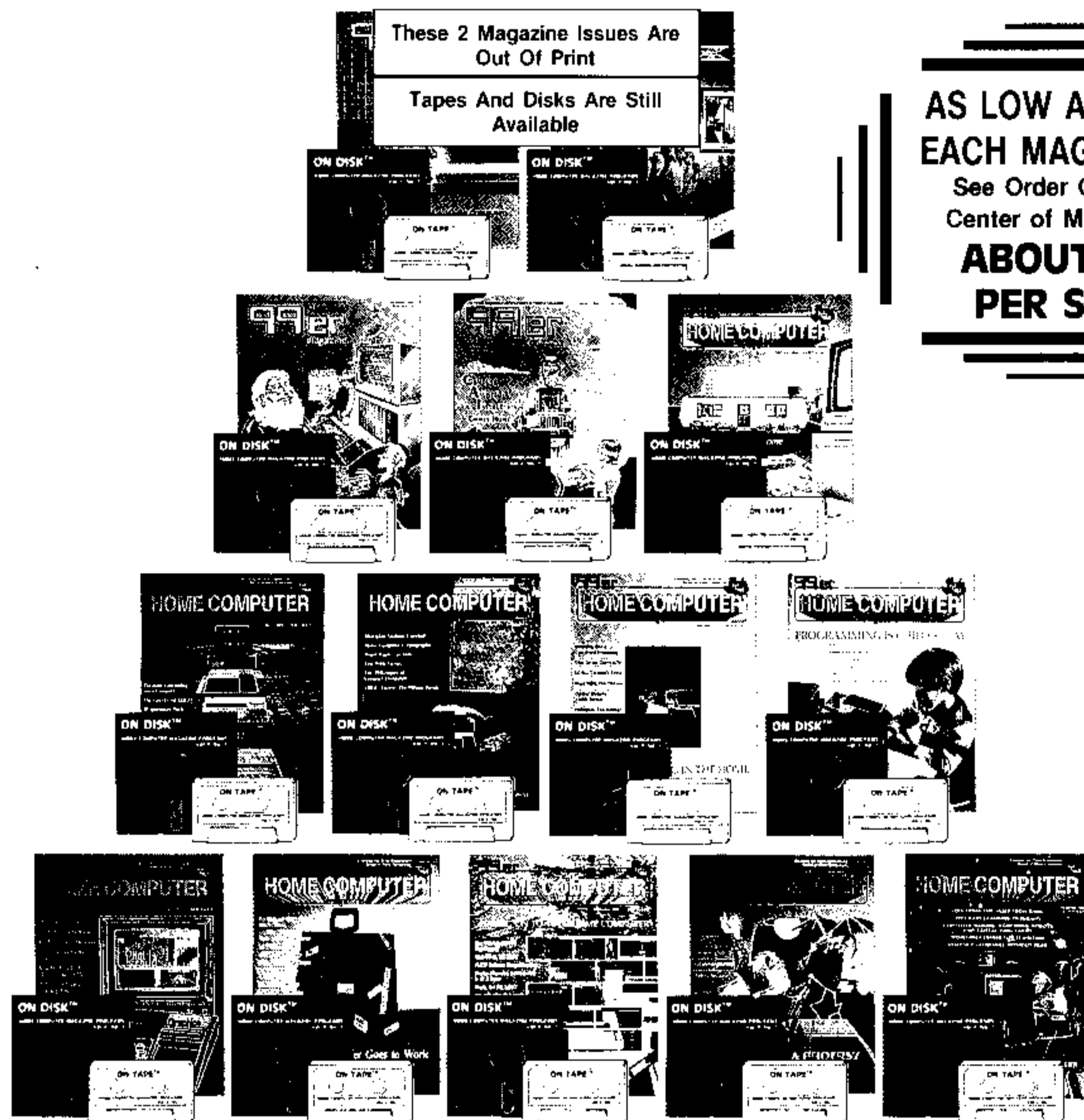
[Special Note: We are hungry for information on TI-related products for inclusion in HCM Product News! Product developers for the TI-99/4A, please send us your press releases.]

LIMITED SUPPLY

A Back-Issue/Software Bonanza Of **99'er** HOME COMPUTER magazine At Unbeatable Prices

The original 99'er Magazine and 99'er Home Computer Magazine were the forerunners of the present-day Home Computer Magazine. Each of these magazine back issues—exclusively covering the Texas Instruments TI-99/4A—is now available with your choice of either a floppy disk or a cassette tape that contains all the programs in that issue.

