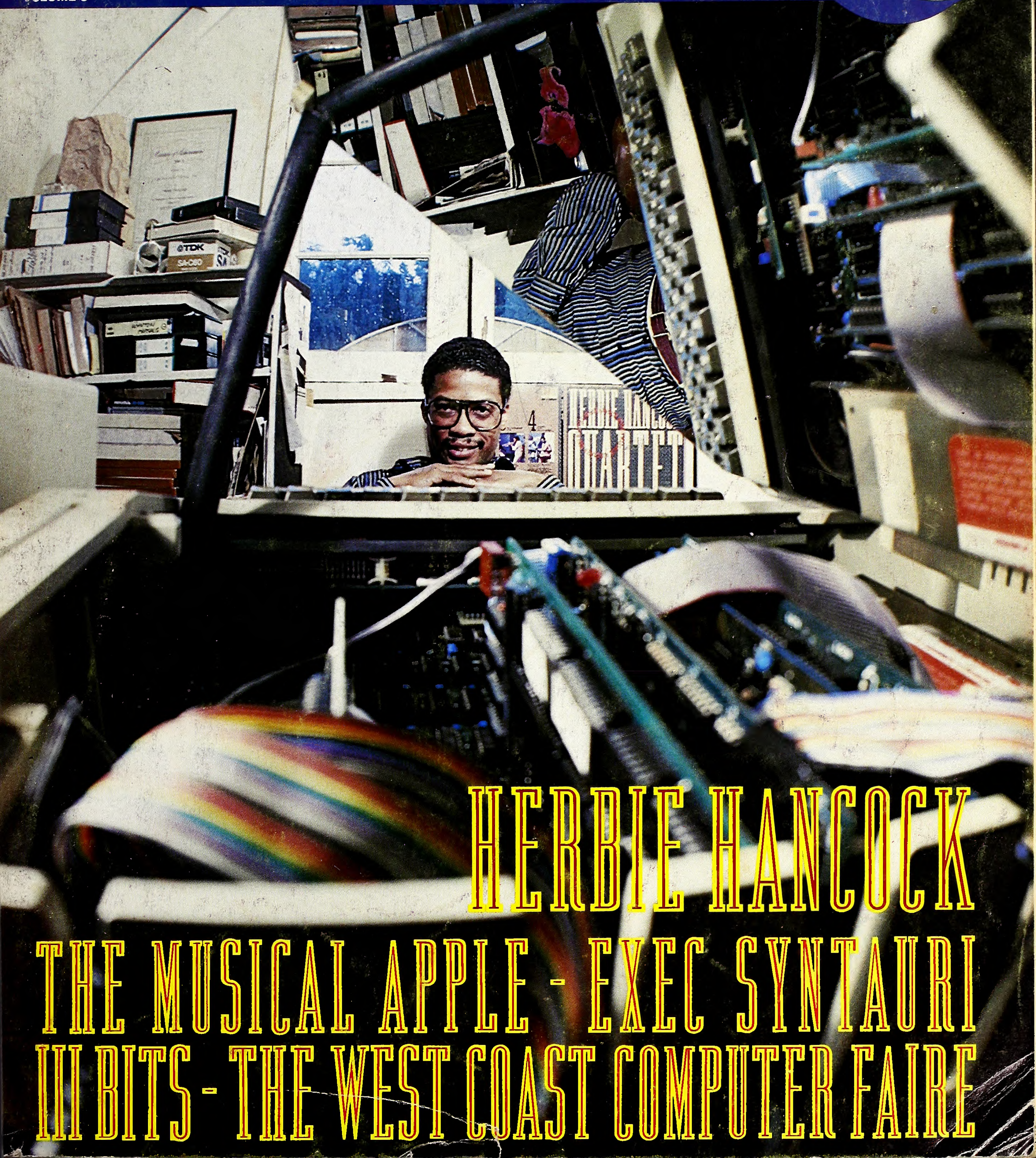
SOFTALK

VOLUME 3 MAY 1983 $3.00

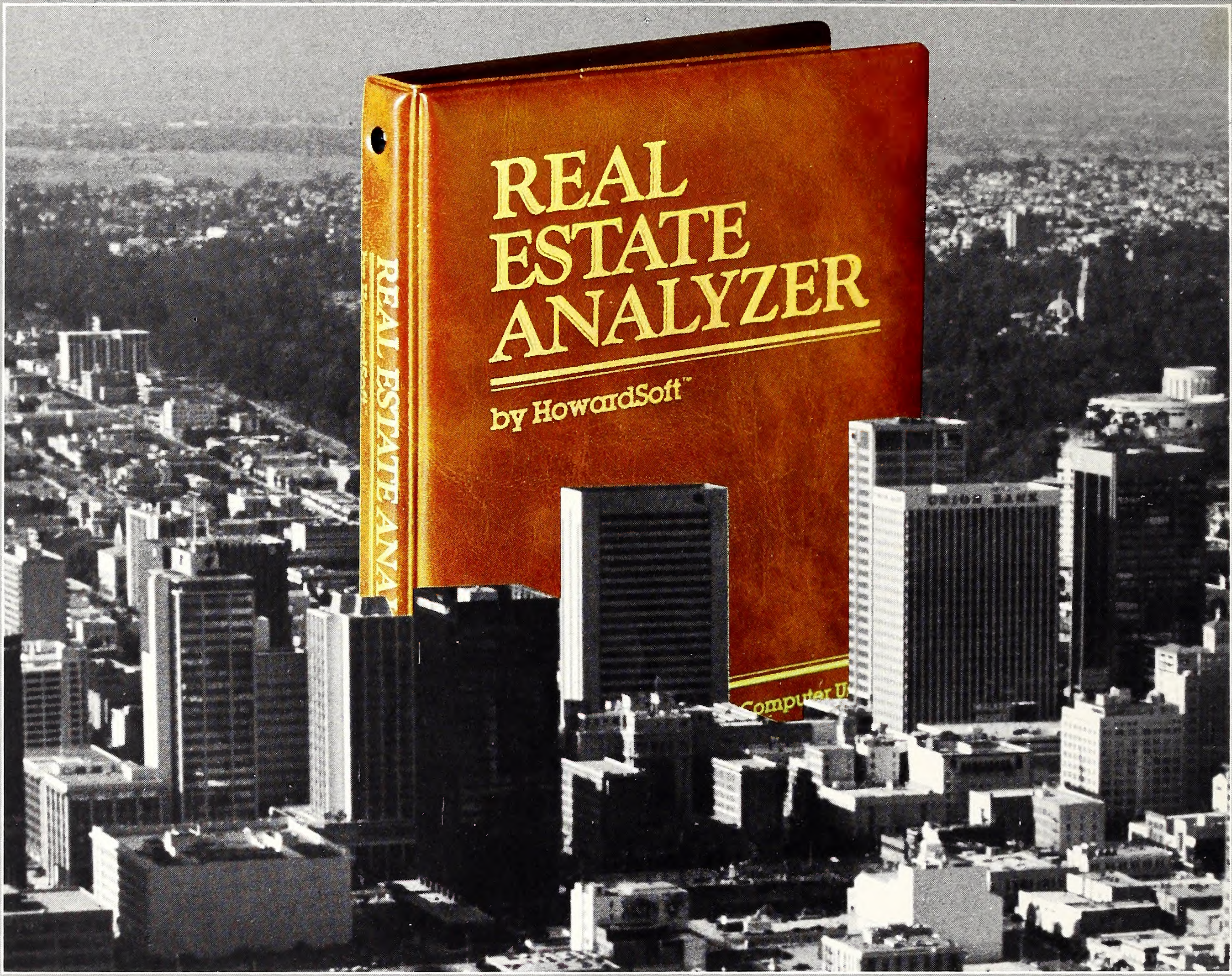


HERBIE HANCOCK

THE MUSICAL APPLE - EXEC SYNTAURI

III BITS - THE WEST COAST COMPUTER FAIRE

THE BEST STANDS OUT



*PRACTICAL DECISION-MAKING
aids buy/hold/sell decisions by projecting cash flow & profitability; computes ROI, IRR, ROE, and FMRR.

*REALISTIC SIMULATIONS
models economy based on 5 separate components of inflation for realistic "what-if" studies

*UP-TO-DATE WITH CURRENT MARKET
takes into account rent control, runaway inflation, money rates, creative financing, and complex lease packages

*UP-TO-DATE WITH TAX LAWS
includes latest tax laws for depreciation (including ACRS), recapture, & capital gains

*PROFESSIONAL PRINTOUTS
detailed 10-year tables of cash flow and ROI in personalized, paginated report format

*ERROR TOLERANT
designed for ease of use by newcomers to computers; fully tested by real estate professionals

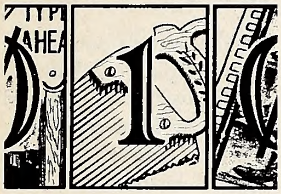
*INSTRUCTIVE MANUAL
100-page book includes easy-to-follow examples and detailed instruction in investment analysis

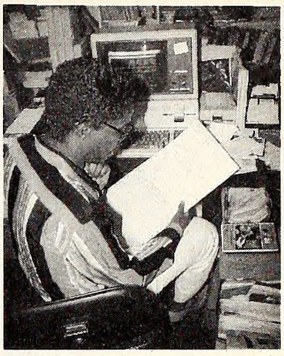
The best investment clearly stands out when you use computer-aided decision-making with the Real Estate Analyzer by HowardSoft. Applying the latest techniques in investment analysis, this software produces detailed, objective projections of cash flow and overall investment return for any commercial or residential property. Even the most dissimilar deals are compared fairly because inflation rates, creative loan packages, complex depreciation schedules, involved leases, and complex tax laws are all taken into account.

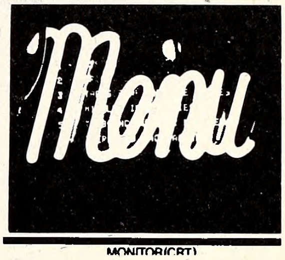
for all Apple and IBM-PC Computers
at your local dealer

HSS Howard Software Services
for the SERIOUS Personal Computer User
8008 Girard Ave., Suite 310 | La Jolla, CA 92037 | (619) 454-0121

SOFTALK







FEATURES · DEPARTMENTS

IN NEXT MONTH'S SOFTALK

Jujube June . . . Arabs like Apples, too—Arabian horses, that is . . . Exec Amdek . . . Parapsychology—real or imagined? Apples help find out . . . Life Begins at Eighty . . . M.A.C. Gate solves another computer mystery . . . and more. . . .

INDEX OF ADVERTISERS

Chairman	John Haller
Publisher	Al Tommervik
Editor	Margot Comstock Tommervik
Art Director	Kurt A. Wahlner
Managing Editor	Patricia Ryall
Associate Editors	Jean Varven
	David Hunter
	Andrew Christie
Special Assignments	Jonathan Miller
Open Discussion Editor	Tommy Gear
News and Trade Editor	Matthew T. Yuen
Reviews Editor	Roe Adams
Assistant Editor	David Durkee
Copy Editor	Cordell Cooper
Proofreaders	Harry McNeil
	Betsy Barnes
Contributing Editors	
Assembly Language	Roger Wagner
Pascal	Jim Merritt
Business	Peter Olivieri
Apple CP/M	Greg Tibbetts
Apple III	Taylor Pohlman
Hardware	Jeffrey Mazur
Applesoft	Doug Carlston
Investing	Kenneth Landis
DOS	Tom Weishaar
Graphics	Mark Pelczarski
Financial Modeling	Joe Shelton
Basic Solutions	William V. R. Smith
Art Manager	Kevin McKeon
Art Production	Don Robertson
Art Assistants	Timothy Durr
	Lucas McClure
	Michael G. Pender
Associate Publisher	Mary Sue Rennells
Director of Operations	Three Tyler
Accounting	Evelyn Burke
Accounting Assistants	Mary Jo Milam
	Carla Swanson
Dealer Sales	Judy Weinstein
	Dan Yoder
Book Sales	Lois Mencsik
Advertising Coordinator	Linda McGuire
West Coast Sales	Mike Antich Softalk 11160 McCormick Street Box 60 North Hollywood, CA 91603 (213) 980-5074
East Coast Sales	Ian Ross Paul McGinnis Advertising Sales 898 Broadway Massapequa, NY 11758 (212) 490-1021
Midwest and Rocky Mountain Sales	Ted Rickard John Sienkiewicz John Bollweg Kevin Sullivan Market/Media Associates 435 Locust Road Wilmette, IL 60091 (312) 251-2541
Circulation	
Trial Subscriptions	Hal Schick
	Deirdre Booth
	Laurie O'Connell
	Marsha Stewart
	Laurie Berger
Paid Subscriptions	Pam Kelley
	Connie Mastelotto
	I. K. Frasca
Back Issues	Ron Rennells
List Maintenance	Pat Adams
Assistant	Cliff Martinez
Systems	Sam Pawley
Guest Reviewers	Dave Albert, William H. Harrington, Robert Moog, Tom Repstad, Sharon Stea, Ethan Winer, Robert Woodhead

Cover: Internationally acclaimed jazz musician Herbie Hancock at home. Photo by Kurt Wahlner.

Credits: Composition by Photographics, Hollywood, California. Printing by Volkmuth Printers, Saint Cloud, Minnesota.

Softalk. Volume 3, Number 9. Copyright © 1983 by Softalk Publishing Inc. All rights reserved. ISSN:0274-9629. *Softalk* is published monthly by Softalk Publishing Inc., 11160 McCormick Street, North Hollywood, California; telephone (213) 980-5074. Second-class postage paid at North Hollywood, California, and additional mailing offices.

Postmaster: Send address changes to Softalk, Box 60, North Hollywood, CA 91603.

Free Subscription: Complimentary trial subscription to all owners of Apple computers in the USA. If you own an Apple, but you've never received *Softalk*, send your name, address, and Apple serial number with a request for subscription to Softalk Circulation, Box 60, North Hollywood, CA 91603. *Softalk* is totally independent of Apple Computer Inc.; sending your warranty card to Apple Computer will not inform Softalk of your existence.

Paid Subscription: $24 per year without sponsor, $18 per year with sponsor. At the end of trial period, each subscriber will be notified; response is required only if you wish to continue receiving *Softalk*. Lack of response will be taken as your choice to discontinue the magazine. Special rates for schools and libraries, $12; additional suhscriptions for schools and libraries, $8 each.

Back Issues: $2 through February 1981; $2.50 through July 1981; $3.50 through September 1982; $4.00 thereafter. November and December 1980, January, February, March, September, October, and November 1981, and December 1982 are sold out. December 1981, February and May 1982, and February 1983 are in short supply.

Problems? If you haven't received your *Softalk* by the fifteenth of the month, or if you have other problems with your subscription, Hal Schick or Pam Kelley can help out. Call (213) 980-5074.

Moving? Send new address and old to Softalk Circulation, Box 60, North Hollywood, CA 91603; telephone (213) 980-5074.

CONTEST: RING AROUND THE MAYPOLE

May.

There's the month of May, May Flowers, May Day, "Mother, May I?" May pull syrup, Maymie Eisenhower, the Mayflower, May din Japan, and Maypole. Maypole? That's a good one; it's contest time.

You don't see a whole lot of Maypoles around anymore. But they were fun while they lasted. People used to dance around them while holding on to brightly colored streamers. Now, in case you're picturing a bunch of party animals getting drunk and doing disco steps around a pole, it wasn't quite like that.

Here's how it worked. There was this great big pole with streamers attached to the top. Kids would form a circle around the pole, each one holding on to a streamer. Numbered from one to however many kids there were, the odd-numbered ones would start walking in one direction around the pole, and the rest would walk in the opposite direction.

At first, they all bumped into each other after taking two steps. Then they got the great idea of weaving over and under each other's streamers. This had two nifty effects. One, the kids didn't bump into each other anymore; and two, the pole ended up elaborately decorated with braided colored streamers.

Well, this month we're going to make our own Maypoles, only we're going to use word chains as streamers. A word chain is a sequence of words, names, or phrases that link from one to the next by words in common, words that sound alike, puns, or special relationships. An example: baseball . . . Yankees . . . keystone . . . cops . . . Copts and Robbers . . . Sirius Software . . . dogstar . . . and so on. Simple.

Our Maypole has six streamers, which means you'll have to make six word chains. And just to create a little brain twisting, we'll add one small catch. Each time streamers (chains) "cross," the links in each chain must also link to the streamer it's crossing. For instance, if our chain given in the previous example were to cross another chain at the link, "dogstar," then the other chain's link must link to "dogstar."

Sure it sounds confusing; and it's even more tricky than it sounds. But just to show that nothing is impossible, we've included an example of a Maypole on the next page.

Send in your entry in matrix form, just like the example on the next page. Each box contains two links. The top link in each box moves from left to right, and the bottom one moves from right to left. In our example, "baseball diamond" in box A (top) moves to box B and changes to "Diamonds Are Forever," while "Sir-tech" in box C (bottom) goes left to box B and changes to "Knight of Diamonds."

Start your first turn by filling in boxes A, C, and E. On the next turn, the top links will move from A to B, from C to D, and from E to F. The bottom links, on the other hand, will move from A to F, from C to B, and from E to D (whew!).

In other words, for each turn, each link moves to the box on its left; each bottom one moves to the box on its right. And don't overlook wraparound. When a top link gets to box F, it moves next to box A; when a bottom link gets to box A, it next moves to box F. It's easier if you just picture the top row as words on a conveyor belt going one way, and the bottom row as words on a belt going the other way. Each time the rows line up, the words have to change, yet still relate to each other.

In our example, each chain has a different color to make the chain easier to follow. We've even left the fifth turn empty for you to try your hand at Maypole building.

Scoring. Give yourself ten points for each link of each chain. You also get twenty bonus (extra) points for each link that represents an Apple-related company or product. In other words, Apple-related links are worth thirty points.

The person with the highest score will reign as the Softalk Maypole Dance Champ and will receive $100 in all sorts of wonderful stuff made by our advertisers. Send in your entry matrix with your name, address, phone number, and what you'd like to win to Softalk Maypole, Box 60, North Hollywood, CA 91603, postmarked by June 10, 1983.

Name ______________________

Address ______________________

City, State, Zip ______________________

Phone ______________________

If I win, I want ______________________

The Home Accountant.™ The #1 best-seller.

Any home finance package will balance your checkbook. But to become the #1 best-seller you've got to be something special.

The Home Accountant™ is.

It's the only one that prints a net worth statement and a personal finance statement. So you know exactly where you stand financially every day of the year. It will even print your checks, automatically.

Not only that, The Home Accountant™ lets you label every transaction. Just imagine sitting down to do your taxes and having every penny you've spent and earned neatly listed by category—and available at the touch of a button. It's an incredible time-saver.

You can also create bar, line and trend analysis graphs for every category—in color. It's great for realistic budgeting.

Sound amazing? Wait, there's more.

Let's say you write a check to pay your Visa. The Home Accountant™ automatically debits your checking account and credits your Visa account.

And it does this with every one of the two hundred* budget categories: credit cards, checking accounts, money markets, cash, rent checks, insurance payments—you customize your own financial package.

Check out The Home Accountant™ soon. You'll find it does a lot more than simply manage your money.

It manages your money simply.

*The Home Accountant™ is available for the Apple II/IBM Personal Computer/Atari 400/800 Computers/Osborne/TRS 80 Model III/Commodore VIC 64. The actual budget capacities will vary with each computer.

Continental Software
A Division of Arrays, Inc.

Apple II is a registered trademark of Apple Computer, Inc. IBM Personal Computer is a registered trademark of IBM Corp. Atari 400/800 are registered trademarks of Atari, Inc., a division of Warner Communications, Inc. Osborne is a registered trademark of Osborne Computer Corp. TRS 80 Model III is a registered trademark of Tandy, Inc. Commodore VIC 64 is a registered trademark of Commodore Business Machines, Inc.

It sells the most, because it does the most!

Continental Software Co., 11223 South Hindry Avenue, Los Angeles, California 90045 Telephone (213) 417-3003 • (213) 417-8031

TURN:	BOX: A	B	C	D	E	F
1	BASEBALL DIAMOND ← CUBS	→ ←	SIR LANCELOT SIR-TECH	→ ←	JOHN CANDY MILK DUDS	→
2		DIAMONDS ARE FOREVER KNIGHT OF DIAMONDS		KING ARTHUR DUDLEY MOORE		SECOND CITY T.V. CHICAGO
3	COMEDY SATURDAY NIGHT LIVE		JAMES BOND ROGER MOORE		ARTHUR ASHE JANE BYRNE	
4		LAF PAK: SPACE RACE MOONRAKER		U.S. SAVINGS BONDS FLAG BURNER		ASH WEDNESDAY EASTER SUNDAY
5						

CONTEST WINNERS

Oracle Final Four's Final Two. In the world of college sports, most fans will remember 1983 as the Year of the Ulcer. Basketball tough kid Ralph Sampson went home for the fourth time without a championship ring, the NCAA swimming and diving championships were decided in the last .48 seconds of the last event, and Georgia football hero Herschel Walker turned professional prematurely—in that *other* league.

Entrants in *Softalk*'s Oracle '83 contest, however, will remember it as the year everything went wrong in the NCAA basketball tournament. That's a bit misleading, though, because it was a great tournament for the underdog.

Unfortunately, when it came to predicting which teams would make the NCAA's Final Four, most contestants took a look at the national rankings as they were at the time (December 1982) and made their selections based on that. If Oracle contestants had their way, it would have been Virginia, UCLA, Indiana, and Georgetown. After all, those were the teams that were heading the wire services' and coaches' polls back in December, and they were the teams that most people picked to be on the road to Albuquerque. Most, but not all.

One of those "not alls" was Paul Duckenfield (Clemson, SC), the winner of Oracle '83, part 2. Of the hundreds and hundreds of entries, most couldn't name even one team; some were able to come up with one. The most teams anyone could name was two, and Duckenfield was one of only three persons to do so. The awesome *Softalk* Random Number Generator came back from vacation just in time to pick Duckenfield as the winner.

Duckenfield predicted Virginia (nope), Georgetown (still nope), Louisville (yep), and Houston (yep again) to make the trip to Albuquerque. Even though he was able to predict only half the teams, it was good enough to win him the customary $100 in prizes for his Apple. When we called to let him know he had won, Duckenfield was too overcome with joy to decide what he wanted to get. Probably some games? A joystick? How about some utilities? "Um, um, um, I dunno."