12 MAGAZINE & MEDIA SETS



These 2 Magazine Issues Are Out Of Print

Tapes And Disks Are Still Available

AS LOW AS \$1.83 EACH MAGAZINE!
See Order Card at Center of Magazine
ABOUT \$4 PER SET!

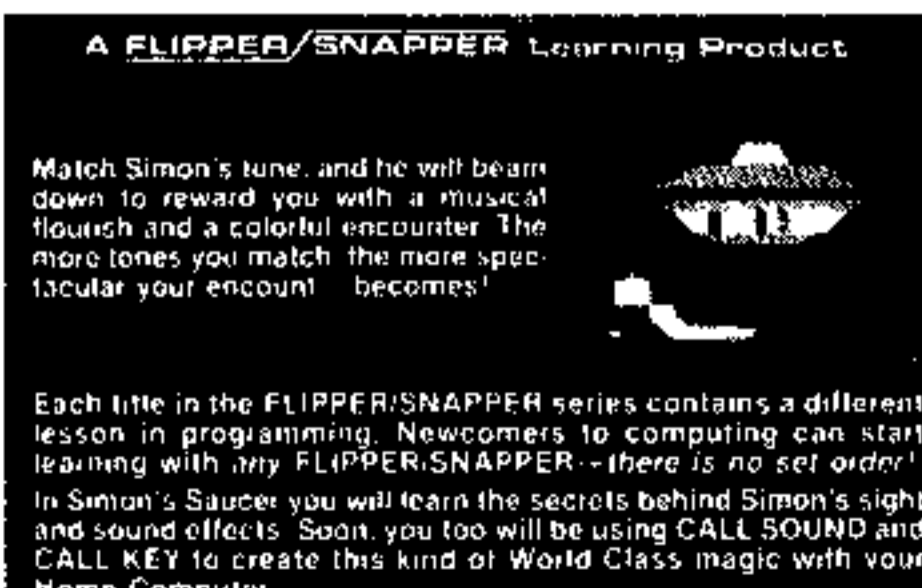
MAGAZINES OR MEDIA MAY BE ORDERED SEPARATELY
To Order, Use Bind-in Card At Center Of Magazine.

*** SPECIAL MIX OR MATCH BONUS ***
—WHILE SUPPLIES LAST—

You will receive a **FREE** Simon's Saucer™ package and a **FREE** TI-FEST™ commemorative poster when your order includes at least **12 items** from this page in any combination of magazines or media.



- A quality, ready-to-run game on cassette tape.
- A durable and attractive ring-binder collector's case for your software library.
- A complete, easy-to-use programming lesson on a deck of colorful flip cards.



SAVE UP TO \$44 AND RECEIVE FREE GIFTS WORTH OVER \$15!

Hurry—Supplies Are Limited. Order Yours Today!

Offer & Prices Subject To Change Without Notice.

**Issues No. 1-5
OUT OF PRINT**

Contents available in book form as **"Best of 99'er"—Vol. 1.**
See Home Computer Digest or Inquire

ISSUE #6 (Partial Contents)
• How To Produce Sound Effects • Debugging a Game Program • How to Start a User's Group • Verbose: A speech Vocabulary Expansion Aid • Color Mapping • Dynamic Manipulation of Screen Character Graphics • The Beginner's Guide to Cassette Operation With the Home Computer • Pro School Block Letters and Data Compression • Picking the Pines in TI BASIC • Battle Star Space Game • 3-D Animation on the Home Computer • Programming Tips • Who is LOGO? • Tower of Hanoi in LOGO • A Review of the TI Lesson Development Software • An Interview with a Game Designer • Learning Assembly Language with a Magic Crayon • and much, much more.

NOVEMBER 1982 (Partial Contents)
• Chatting with Your Micro Languages for the Home Computer • A Review of the TI Chess Game • Daisy Wheel Printer • The Micro Jam Arcade Game • A Knight's Tour in TI BASIC • LOGO Has Style • ASPIC: A Language for Children • A p-System Beginners Guide • An Interview with the TI Lesson Development Software • An Interview with a Game Designer • Learning Assembly Language with a Magic Crayon • and much, much more.