For those who got disgusted and quit following the tournament when their predicted teams were eliminated, North Carolina State

squeaked by Virginia, and Georgia beat the tar out of the North Carolina Tar Heels to make the trip to the Final Four as well.

The Peru Brothers, Shawn McLaughlin (Peru, NY) and Larry Houston (Peru, IN), were the only others who could name two teams. McLaughlin picked Houston (the school, not Larry) and Louisville, while Houston (Larry, not the school) tagged Louisville and North Carolina State to make the final cut.

A special mention goes to James M. Wilson III (Skaneateles, NY), who was the only one to predict that Georgia would be one of the last four teams left. That's not a bad feat, considering that this was the first time in the school's seventy-eight-year basketball history that it's ever been invited to take part in the tournament. And who would have guessed at the beginning of the season that Georgia would even be invited? Only Wilson.

As for those ten bonus points for predicting the winner . . . sorry, gang. You could count on one hand the number of people who predicted Houston and North Carolina State to make the Final Four, but not one person predicted that North Carolina State would pull a slam dunk out of nowhere with one second left to win the national championship.

Before we get rolling with the results of February's contest, a special thanks goes out to Marlene Sutton (Watertown, SD), who sent the most thoughtful get-well card to our injured mail carriers. The mail carriers thank you, the guard dog thanks you, and we thank you.

Now, the Juicy News. For a lot of contestants in our Funky February Puns 'n' Anagrams contest, February came to an end none too soon. The contest was mean; the clues killers. One man who didn't think it was too bad is Lauren Flewelling (Boxboro, MA), who won the first part of the contest, solving the puns-and-anagrams clues to complete the crossword puzzle.

Flewelling, an avid reader of *Games* magazine, spent just a few days deciphering, demystifying, and depunning all the clues to the crossword monstrosity. His $100 certificate will go toward the purchase of Sensible Software's *Sensible Speller*, which he'll pick up at his local store, the Game Shop, in nearby Acton.

When putting together clues for the contest, *Softalk*'s contest conjurers laughed with diabolical glee; they thought no one would even come close to solving them all. By the time the last entry was wrenched from a bruised and tattered mail carrier's hand, the contest conjurers weren't laughing very loud. Quite a number of contestants managed to get all but one, and a relative handful turned in a perfect score.

Rest assured, all you who tore your hair out and kicked the dog because you thought some clues had no answer; the answers to the crossword appear at the end of this section.

The second part of the contest, figuring out what each answer had to do with the month of February, involved much more than an agile mind; it required a good pair of feet to walk down to the library and a good set of eyes to look up all those names, events, and dates.

Even though contestants' success with part 1 put a damper on things at the contestmeisters' Valentine's Day party, part 2 managed to arch a few supercilious eyebrows. Nobody got all the dates right.

But that's okay; there were enough obscure names and dates to drive even the best researchers batty. Leading the way in looking-up-stuff skills was Bruce Buzbee (San Jose, CA), who got every name, event, and date right except for four of them. Selecting prizes presented somewhat of a problem for Buzbee. "If I win the crossword puzzle part," he wrote on his entry, "I'd like to get *The Arcade Machine* from Broderbund and BudgeCo's *Pinball Construction Set* (that should keep me busy for a while)."

However, he also wrote, "If I win the date-finding part, I promised my wife she could have whatever she wants. She would like to get Sirius's *Type Attack* and Synergistic's *Microbe*. If I win both parts, I'd like to get all of the above and retire as the undefeated *Softalk* Crossword Solver Supreme."

Well, since Buzbee did win the second part, is he going to go away empty-handed and let his wife have what she wants? "No way. We're going to split the prizes."

Now, the Dirt. Three entries that caught our eye were those by Alice Asling (Davis, CA), Jason Kantor (Palo Alto, CA), and Dick Timberlake (Colorado Springs, CO), who solved clue twelve across to be "Pia [Zadora]." We're still scratching our heads trying to figure out how they arrived at that answer.

The name of the game for the Jeffries (Denver, CO) household was team effort. Accompanying entries by Ruth, Tony, and William Jeffries was a cover letter that said, "Enclosed are three identical entries because this has been a family venture. No, we didn't put in an entry for every member of the tribe. Steve said it was sheer madness, Patty kept the fiends in food, Kenney took phone calls, and Nancy let the dog out."

We think people like Russell Negus (Mansfield, MA) are just swell. Negus wasn't picky about what he wanted for prizes if he won. If you use a little recollection, you'll remember that the coupon asked contestants to fill in their choices of preferred prizes after the prompt, "Gee, I'd like to win: ." But Negus didn't care what he won; he changed his entry blank to read, "Gee, I'd like to win!!!" Sorry, Russell. You didn't win.

In addition to being one of the three who figured Pia Zadora to be "a most beautiful woman," Jason Kantor is also this month's *Softalk* Slob. Kantor's entry arrived complete with smeared ink, frayed edges, grease stains, and marks that looked like the remains of some sort of brown fluid. His penmanship was nice, though.

Here are the answers to the February Puns 'n' Anagrams contest:

Across

2. **Eniac.** "Mutiny retarded" = *Caine* reversed; "Able [Abel] brother" = Cain; "shows tuby computer" = definition of event. Eniac was unveiled February 14, 1946, and dedicated on February 15.

4. **Bottlecap.** "Spinning instrument with

YOU WILL NEVER AGAIN HAVE TO WASTE TIME WAITING FOR YOUR PRINTER.

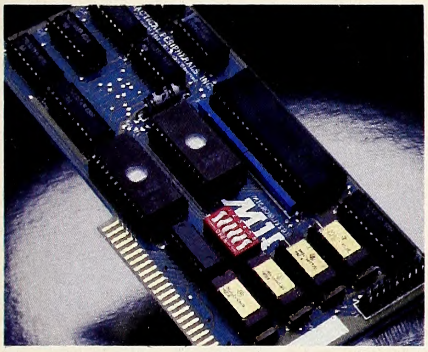
MICROBUFFER™ ALLOWS YOU TO PRINT AND PROCESS SIMULTANEOUSLY.

Microbuffer will instantly increase your efficiency — and eliminate the frustration of waiting for your slowpoke printer.

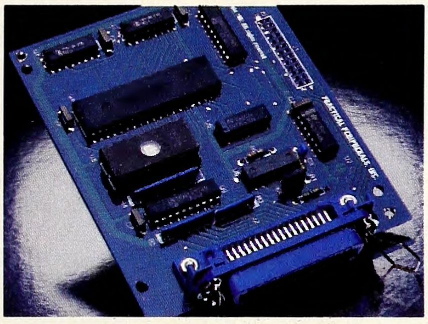
Now you can simply dump your printing data directly to Microbuffer and *continue processing*. Microbuffer accepts the data as fast as your computer can send. It stores the data in its own memory buffer, then takes control of your printer.

THERE IS A MICROBUFFER FOR ANY COMPUTER/PRINTER COMBINATION.

Whatever your system, there is a specific Microbuffer designed to accommodate it.



FOR APPLE II COMPUTERS, Mircobuffer II features on-board firmware for text formatting and advanced graphics dump routines. Both serial and parllel versions have a power-efficient low-consumption design. Special functions include Basic listing formatter, self-test, buffer zap, and transparent and maintain modes. The 16K model is priced at $259 and the 32K, at $299.



FOR EPSON PRINTERS, Microbuffer/E comes in two serial versions — 8K or 16K (upgradable to 32K) — and two parallel versions — 16K or 32K (upgradable to 64K). The serial buffer supports both hardware handshaking and XON-XOFF software handshaking at baud rates up to 19,200. Both interfaces are compatible with standard Epson commands, including GRAFTRAX-80 and GRAFTRAX-80+. Prices range from $159 to $279.

ALL OTHER COMPUTER/PRINTER COMBINATIONS are served by the stand-alone Microbuffer In-line.

The serial stand-alone will support different input and output baud rates and different handshake protocol. Both serial and parallel versions are available in a 32K model at $299 or 64K for $349. Either can be user-upgraded to a total of 256K with 64K add-ons — just $179 each.

SIMPLE TO INSTALL.

Microbuffer II is slot-independent. It slips directly inside the Apple II in any slot except zero.

Microbuffer/E mounts easily inside the existing auxiliary slot directly inside the Epson printer.

The stand-alone Microbuffer is installed in-line between virtually any computer and any printer.

MICROBUFFER FROM PRACTICAL PERIPHERALS.

So what are you waiting for? Write to us for more information or ask your dealer for a demonstration.

When you see how much freedom Microbuffer will allow, you'll understand why it's so silly to be without one.

PRACTICAL PERIPHERALS, INC.™
31245 LA BAYA DRIVE
WESTLAKE VILLAGE, CA 91362
(213) 991-8200

. . .game" = bottle (as in "spin the"); "blank" = cap (as in gun); "*bell top cat* strangely without annex" = anagram for bottlecap, dropping one L; stopper = definition. The bottlecap was patented on February 2, 1892.

8. **Malcolm X.** "Thousand" = M (Roman) add to "odd *calm lox*" = anagram Malcolm X; "Muslim leader" = definition; "to the end" = clue to event. Malcolm X shot to death in Harlem on February 21, 1965.

9. **Rand.** "*Darn*" = anagram; "objective author" = author and founder of Objectivism; "sci-fi" = *Atlas Shrugged* was science fiction. Ayn Rand, born February 2, 1905.

11. **GPO.** "In the beginning, *g*ames and *p*uzzles *o*nly" = take beginning letters of games, puzzles, and only; "sound like male institution" = horrible pun on mail; "institution" = definition; GPO = General Post Office. Thomas Neale got patent for North American Post Offices on February 17, 1692. George Washington signed Postal Service Act on February 20, 1792.

12. **Liz.** "Within Ba*li*, *Z*anzibar" = hidden word; "most beautiful woman" = clue to person. Elizabeth Taylor, born February 27, 1932.

16. **Los Angeles Earthquake.** Charade: "City . . .planet. . .fear" = Los Angeles, Earth, quake; "what's shakin'?" = descriptive clue. Los Angeles earthquake, February 9, 1971.

17. **Ted.** Referential definition: "Younger brother" of Kennedy clan, brothers in Washington, which was nicknamed "Foggybottom"; "shouldn't be [no B] in *debt*" = det; "it's backward" = reverse to get ted. Ted Kennedy, born February 22, 1932.

20. **A. Johnson.** "Baptist leader" = John (the Baptist); "leader of the reconstruction" = definition; "in one child" = container; put *John* within *a son*. On February 24, 1868, Congress decided to impeach Andrew Johnson.

21. **Glenn.** "*Len$ n. g.*" = anagram; "Tom Corbett" (astronaut on early television); (C)orbett = clue to event. John Glenn orbited the earth three times on February 20, 1962.

23. **Kilby.** "Icy maker" = IC maker, significance of person; "twists *by, li*kes" = hidden anagram; "swat a wasp" = charade—kill bee (sort of). Jack Kilby awarded the Medal of Science for the integrated circuit on February 16, 1958.

24. **N.Y. Subway.** "Mixed up *bus yaw*ns" = anagram with "no essen"—less one S and N—in it; "rapid transit" = definition; "initting city" = initials of New York. New York Subway unveiled on February 26, 1870.

27. **Maine.** Double reference: "Remember . . .a ship" = "Remember the Maine," a popular slogan in history; "the whole nation follows" = reference to election time saying, "As Maine goes, so goes the nation"; "swerving around *a mine*" = anagram; ship = definition. The Maine was blown up in harbor on February 15, 1898.

28. **Reagan.** "Old-time cowboy . . .politic" = descriptive definition; "Eastern" = location clue. Ronald Reagan, born February 6, 1911.

29. **National League.** "*Strange U.N. allegation, e.a.*" = anagram; "countrywide association" = definition; "dodgers" = reference. Baseball's National League formed on February 2, 1876.

30. **Pluto.** "*O tulips* isn't" drop "is" from "O tulips" and then anagram; double definition: orange dog = Disney's dog, Pluto, "planet" = second definition. The planet Pluto was discovered February 18, 1930.

Down

1. **Abscam.** "*A*ble *B*aker ripoff" = charade for A B scam; "no eyes. . .*ASC*II. *Bam*" = take out I's to get Asc Bam and anagram. On February 2, 1980, the Federal Bureau of Investigation revealed the results of its two-year investigation of public officials and stolen works of art.

3. **Chicago Seven.** "Oddly, *seven* years a*go, a chic*" = anagram and hidden word; "protesters" = definition; "upset the processes of the nation" = clue to event. February 18, 1970, was when the Chicago Seven were found innocent of conspiracy to incite riot. Five of them, however, were found guilty of crossing state lines to do so.

5. **Cochrane and Boone.** "Rooster hurried" = charade: cock ran; "Pat or Dan'l" = Boone; "inventors" = definition; "very tiny potato flake" = microchip—what they invented. Cochrane and Boone got patent for first micro on a chip, February 14, 1978.

6. **Galileo.** Charade: "jeune fille" = gal; "Langtry" = Lil; "MGM's lion" = Leo; "old astronomer" = definition. Galileo Galilei, born February 15, 1564.

7. **Iwo Jima.** "*Iowa Jim* was crazy" = anagram; "Marine's great victory" = descriptive definition; "Raise the flag" = clue to event. Marines landed on Iwo Jima February 19, 1945. They raised the flag on Mount Suribachi and posed for photos and statues of the event on February 23, 1945.

10. **DPT.** "A shot" = definition; "at kids" = reference to who gets DPT shots; charade extended description: "cool off" = *d*iptheria; "don't cough" = *p*ertussis; "jaw loose" = *t*etanus.

11. **GOP.** "*Pog*" = anagram; "elephants" = reference to mascot; "party" = definition. The Republican party formed on February 28, 1854. Its first convention was on February 22, 1856.

13. **Fidel.** "Messy *field*" = anagram; "faithful" = translation; "first in cigars" = charade definition: premier in Havana. Fidel Castro became the Premier of Cuba on February 16, 1959.

14. **Florida.** "*Id*entify *Flora*" = I.D. Flora, anagram; "she'll be in a real state" = event. Spain ceded Florida to the United States on February 22, 1819.

15. **CSA.** "Turn around. . .start of *As*cot" = reversal; "dixie" = definition by nickname; "rebels" = referential definition. First six states met to form the Confederate States of America on February 4, 1861. They drew up a provisional constitution on February 8, 1861.

16. **Linus Pauling.** "*Line is appalling*" = very bad pun; "scientist" = definition. Linus Pauling, who suggested the use of vitamin C to fight colds, born February 28, 1901.

18. **Dagwood.** "Dogwood. . .mid$(tree$,2,1) = programs to Dagwood; "sandwiches, blondes," "bums instead," "funnies" = lots and lots of clues. Dagwood Bumstead married Blondie Boopadoop in the comic strip *Blondie* on February 17, 1933.

19. **Freedom Day.** "Twenty-four hours' leave" = definition, sort of; "mais non, *Freddy and Moe* screwed up" = anagram, dropping n (non = no n). Freedom Day is February 1, to commemorate the signing of the Thirteenth Amendment on February 1, 1865, which abolished slavery. The first Freedom Day was February 1, 1949.

22. **Hearst.** "*Her s*tory" = another bad pun; "*pat*tycake" = first name, clue to person; "kid . . .nap" = clue to event. Patty Hearst was kidnapped by members of the Symbionese Liberation Army (SLA) on February 4, 1974.

23. **King Tut.** "*Cutting* the C but not the *K*" = take C out, put in K to form anagram; "tongue-clucking from British mother" = tut, tut; "royal . . .young monarch" = definition. The sepulchral chamber to King Tut's tomb was opened on February 16, 1923. The sarcophagus was opened February 12, 1924, and the casket on February 11, 1927.

25. **BSA.** Charade: "short lad" = boy; "Tontos" = scouts; "country" = America; "prepared" = motto; "troops. . .dens" = reference clues. The Boy Scouts of America was incorporated February 8, 1910.

26. **Alamo.** "Crock it" = extremely bad pun on (Davy) Crockett; "remember the mission" = play on "Remember the Alamo"; charade: "first" = A, "brief Louisiana and Missouri" = abbreviate states' names, LA and MO; double definition: "mission" and "battle" = event. Santa Anna's army entered San Antonio to begin its attack on the Alamo on February 23, 1936.


CROWE
MEET
SAMMY LIGHTFOOT
TM
OUR HOT NEW STAR!
Requires Apple II/II+ or Apple IIe (48K) and one disk drive.
Sierra On-Line Building, Coarsegold, CA 93614
SIERRAVISION TM
TM designates a trademark of SIERRA ON-LINE, INC.
Sierra ON-LINE INC

FASTALK

Fastalk is your quick guide to popular, specialized, or classic software. Programs appearing in Fastalk must meet one or more of the following criteria: (1) equal or surpass in sales the least-selling program to appear on any of the current bestseller lists; (2) relate to a specialized subject area and be in general distribution (more specialized packages and areas will be included as Fastalk matures); (3) be new and of professional quality (such programs will be carried for one month only—after that, they must meet other criteria for inclusion); (4) stand out as extraordinary.

Designation as a classic is noted by a bullet preceding a program's title.

Where opinion is expressed, *Softalk* has seen the software in question; the date of *Softalk*'s review, if any, is given at the end of the item.

Softalk may arbitrarily omit any package from Fastalk, whether or not it meets the foregoing criteria.

Adventure

•**Adventure.** Crowther, Woods. The original text adventure, created on mainframe, contributed to by many over a long time. Very logical within fantasy framework, excellent puzzles, maps; complex, convoluted, and great. Several publishers: Microsoft, 10700 Northup Wy., Bellevue, WA 98004. $28.95. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $35. Frontier Computing, Box 402, 666 N. Main, Logan, UT 84321. $10.

Critical Mass. Blauschild. Nine days to race all over the world, solving sophisticated puzzles and getting through some tight places, to find and deactivate five thermonuclear devices and their owner. Hi-res graphics. Sirius, 10364 Rockingham Dr., Sacramento, CA 95827. $39.95.

•**Cyborg.** Berlyn. Text adventure with brief action skill game hidden in plot. As a futuristic cyborg, you're lost in a strange forest, desperately needing food and power. In its realism and use of true plot, it represents one of the most significant advances in adventuring since the original *Adventure*. Sentient, Box 4929, Aspen, CO 81612. $32.95. *11/81.*

The Dark Crystal. Williams. Hi-res adaptation of popular fantasy movie. Puzzles to challenge even those who've seen the movie. Includes player option to let the Skeksis win. Sierra On-Line, Sierra On-Line Building, Coarsegold, CA 93614. $39.95. *4/83.*

Deadline. Blank, Lebling. Episode one in a projected series of murder mysteries by the authors of *Zork*. Interrogate, accuse, make transcripts. Includes inspector's casebook, lab report. Text. Infocom, 55 Wheeler St., Cambridge, MA 02138. $49.95. *8/82.*

Escape from Rungistan. Blauschild. Graphics adventure with some animated real-time puzzles. Espionage theme. Sirius, 10364 Rockingham Dr., Sacramento, CA 95827. $29.95. *8/82.*

Genesis. Pritchett. Adventure program generator. Develops standard format, two-word-parser adventures with rooms, objects, flags; up to 99 apiece. No program knowledge necessary whatsoever. Fun. Hexcraft, Box 39, Cambridge, MA 02238. $49. *4/83.*

•**Hi-Res Adventure #1: Mystery House.** Williams. Whodunit in a Victorian mansion. First adventure with pictures. 2-word parser with logical comprehension. Sierra On-Line, Sierra On-Line Building, Coarsegold, CA 93614. $24.95.

Hi-Res Adventure #2: The Wizard and the Princess. Williams, Williams. Attempt to rescue princess from vengeful wizard. First graphic adventure in full color. Sierra On-Line, Sierra On-Line Building, Coarsegold, CA 93614. $32.95. *11/80.*

Hi-Res Adventure #3: Cranston Manor. DeWitz, Williams. More full-color adventuring involving the redistribution of wealth. Long on great riddles, short on plot. Sierra On-Line, Sierra On-Line Building, Coarsegold, CA 93614. $34.95. *9/81.*

Labyrinth of Crete. Johnson, Pinero. Player is Jason and Hercules, simultaneously or independently, searching for golden fleece in a three-level labyrinth. Text with occasional graphics. Maps included. Adventure International, Box 3435, Longwood, FL 32750. $29.95. *3/83.*

Mask of the Sun. A unique animated graphic quest with full though sometimes frustrating parsing. Moving from room to room involves seeing scenery along the way go by—a graphics breakthrough with nice puzzles. Ultrasoft, 24001 S.E. 103rd St., Issaquah, WA 98027. $39.95. *11/82.*

New World. Decker. Representatives of Spain, England, and France dodge pirates, lousy weather, disease, bankruptcy, and each other as they vie for dominance in the exploitation of North and South America. Epyx, 1043 Kiel Ct., Sunnyvale, CA 94086. $29.95.

•**Prisoner 2.** Mullich. Totally relandscaped but loyal version of original game: full-color hi-res graphics added, puzzles reworded, obstacles expanded. Sophisticated and difficult exercise in intimidation with elements of satire. Escape from an island requires player to solve logical puzzles, overcome obstacles, and answer riddles. Excellent computer fare; nothing else like it. Edu-Ware, Box 22222, Agoura, CA 91301. $32.95. *The Prisoner, 3/81*; *Prisoner 2, 10/82.*

•**S.A.G.A. Series.** Adams. Scott Adams's prototypical adventures—12 in all—spruced up with 100-color graphics and Votrax vocals. Fun, not always logical, very story-oriented series. Each adventure has its own theme and often exotic locale. They map small but score big on imagination. Adventure Intl., Box 3435, Longwood, FL 32750. $29.95 each.

Sherwood Forest. Holle, Johnson. Dating game in legendary times. In premiere Softoon adventure featuring neat UltraRes graphics, Robin Hood woos Maid Marian all the way to the honeymoon. Go for it. Phoenix Software, 64 Lake Zurich Dr., Lake Zurich, IL 60047. $34.95.