DECEMBER 1982 (Partial Contents)
• Trx Scribble: A Text Editor for the Home Computer • A Christmas Computer Carol • Managing a Making List the Future Way • Parsing: The Arcade Game • Picking With the Home Computer • Pixel by Pixel • Preventing the Situation • Oh No! Memory Full • A Colorful Tour of TI Fest: The Home Computer Show • Santa's Workshop: The Making of a Home Computer • The Turtle Avails: Movies & Video Games in LOGO • Controlling a BASIC Terminal • The 99'er: Gold Rush • An Arcade Adventure in the Home • Mini Digest of News & Happenings in the TI World • Plus: Games, Reviews, and much, much more.

JANUARY 1983 (Partial Contents)
• Computer Assisted Instruction for the Home Computer • System Basics • Delving into LOGO • The Dow & Battle Flight Simulator • Note Wiz and Poch Master Musical Game Reviews • Learning With the PLATO Computer Library • Strategies for Adventure Gaming • Death Drones • Using the Line By Line Assembler • Close Encounters of the Simon Kind • Electrical Engineering Education Program • Interview With an Arcade Game Designer • TI Invaders • Programming With Pascal • Cyber Duet • News and Happenings in the Home Computer World • Arcade Game Reviews • The Thief Adventure Game • Programming Tips • and much, much more.

FEBRUARY 1983 (Partial Contents)
• Texas Instruments at the Winter Consumer Electronics Show • Home Computer Printers on Review • How to Create Math Dances in LOGO • Vectors in LOGO • ASPIC: A Language for Teachers • The Joys of Advertising: Part 2 • Programming Printers with Chuck Luck: Part 4 • Interview With the Voice of Harsec • Why You Need a Printer for Your Home Computer • I Believe in Titan Space Game • Night Blockade Battleship Game • Tower of Hanoi Pool Program • Computer Gaming Software Reviews • News of Late Developments in the World of Home Computers • and much, much more.

MARCH 1983 (Partial Contents)
• An Introduction to the TI-99/2 Basic Computer • The Hex Bus and the 41A Connector • Making Your Own Say and Spell Game • Disabled Children Learn and Grow • Super Cataloger • A Review of a Disk Library Utility Program • TI's New CC 40 Compact Computer • Robots and Their Social Impact • Twenty Questions With Robot Redford • The Gravity of LOGO • Joystick Jockey—An Overview of Remote Control • Parso: Strategy • Converting Extended BASIC to Assembly Language • Matrix Muncher • Mini Memory Disassembler Utility • Pulling the Strings in Sprites • Letters on LOGO • Tiny Tutorials • Games, programs, reviews, and much, much more.

APRIL 1983 (Partial Contents)
• Computer Assisted Savings Planning to Burn Your Next Egg • Tex-Cipher: Codes and Decodes • Secret Messages • Crosswords—Computer Vocabulary Crossword Puzzle • Cutting Corners On Your Final Budget Using Coupons • Introducing Financial Planning with Multician • The Design Philosophy of the Compact Computer • LOGO Takes On the Pipar Fifteen Puzzle • Super Language • Finding my Sprites • Mini Memory • Confusing Words • Psychological Readiness for Physicists • Gamewar Buffet: A Moving Box Alley Game • Giant and Dwarf Fight! • Game Reviews • Programming Tips • Money Saving Hints • and much, much more.

MAY 1983 (Partial Contents)
• A Consumer's Guide to Word Processing • Word Processing Market Basket • A Generalized Filing Program for VIPS • The Multician Medium Balances Your Checkbook and Budget • Activity Accountant Helps School Secretaries with Extracurricular Activities • Maximizing Your Mini Memory's 4K of RAM • Exploring Enhanced BASIC on the Compact Computer • The LOGO Tortoise Debates the TI-99/4A • Footnote Program to Organize Data with Linked Lists • Mentally Handicapped Learners Team Up with the TI-99/4A • The Wonders of Diskette Storage • Beeline—a Multi-Screen Strategy Game • Lost Run—a 3-D Adventure Game • 3-D Illusions with Sprites in Depth • Game reviews • Group Grapevine, and much, much more.