Starcross. Science fiction prose adventure that comes wrapped in a flying saucer. Set in the year 2186, main puzzle is to discover raison d'etre of miniworld asteroid. Likable, engaging. Infocom, 55 Wheeler St., Cambridge, MA 02138. $39.95. *11/82.*

Suspended. Berlyn. Well-plotted prose adventure demands control of six independent robots who can act simultaneously. Intelligent, challenging exercise in logic. A milestone. Infocom, 55 Wheeler St., Cambridge, MA 02138. $49.95. *4/83.*

•**Swordthrust Series.** Set of adventures, seven so far, that integrate fantasy role playing. Create one character, make new friends in each adventure, battle monsters and achieve goals together. Good stories, fun to map. Vocabulary no mystery but puzzles are. Single character goes through all. CE Software, 801 73rd St., Des Moines, IA 50312. Number 1 prerequisite for rest. Each adventure, $29.95. *8/82.*

•**Zork I.** Part one of mainframe adventure; understands complete compound sentences and questions. Simultaneous manipulation of objects. Text, but so what. Infocom, 55 Wheeler St., Cambridge, MA 02138. $39.95. *6/81.*

•**Zork II.** Lebling, Blank. *Zork* comes into its own. Great text adventure technique and communication. Infocom, 55 Wheeler St., Cambridge, MA 02138. $39.95. *3/82.*

Zork III. Lebling, Blank. Text lives! A masterpiece of logic and a grand adventure to revel in. Hard, logical puzzle with unique point system. Benevolence conquers. Infocom, 55 Wheeler St., Cambridge, MA 02138. $39.95. *9/82.*

Business

Accounting Plus II and IIe. *II* version is integrated package; general ledger, accounts receivable and payable, and inventory-purchasing modules. Menu-driven; prompting. *IIe* version stripped and rebuilt to take advantage of available functions. Software Dimensions, 6371 Auburn Blvd., Citrus Heights, CA 95610. *II*, $1,250; *IIe*, $995.

Apple II Business Graphics. Converts numerical data into charts and graphs. Features mathematical and statistical functions. Requires 64K. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $175.

BPI System. Popular five-module business package; programs also available separately. Includes general ledger (a bestseller), accounts receivable, accounts payable, payroll, inventory control, and job costing. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $395 each; job costing, $595.

Cdex Training for VisiCalc. Brandt. Self-contained

Apple-assisted training program and reference guide for the #1 electronic spreadsheet. User-selectable information. Cdex, 5050 El Camino Rd., Los Altos, CA 94022. $49.95. *3/83.*

Computer Programmed Accountant. Five-module package: general ledger (very popular), accounts receivable, accounts payable, payroll, and property management. All other modules post automatically to general ledger. Continental, 11223 S. Hindry Ave., Los Angeles, CA 90045. $1,495; separate modules: $250 each; property management: $495.

The Data Factory. Passauer. Database management system allows listing files, getting file statistics, transferring records, and adding fields to update forms. Disk swapping required; excellent product overall. Several compatible products available. Micro Lab, 2310 Skokie Valley Rd., Highland Park, IL 60035. $150. *8/81.*

dBase II. Speedy relational database management system. Requires SoftCard. Ashton-Tate, 9929 W. Jefferson Blvd., Culver City, CA 90230. $700.

DB Master. Comprehensive database management system with password protection, extensive report creation options. 1,000 characters per record. Stoneware, 50 Belvedere St., San Rafael, CA 94901. $229. *10/81.*

DB Master Utility Pak #1 and **Utility Pak #2.** Compatible with version III. Translates DB files to Apple text, restructures existing files, replicates and merges, and recovers crashed files. *Pak #2* includes label printer, global editor, file merge, reblocker, and forms printer. Stoneware, 50 Belvedere St., San Rafael, CA 94901. $99 each.

Desktop Plan. Models and analyzes budgets, profits and losses, sales forecasts, cash flow; what-if calculations. VisiCorp, 2895 Zanker Rd., San Jose, CA 95134. $250.

Dow Jones Market Microscope. Stock analysis for money managers. Follows buy/sell indicators and allows sorting and ranking. Dow Jones Software, Box 300, Princeton, NJ 08540. $700.

General Manager. User-definable database management system; can use one to four disk drives or hard disk. Change screen and field formats without reentering data. Current version supports IIe and 80-column card at no extra cost. Sierra On-Line, Sierra On-Line Building, Coarsegold, CA 93614. $229.95. Hard disk version, $374.95.

The Incredible Jack of All Trades. Word processor, database, and spreadsheet, plus mailing label print and sort. Gives 80-column dual-case display automatically on the IIe, with 64K, 80-column card on II Plus. Business Solutions, 60 E. Main St., Kings Park, NY 11754. $129.

List Handler. List-lover's delight. Prints lists, labels, and letters. Handles 3,000 records per disk and eight disk drives. Takes requests. Silicon Valley Systems, 1625 El Camino Real, #4, Belmont, CA 94002. $89.95. *2/83.*

Multiplan. Easy-to-learn electronic work sheet using plain English commands. Powerful modeling and presentation capabilities. For use in analysis, forecasting, technical engineering, and the home. Versions 1.04 and up use 80 columns and extended memory of the IIe. Microsoft, 10700 Northup Wy., Bellevue, WA 98004. $275.

PFS:File (formerly *Personal Filing System*). Page, Roberts. User controls data in totally unstructured database. Up to thirty-two pages (screens) of information in each record. IIe version has 80 columns, u&lc. Software Publishing, 1901 Landings Dr., Mountain View, CA 94043. $125. *10/80.*

PFS:Graph. Chin, Hill. Works alone or interfaces with files created with *PFS:File* and *VisiCalc.* Produces bar, line, and pie charts merging data from several sources. 80 columns and increased graphics support in IIe version. Software Publishing, 1901 Landings Dr., Mountain View, CA 94043. $125. *5/82.*

PFS:Report. Page. Powerful report generator designed for use with *PFS:File.* Sorts, calculates, totals, formats, and prints presentation-quality columnar reports. Software Publishing, 1901 Landings Dr., Mountain View, CA 94043. $125. *6/81.*

Quick File IIe. Easy-to-use personal database filing system. Fifteen fields; files as long as disk allows. IIe, 2 disk drives. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $100.

Quick-Vis. A quick revision subroutine that adds Kraft joystick cursor control to *VisiCalc,* eliminating separate procedures required for cursor movement using keyboard. Kraft, 450 W. California Ave., Box 1268, Vista, CA 92083. $22.95.

Risk Simulator. Estimates probability distributions associated with risk situations, such as automobile maintenance expenses or employer funding of health benefits. Actuarial Microcomputer Software, 3915A Valley Ct., Winston-Salem, NC 27106. $185.

State of the Art General Ledger and **Budget and Forecasting Module.** The ledger does 12-period accounting, two-digit subaccounts; handles up to 470 accounts; enter 100 transactions before updating to permanent files. Budget module extends the account number to nine digits; custom designs reports; does previous year comparisons. State of the Art, 3183A Airway Ave., Costa Mesa, CA 92626. $495; budget module, $395.

SuperCalc. Interactive CP/M spreadsheet. Scrolls over entire worksheet, up to 63 columns and 254 rows, using cursor controls. Sorcim, 2310 Lundy Ave., San Jose, CA 95131. $295.

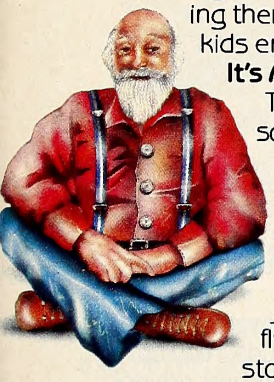
VersaForm. Business forms generator for invoicing, mailing lists, sales analysis, inventory. Hard disk compatible. Applied Software Technology, 14125 Capri Dr., Los Gatos, CA 95030. $389. *6/82.*

Videx Preboot VisiCalc. Prepares *VisiCalc* to run in 80 columns, u&lc. Advanced version uses mixture

This year, thousands of kids will be searching for the most amazing thing.

At Spinnaker, we don't believe in the "kill or be killed" concept behind most computer games. In fact, we believe computer games should be instructive. Not destructive. But just as importantly, they should be fun.

That's why IN SEARCH OF THE MOST AMAZING THING™ is designed to let your kids negotiate with aliens instead of destroying them. Because given the opportunity, kids enjoy using their minds.



It's Amazingly Fun.

The Most Amazing Thing is out there somewhere. Finding it won't be easy. But relax, your kids will have the help of their old uncle Smoke Bailey. He'll give them a B-liner (sort of a cross between a hot air balloon and a dune buggy) to use on their journey. They'll have to learn how to fly the B-liner and navigate it through storms and fog. But before they do anything, your kids will have to talk to Old Smoke. He'll tell them about the Mire People and the strange language that they speak. He'll also tell them to avoid the dangerous Mire Crabs and how to get fuel for the B-liner.

Your kids will visit the Metallican Auction where they'll trade with the aliens for valuable chips. Your kids will then use these chips to buy things they'll need for their trip. And your kids will learn how to fly over the planet using their jet pack.

The Most Amazing Thing holds great powers, but it will take great skill, persistence and imagination to find it.

It's Amazingly Educational.

IN SEARCH OF THE MOST AMAZING THING is written by Tom Snyder, educator and author of the best-selling Snooper Troops™ Detective Series.

And like all Spinnaker games, IN SEARCH OF THE MOST AMAZING THING has real educational value. For instance, your kids will sharpen their ability to estimate distances and quantities. And since they'll be navigating their B-liner, they'll become aware of distance, direction and time. They'll also develop a knack for economic and monetary principles through trading with the aliens. And they'll solve problems through trial and error.

They'll learn all of these things, plus they'll learn that nothing is impossible if you put your mind to it.

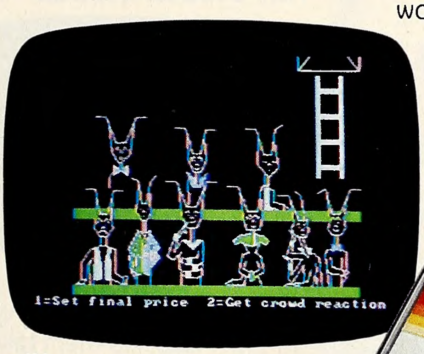
A Novel Approach to Computer Games.

Besides offering your children all of the above, IN SEARCH OF THE MOST AMAZING THING gives them an opportunity to develop their reading skills. Because included with the game is Jim Morrow's new novel The Adventures of Smoke Bailey.* So your children will have hours of fun reading the book or playing the game. And they'll be learning at the same time.

Parental Discretion Advised.

If you're a parent who would rather see your kids reason with aliens than destroy them, you've got plenty of reasons to ask your local software retailer for IN SEARCH OF THE MOST AMAZING THING. It's compatible with Apple,® IBM,® Atari,® and Commodore 64™ computers. And it offers so much fun you'll probably be tempted to play it yourself. Or you can write us directly at: Spinnaker Software, 215 First Street, Cambridge, MA 02142.

You'll find this is one computer game that won't alienate you from your children.



SPINNAKER™
We make learning fun.

SPINNAKER'S LINE OF EARLY LEARNING GAMES IS GROWING AS FAST AS YOUR CHILD'S MIND.

Watching your kids grow up is a lot of fun. But making sure their minds grow as fast as their bodies is even more rewarding. That's where we can help. With a growing line of Early Learning Programs that are not only lots of fun to play, but also educational.

Some of the games you see on these two pages help exercise your child's creativity. Others help improve vocabulary and spelling skills. While others improve your child's writing and reading abilities. And all of them help your child understand how to use the computer.

So if you're looking for computer programs that do more than just "babysit" for your kids, read on. You'll find that our Early Learning Programs are not only compatible with **Apple,® Atari,® IBM®** and **Commodore 64™** computers, but also with kids who like to have fun.

HEY DIDDLE DIDDLE™ Poetry in motion. Ages 3 to 10.

Kids love rhymes.

And since HEY DIDDLE DIDDLE features 30 classic rhymes with full color graphics and the neatest computer music you've ever heard, it makes rhyme games more fun than ever before.

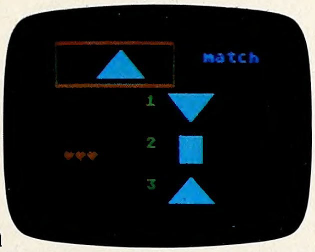
Plus, it makes it fun for kids to read, helps them understand how words and rhymes create poetry and lets them take fragmented thoughts and rearrange them to form coherent verse.

The story of STORY MACHINE.™ Ages 5 to 9.

STORY MACHINE is like a storybook come to life. Using the keyboard, your children write their own fun little stories. The computer then takes what they've written and animates their story on the screen, complete with full color graphics and sound.

STORY MACHINE helps your children learn to write correctly, become familiar with the keyboard, and lets them have fun exercising their creativity at the same time.

KINDERCOMP.™ Numbers, shapes, letters, words and drawings make fun. Ages 3 to 8.



KINDERCOMP is a game that allows very young children to start learning on the computer. It's a collection of learning exercises that ask your children to match shapes and letters, write their names, draw pictures, or fill in missing numbers. And KINDERCOMP will delight kids with colorful rewards, as the screen comes to life when correct answers are given.

As a parent, you can enjoy the fact that your children are having fun while improving their reading readiness and counting skills.

FACEMAKER™ makes faces fun. Ages 4 to 12.

FACEMAKER lets children create their own funny faces on the screen. Once a face is completed, your children will giggle with delight as they make it do all kinds of neat things: wink, smile, wiggle its ears, or whatever their imagination desires.

Plus, FACEMAKER helps children become comfortable with computer fundamentals such as: menus, cursors, the return key, the space bar, simple programs, and graphics. FACEMAKER won't make parents frown because their children will have fun making friends with the computer.

SPINNAKER™
We make learning fun.

of existing memory cards. Videx, 897 N.W. Grant St., Corvallis, OR 97330. $49; advanced: $89.

• **VisiCalc.** Bricklin, Frankston. Electronic work sheet for any problem involving numbers, rows, and columns. No programming necessary. VisiCorp, 2895 Zanker Rd., San Jose, CA 95134. $250. *10/80.*

VisiFile. Creative Computer, Jameson, Herman. Database management system for organization and retrieval of information, allowing sort and modification of records. VisiCorp, 2895 Zanker Rd., San Jose, CA 95134. $250.

VisiSchedule. Critical path PERT schedule planner. VisiCorp, 2895 Zanker Rd., San Jose, CA 95134. $300.

VisiTrend/VisiPlot. Kapor. Combines *VisiPlot* graphics with time-series manipulation, trend forecasting, and descriptive statistics. VisiCorp, 2895 Zanker Rd., San Jose, CA 95134. $259.95. *7/81.*

Communications

Address Dialer. Phone appointment management system for Novation Apple-Cat or Hayes Micromodem II. Automatic dial, redial, and date reminder, plus label printing and mailing list features. Christopher Systems, 2775 Glendower Ave., Los Angeles, CA 90027. Hayes version, $59; Novation, $79.

Apple Link. Jaffe, Pierce. Creates intelligent terminal at receiving end with no additional software. Only modem software known to man that can transmit *ScreenWriter* text files. Also transmits random access text files. Computer Applications, 13300 S.W. 108 Street Circle, Miami, FL 33186. $59.95.

ASCII Express: The Professional. Robbins, Blue. Greatly improved version of original modem software package features automatic redial, individual macro files, and conversion of Integer, Applesoft, or binary programs into text files. Works with a plethora of hardware. Southwestern Data, 10761-E Woodside Ave., Santee, CA 92071. $129.95. *12/82.*

Data Capture 4.0. Copyable, modifiable smart terminal program; compatible with Apple III and most lower-case adapters. Southeastern Software, 6414 Derbyshire Dr., New Orleans, LA 70126. $65.

Dow Jones Connector. Guide to the use of the company's news retrieval service and Blue Chip membership, too. Dow Jones Software, Box 300, Princeton, NJ 08540. $95.

Dow Jones News and Quotes Reporter. Telecommunications software that gives instant access to the Dow Jones News/Retrieval Service for viewing and printout of any desired reports and articles—but won't save them to disk. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $95. *2/82.*

Hayes Terminal Program. Standalone disk designed for the Micromodem II lets CP/M, DOS 3.3, and Pascal disks create, list, delete, send, and receive files. Opens access to nonkeyboard ASCII characters and prints incoming data as it is displayed. Hayes Microcomputer Products, 5835 Peachtree Corners East, Norcross, GA 30092. $99.

Micro/Courier. Electronic mail program. Provides file transfer of any DOS 3.3. file (correspondence, *VisiCalc,* charts) automatically and unattended, connected to another *Micro/Courier.* Built-in text editor; maintains 100 mailboxes; permits optional clock and calendar scheduling. Microcom, 1400A Providence Hwy., Norwood, MA 02062. $250.

Micro/Terminal. Access and exchange information with mainframes and minis, databases like the Source, and other remote terminals and personal computers. Allows keyboard mapping, u&lc, 80-column cards. Microcom, 1400A Providence Hwy., Norwood, MA 02062. $84.95.

P-Term: The Professional. Supports all Pascal-compatible interfaces, asynchronous serial cards, Apple-compatible modems, and baud rates up to 2400. Southwestern Data, 10761-E Woodside Ave., Santee, CA 92071. $129.95.

Tekterm. Intelligent graphics terminal software. Five modes: 70-column hi-res display, Tektronix 4010 graphics terminal simulation, 19,200 baud rate, or predefined automation of communications sequences. Fountain, 1901 Kipling, Lakewood, CO 80215. $90.

Transend 1, 2, 3. Intelligent terminal software with multiple hardware compatibility. Advanced, easy to use. 1 sends text only; menu driven, limited editor. 2 sends text and files like *VisiCalc*; verifies transmission. 3 does both and handles electronic mail with auto-redial, clock calendar, and password protection. Upgrade: difference in price between two packages plus $20 service fee. SSM, 2190 Paragon Dr., San Jose, CA 95131. $89, $149, $275. *9/82.*

VisiTerm. Hi-res 60-character display; wide range of protocols for sending text. Well-planned and comprehensive. VisiCorp, 2895 Zanker Rd., San Jose, CA 95134. $129. *9/81.*

Z-Term: The Professional. More than an update. Compatible with a great variety of modems, interface cards, and screen modes. Simple file transfer with integrity. Southwestern Data, 10761-E Woodside Ave., Santee, CA 92071. $149.95.

Fantasy

Ali Baba and the Forty Thieves. Smith. Fanciful Arabian Nights role-playing game with a sense of humor. Fresh, fast action, challenging options, and secrets that are a joy to discover. Quality, 6660 Reseda Blvd., #105, Reseda, CA 91335. $32.95. *11/82.*

Apventure to Atlantis. Clardy. Sequel to *Odyssey*. Many refinements including recruitable entourage of wizards with individual attributes. Included cheat sheet is invaluable. Synergistic, 830 N. Riverside Dr., #201, Renton, WA 98055. $40. *6/82.*

• **Beneath Apple Manor.** Worth. The original dungeon game for the Apple, created in 1978. Newly released version has hi-res, sound effects, a few more magic items, but still the classic game. Quality, 6660 Reseda Blvd., #105, Reseda, CA 91335. $29.95. *2/83.*

Empire II. Mullich. Second scenario in the promised Empire trilogy. With civilization at the apex of its power and complexity, you cut through red tape to gain freedom and dignity. Edu-Ware, Box 22222, Agoura, CA 91301. $32.95.

Galactic Adventures. Reamy. Role-playing science fiction adventure revision of *Galactic Gladiators* strategy game. 26 scenarios. Allows creation and saving of your own adventures. Strategic Simulations, 465 Fairchild Dr., #108, Mountain View, CA 94043. $49.95. *4/83.*

Knight of Diamonds. Second scenario of *Wizardry,* requiring thirteenth-level characters from the original. Individual quests on each of six dungeon levels. Great. Sir-tech, 6 Main St., Ogdensburg, NY 13669. $34.95. *7/82.*

Microbe. Clardy, Zalta. An internal course in medicine, disguised as a fantasy/adventure/arcade/simulation. "Enjoy your next viral infection!" Good game, great educational tool. Synergistic, 830 N. Riverside Dr., #201, Renton, WA 98055. $44.95.

• **Odyssey: The Compleat Apventure.** Clardy. Fantasy adventure far beyond one place and one setting. Castles, catacombs, an ocean voyage, and the orb of power. Synergistic, 830 N. Riverside Dr., #201, Renton, WA 98055. $30. *10/80.*

• **Temple of Apshai.** Lead title in Dunjonquest series, winner 1981 Academy of Adventure Gaming Arts and Design "Computer Game of the Year" award. Epyx/Automated Simulations, 1043 Kiel Ct., Sunnyvale, CA 94086. $39.95.

Ultima II. British. Faster play in a bigger universe with a time-travel option. Typically British look and feel. Events are much more interdependent; larger realm of fantasy with more transactions available. Sierra On-Line, Sierra On-Line Building, Coarse-

gold, CA 93614. $59.95.

• **Wilderness Campaign.** Clardy. First fantasy game to leave the dungeon for the great outdoors; first in hi-res; first to bargain with merchants; and more. Synergistic, 830 N. Riverside Dr., #201, Renton, WA 98055. $17.50.

• **Wizardry.** Greenberg, Woodhead. Ultimate role-playing fantasy; ten-level maze in hi-res. Generate twenty characters, six at a time on expeditions. Gripping game; superbly produced. Sir-tech, 6 Main St., Ogdensburg, NY 13669. $49.95. *8/81.*

Graphics

Alpha Plot. Kersey, Cassidy. Hi-res graphics and text utility with optional xdraw cursor and proportional spacing. Beagle Bros, 4315 Sierra Vista, San Diego, CA 92103. $39.50.

Banner Magic. Star. Just like it says. This and your printer will give you banners for all occasions, in seven-inch letters. Phoenix, 64 Lake Zurich Dr., Lake Zurich, IL 60047. $24.95.