JUNE 1983 (Partial Contents)
• Children and Computers Make the 99'er Connection • Tune Your Guitar with Our TI Tuning Fork • Talk to Your Computer—Voice Technology • Review of the Game Buffet's Eat or Be Eaten Arcade Game • Protect Your Station in the Space Zapper Game • What Multician Can and Can't Do • Understanding the Output in Drive For Diskettes—Part 2 • Calculate Loan Schedules on the CC 40 • Go on a LOGO Vacation • Letters on LOGO • A Review of Upper Room Software's Programs for Special Learners • Conquering the R202 Joystick Interface • Group Grapevine • Shopping Bus • A Natural Language Interface for the Professionals • Game Reviews • and much, much more.

JULY 1983 (Partial Contents)
• The Evolution of Home Computer Graphics Comes Alive in Graphics Grows Up • Five Data Organizers in Never Out of Sorts • TI & 99'er at the Consumer Electronics Show • WarGames: The Movie and the Book • Editing with Multician • The LOGO Logician Presents To Model is to Learn • LOGO Music: Designing the Screen • Your Speech Synthesizer as a Spelling and Foreign Language Teacher • Software for Your Lowest Printer Port • Gamewar Buffet's Treasure Island and the Confusing Switch • A Review of the TI-99/4A • Learn BASIC for CC-40 Users • 3-D Animation with the TMS 9918A Video Chip • Game Reviews • Group Grapevine • and much, much more.

AUGUST, 1983 (Partial Contents)
• The Home Computer Goes To Work • Bit One: Part Two at the Fashion Factory • Better Business Bar Graphs in Graphic Persuasion • An Ensemble of Assemblers • Cashflow Helps Money Management • Keystrokes for Thirty Fools • Review of Tower of Hanoi Game Reviews of Cavern Quest and Starprobe '83 • Counting Fun for Preschoolers • Peripheral View '83 • Map, Mapping, and Small Puzzles • Multician Program Groups: Cellists, Encoders, and the TI-99/4A • LOGO Word Processor • Group Grapevine • Hello, Hello, Hello • CC 40 Speaks to 99'er • Plain's Progress: Unofficial Student Assessment • Pocket Sunrise • Puzzle Investigator • Gamewar Buffet's Justice Jim, Success Formula • and much, much more!

SEPTEMBER, 1983 (Partial Contents)
• Adding on to Your Home Computer • 99'er Directory of Commercially Available Software, Accessories and Peripherals • Peripheral View '83 Hardware Reviews • Pocket Sunrise, Part Two in Extended BASIC • Byte Lightning—Mini Memory Plays a Game • TI-WRITER Tutorial • LOGO's Functions, Sets and Turtles • The CC-40 and 99'er: Take the Data and RUN • PLATO's Progress Looks at Geometry Courseware and the Shape of Things to Come • Gamewar Buffet's Challenge of Camelot and The Fly • Game Reviews of Nit Wit and Crime and Punishment • Group Grapevine '83 • Hall of Fame • 99'er Digest Update on New Products • and much, much more!

OCTOBER, 1983 (Partial Contents)
• Adventures and Fantasy with Your Home Computer • Do-It Yourself: Adventure • Proprietary Emulator • A Bit Microprocessor • Have No Fear: Assembly Language Won't Bite, Part 1 • Make Your Mark • Les Incore and Debug Computing Carbons • Once Upon a Twinkle: Shall... A LOGO Adventure • Turtle Situ • Number Nibbles for Children • Lots of Hints on Your CC 40 and HX1000 Two-Color Printed/Plotted • Multiplication Medium • A Gaily Adventure Bear Hunt • Escaper From Awards Keep in Extended BASIC • Game Reviews • Shopping Bus • and much, much more.

NOVEMBER, 1983 (Partial Contents)
• Education with your Home Computer • Five Creative Learning Activities for Children • Let's Build America • Have No Fear: Assembly Language Won't Bite, Part 2 • Squaring the Mini Out of TI BASIC • TI-WRITER Tutorial • LOGO I mean it • Interview with Dale Osborn • TI-WRITER At Home in the Office • ILAI's Progress • Interview and Debate • The Multician Medium • Computer Assisted Instruction—99'er Interview: The Kids • Gamewar Buffet: Jack Man and Robo Chase • Game Reviews of Jail Break and Arithmetic • Hall of Fame • 99'er Digest • and much, much more.

These 12 Magazine Sets are Still Available