The Complete Graphics System II. Pelczarski. A wealth of graphics tools at a reasonable price. Make 2-D drawings with game paddles, add text in destructive, nondestructive, or reverse modes, create 3-D figures and shape tables. Manual features complete outline of command structure. Penguin, 830 4th Ave., Geneva, IL 60134. $69.95; Apple Graphics Tablet version, $119.95. *7/81.*

GraForth. Lutus. A graphics language rewritten for maximum speed. Plotting, line, text display, character image, and high speed 3-D graphics, with variety of colors and drawing options. Includes music synthesizer. Insoft, 10175 S.W. Barbur Blvd., #202-B, Portland, OR 97219. $75. *8/82.*

The Graphics Magician. Jochumson, Lubar, Pelczarski. Outstanding animation package consisting of picture editor and shape table extender. Comes with utility program to transfer binary files. Penguin, 830 4th Ave., Geneva, IL 60134. $59.95; Apple Graphics Tablet version, $69.95. *5/82.*

The Graphic Solution. Graphics editor and bit-mapping animation system using film-editing techniques. Saves hi-res screen as standard DOS file. No programming knowledge necessary. Accent, 3750 Wright Pl., Palo Alto, CA 94306. $149.95.

Imaginator. Entry and professional level 3-D graphics programs for creating, editing, and manipulation of 3-D images. Townsend Microware, 921 Water St., Box 1200, Port Townsend, WA 98368. *Imaginator I*, $79; *Imaginator II*, $119.

LPS II. Superb hi-res graphics drawing system with light pen. Draw freehand or use circles and lines to create geometric shapes. Fill routine with colors and patterns; fun animation demo; programmable Pentrak driver. Gibson, 23192-D Verdugo Dr., Laguna Hills, CA 92653. $349. *10/82.*

Zoom Grafix. Holle. Graphics printing utility allows display of picture on screen prior to print; prints out selected portion at any size. Phoenix, 64 Lake Zurich Dr., Lake Zurich, IL 60047. $39.95. *2/82.*

Home

The Accountant. Forman. Simple-to-use double-entry finance system features seven integrated files and a set of automatic transactions. A sleeper just beginning to get wider distribution. Decision Support, 1438 Ironwood Dr., McLean, VA 22101. $129.95. *1/82.*

The Beer Game. Quiz game and database of every single brand of beer available in the United States. Records brands you know, then compares to grand total list. An obvious labor of love, and the price is right. Paul's Electric Computer, Box 74157, Los Angeles, CA 90004. $4.

Bowling Data System. Data Dynamics. Two-disk record keeping and report preparation program for infinite number of leagues, up to 40 teams. Weekly recap, season average, more. Rainbow Computing, 9719 Reseda Blvd., Northridge, CA 91324. $149.95.

Chequemate. Home finance package that handles checks, charge cards, cash control, automatic tellers, and more. Reports to screen or printer. A bargain. Masterworks, 25834 Narbonne Ave., Lomita, CA 90717. $39.95. *4/82.*

• **Crossword Magic.** Crossword puzzle maker. Choose subject, words, and clues; program automatically connects words. Play on-screen or make printout. L & S Computerware, 1589 Fraser Dr., Sunnyvale, CA 94087. $49.95.

Dow Jones Market Analyzer (formerly *RTR Market Analyzer*). Automatically collects, stores, and updates historical and daily market quotes. Provides technical analysis and plots eighteen different types of charts. Dow Jones Software, Box 300, Princeton, NJ 08540. $350.

Electric Duet. Lutus. Two-voice music without hardware. A bit involved, but superb sound quality. Insoft, 10175 S.W. Barbur Blvd., #202-B, Portland, OR 97219. $29.95. *7/12.*

Family Roots. Professional genealogy database with unlimited records capability. Unprotected; works with 80-column and lower case. Extensive documentation. Quinsept, Box 216, Lexington, MA 02173. $185.

File-Fax. Simple general-purpose DBMS with 8-level sort and report generator. TMQ, 82 Fox Hill Dr., Buffalo Grove, IL 60090. $175.

Home Accountant. Schoenburg. Thorough, powerful home finance program. Monitors live checking accounts against a common budget, plus credit cards and cash; one-step record or transfer of funds. Continental, 11223 S. Hindry Ave., Los Angeles, CA 90045. $74.95. *4/82.*

Hoss. Calhoun. Systematic Thoroughbred handicapping with user modeling coefficients. Five systems and handbook. Tout, 360 S. Gordon St., Pomona, CA 91766. $89.

Know Your Apple IIe. Tutorial program with everything you wanted to know about the soul of your new machine. Muse, 347 N. Charles St., Baltimore, MD 21201. $24.95.

Micro Cookbook. Recipe management system allows entry and modification; selection of recipes by common ingredients, name, or classification. Calorie and nutrition guide. Virtual Combinatics, Box 755, Rockport, MA 01966. $40.

Money Street. Payne. Does accounting, collects data, and balances unlimited number of checkbooks. 100 user-defined categories, 13 reports. Computer Tax Service, Box 4845, Incline Village, NV 89450. $99.95.

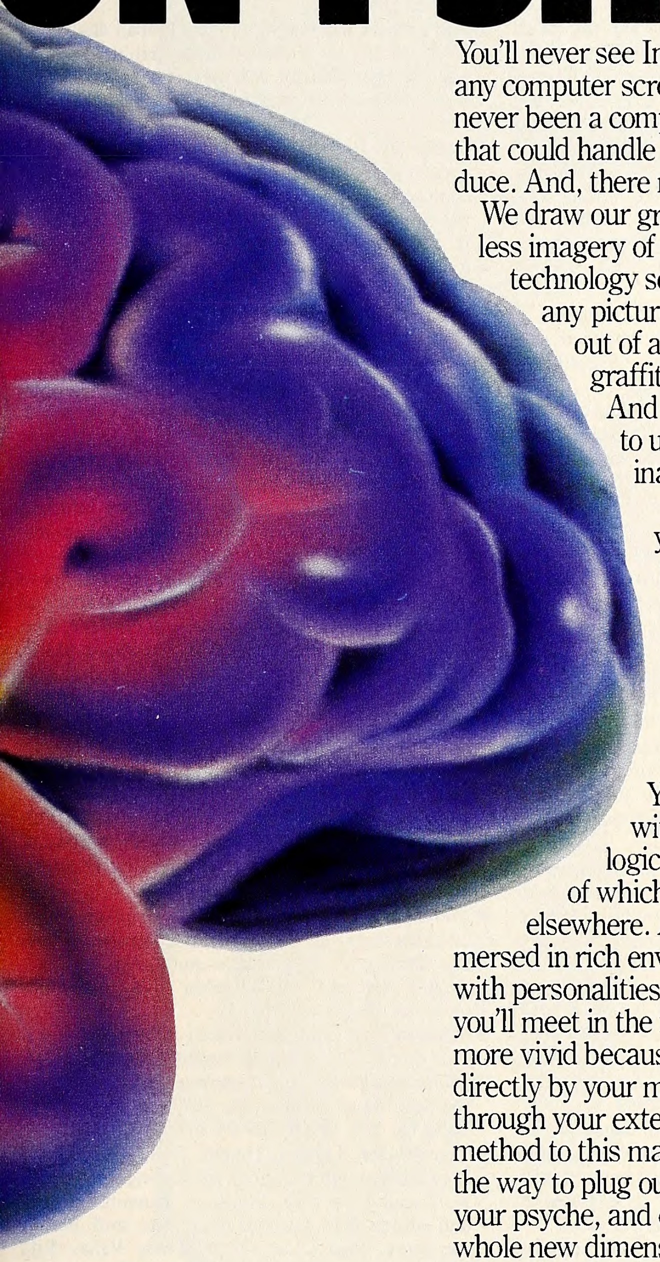
PDQ. Alternative to complex database programs. Scans for record, performs two-word search, displays, and prints out. Saves up to 114,500 characters in 4 files. Howard W. Sams, 4300 W. 2nd St., Indianapolis, IN 46268. $59.95.

Personal Finance Manager. Gold, Software Dimensions. Handles 200 entries a month from 14 separate accounts. Search-sort-enter routine. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $75. *11/81.*

Pick That Tune. Swearingen. Just like the famous show of almost the same name. Players select number of notes they think they'll need to I.D. any melody in Pop, Country/Western, Children, and TV categories. 16 variations, 1 to 10 players. Additional categories purchased separately. Swearingen Software, 6312 W. Little York, #197, Houston, TX 77088. $29.95.

Professional Tax Plan. Problem-solving program for what-if tax situations allowing examination of five tax strategies or projections of up to five years. Automatic income averaging and ten-year averaging. Aardvark Software, 783 N. Water St., Milwaukee, WI 53202. $350.

RAPHICS WHERE ON'T SHINE.



You'll never see Infocom's graphics on any computer screen. Because there's never been a computer built by man that could handle the images we produce. And, there never will be.

We draw our graphics from the limitless imagery of your imagination—a technology so powerful, it makes any picture that's ever come out of a screen look like graffiti by comparison.

And nobody knows how to unleash your imagination like Infocom.

Through our prose, your imagination makes you part of our stories, in control of what you do and where you go—yet unable to predict or control the course of events. You're confronted with situations and logical puzzles the like of which you won't find elsewhere. And you're immersed in rich environments alive with personalities as real as any you'll meet in the flesh—yet all the more vivid because they're perceived directly by your mind's eye, not through your external senses. The method to this magic? We've found the way to plug our prose right into your psyche, and catapult you into a whole new dimension.

Take some tough critics' words about our words. SOFTALK, for example, called ZORK® III's prose "far more graphic than any depiction yet achieved by an adventure with graphics." And the NEW YORK TIMES saw fit to print that our DEADLINE™ is "an amazing feat of programming." Even a journal as video-oriented as ELECTRONIC GAMES found Infocom prose to be such an eye-opener they named one of our games their Best Adventure of 1983.

Better still, bring an Infocom game home with you. Discover firsthand why thousands upon thousands of discriminating game players keep turning everything we write into instantaneous bestsellers.

Step up to Infocom. All words. No graffiti. The secret reaches of your mind are beckoning. A whole new dimension is in there waiting for you.

INFOCOM™

The next dimension.

Infocom, Inc., 55 Wheeler St., Cambridge, MA 02138

For your: Apple II, Atari, Commodore 64, CP/M 8,* DEC Rainbow, DEC RT-11, IBM, NEC APC, NEC PC-8000, Osborne 1, TI Professional, TRS-80 Model I, TRS-80 Model III.

Stock Portfolio System. Tracks investments, gives record-keeping reports and timing notices, stores quotes for historical recall. Smith Micro Software, Box 604, Sunset Beach, CA 90742. $185.

The Tax Advantage. General tax preparer program for Form 1040 and related schedules. Computes tax and itemizes at each line; prints out. Modifiable tax tables. Continental, 11223 S. Hindry Ave., Los Angeles, CA 90045. $59.95.

Tax Manager. Helps prepare federal returns and print schedules. Micro Lab, 2310 Skokie Valley Rd., Highland Park, IL 60035. $150.

Tax Mini-Miser. Sunrise. Tax-planning package computes six tax strategies over one year or one strategy up to six years. Starsoft, 4984 El Camino Real, #125, Los Altos, CA 94022. $295.

Tax Preparer. Record-keeping program with wide variety of federal tax forms and schedules; creates itemized lists. Yearly updates. Howard Software, 8008 Girard Ave., #310, La Jolla, CA 92037. $99.

Home-Arcade

A.E. Wada, Horai. Blasting away like mad in 3-D. Time the release and detonation of missiles and repel the next wave. Innovative graphics, new firing technique, and fugues to boot. Broderbund, 1938 4th St., San Rafael, CA 94901. $29.95. *2/83.*

● **Alien Rain (Apple Galaxian).** Suzuki. Monsters in this classic seem to take it personally when you gun down one of their kind. Broderbund, 1938 4th St., San Rafael, CA 94901. $29.95. *9/81.*

● **Apple Panic.** Serki. Rid a five-story building of crawling apples and butterflies by running up and down connecting ladders, digging traps, then covering critters before they devour you. Extremely addictive, excellent hi-res play. Broderbund, 1938 4th St., San Rafael, CA 94901. $29.95. *9/91.*

The Arcade Machine. Jochumson, Carlston. Step-by-step arcade game designer—shapes, scoring, sound, and titles. Begin with variations on five games included, then on to your own. Broderbund, 1938 4th St., San Rafael, CA 94901. $59.95. *11/82.*

Aztec. Stephenson. Graphic fantasy arcade with animation throughout. DataMost, 8943 Fullbright Ave., Chatsworth, CA 91311. $39.95. *1/83.*

Bandits. Ngo. Fight off waves of multiple menaces intent on killing you and stealing your fruit supply. Delirious nonstop action, animated to the teeth. Sirius, 10364 Rockingham Dr., Sacramento, CA 95827. $34.95. *7/82.*

Beagle Bag. Kersey. Twenty games and miscellany, written in Basic and unprotected. Great humor, good two-player games. Manual is worth the price of admission. Beagle Bros, 4315 Sierra Vista, San Diego, CA 92103. $29.50. *1/83.*

Bolo. Micro version of sci-fi fantasy. Huge maze where you don't eat anything. Drive around in tank and destroy enemy bases as you're dogged by intelligent assassin tanks. Much depth, many months' fun. Top class. Synergistic, 830 N. Riverside Dr., #201, Renton, WA 98055. $34.95. *2/83.*

Bug Attack. Nitchals. Sing along with dagger-wielding ants, blue worms, swarming Medflies, a millipede, the *1812* Overture, lots of bright colors, terrific hi-res animation, and a bouncy style. Cavalier, Box 2032, Del Mar, CA 92014. $29.95. *11/81.*

Cannonball Blitz. Lubeck. In the cold light of *dawn*, you must find the *key* to victory, no matter how in*congr*uous. Sierra On-Line, Sierra On-Line Building, Coarsegold, CA 93614. $34.95. *7/82.*

Canyon Climber. Mountford. Scale the levels and ladders while avoiding arrows, gorges, and hi-res sheep (no cows). Datasoft, 19519 Business Center Dr., Northridge, CA 91324. $29.95.

● **Choplifter.** Gorlin. Fly your chopper to rescue 64 hostages, avoiding interceptor jets, homing mines, and tanks. Challenging, realistic, and playful. Stunning graphics. Broderbund, 1938 4th St., San Rafael, CA 94901. $34.95. *7/82.*

Crime Wave. Your beat: the city. Bank robbers strike; can you catch them? Metropolitan chase-'em-up on city streets or at the scene of the crime. Penguin, 830 4th Ave., Geneva, IL 60134. $19.95. *4/83.*

Crisis Mountain. Schroeder. Run, crawl, walk, and leap through mountain maze fraught with rolling rocks, geysers, and chasms; defuse nuclear devices. Synergistic, 830 N. Riverside Dr., #201, Renton, WA 98055. $34.95. *10/82.*

● **Crossfire.** Sullivan. Aliens come at you from four directions on a grid laid out like city blocks. Strategy and intense concentration required. Superb, smooth animation of a dozen pieces simultaneously. One of the great ones. Sierra On-Line, Sierra On-Line Building, Coarsegold, CA 93614. $29.95. *1/82.*

David's Midnight Magic. Snider. Pinball challenger to *Raster Blaster*. Excellent hi-res graphics and animation. Provision for earning extra balls. Broderbund, 1938 4th St., San Rafael, CA 94901. $34.95. *2/82.*

Dogfight II. Revved-up version allows one to eight players to fly jets into increasingly difficult levels of combat, singly, in teams, and against the computer. Micro Lab, 2310 Skokie Valley Rd., Highland Park, IL 60035. $29.95. *1/81.*

● **Epoch.** Miller. Superbly stylized animation enhances this filmic shoot-'em-up. Tremendous sense of being in space; neat classical music and dramatic time warp sequence. Sirius, 10364 Rockingham Dr., Sacramento, CA 95827. $34.95. *10/81.*

Frogger. Lubeck. Not even close. Sierra On-Line, Sierra On-Line Building, Coarsegold, CA 93614. $34.95. *12/82.*

● **Gorgon.** Nasir. Fly over planet shooting and dodging invaders and saving kidnapped inhabitants. Outstanding hi-res graphics, challenging refueling sequence. Sirius, 10364 Rockingham Dr., Sacramento, CA 95827. $39.95 *8/81.*

Jawbreaker 2. Bueche. No relation or resemblance to *Jawbreaker 1* or Bueche's first. Very playable and addictive. New, fun, and fresh. Sierra On-Line, Sierra On-Line Building, Coarsegold, CA 93614. $34.95. *1/83.*

Lode Runner. Smith. Design your own puzzles, scenes, and setups, in quest to steal Bungeling Empire's gold. Use tightropes, trap doors, and ladders to your advantage. Broderbund, 1938 4th St., San Rafael, CA 94901. $34.95.

Lunar Leeper. Bueche. Silly, enjoyable rescue mission with challenging ship control and unpredictable foes. Sierra On-Line, Sierra On-Line Building, Coarsegold, CA 93614. $29.95. *1/83.*

● **Meteoroids (Asteroids) in Space.** Wallace. Make little asteroids out of big ones, plus occasional hostile alien ships. Hyperspace, autobrake, autofire. Quality Software, 6660 Reseda Blvd., #105, Reseda, CA 91335. $19.95.

● **Microsoft Decathlon** (formerly *Olympic Decathlon*). Smith. Ten standard decathlon events. Hi-res animated athletes, muscle-stirring music; you provide the sweat. Microsoft, 10700 Northup Wy., Bellevue, WA 98004. $29.95. *6/81.*

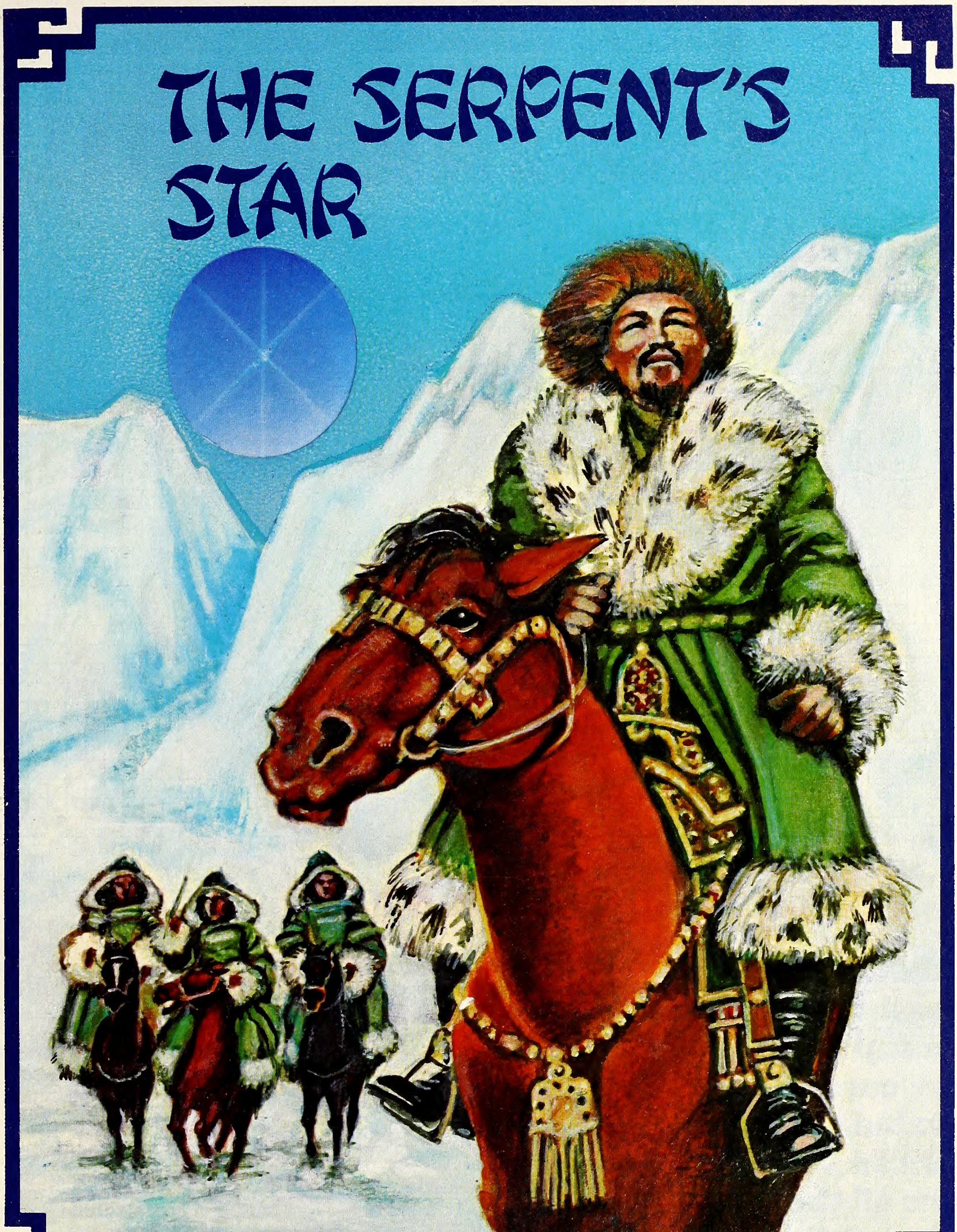
Miner 2049er. Livesay, Hogue. Run, jump, climb, and slide through the mines, reinforcing the groundwork along the way. Elevators, cannons, chutes, and ladders help; mutants don't. Hot stuff, best of the genre. Micro Lab, 2310 Skokie Valley Rd., Highland Park, IL 60035. $39.95. *1/83.*

Pinball Construction Set. Budge. Design and play your own computer pinball games, on-screen, with zero programming. A miracle of rare device. Superior. BudgeCo, 428 Pala Ave., Piedmont, CA 94611. $39.95. *2/83.*

● **Pool 1.5.** Hoffman, St. Germain, Morock. Makes most shots you could on a real table, with the advantages of instant replay and slow motion. Four different games. IDSI, Box 1658, Las Cruces, NM 88004. $34.95. *6/81.*

● **Raster Blaster.** Budge. First realistic pinball game. *Softalk* readers' Most Popular Program of 1981.

SEEK THE GEM OF IMMORTALITY



ANIMATION IN ULTRAvision™!!! Fabulous graphics, full screen animation, sound effects and a wonderfully challenging adventure make this the graphics adventure of choice.

Requires 48K Apple II, DOS 3.3

Apple II is a Registered Trademark of Apple Computer, Inc.

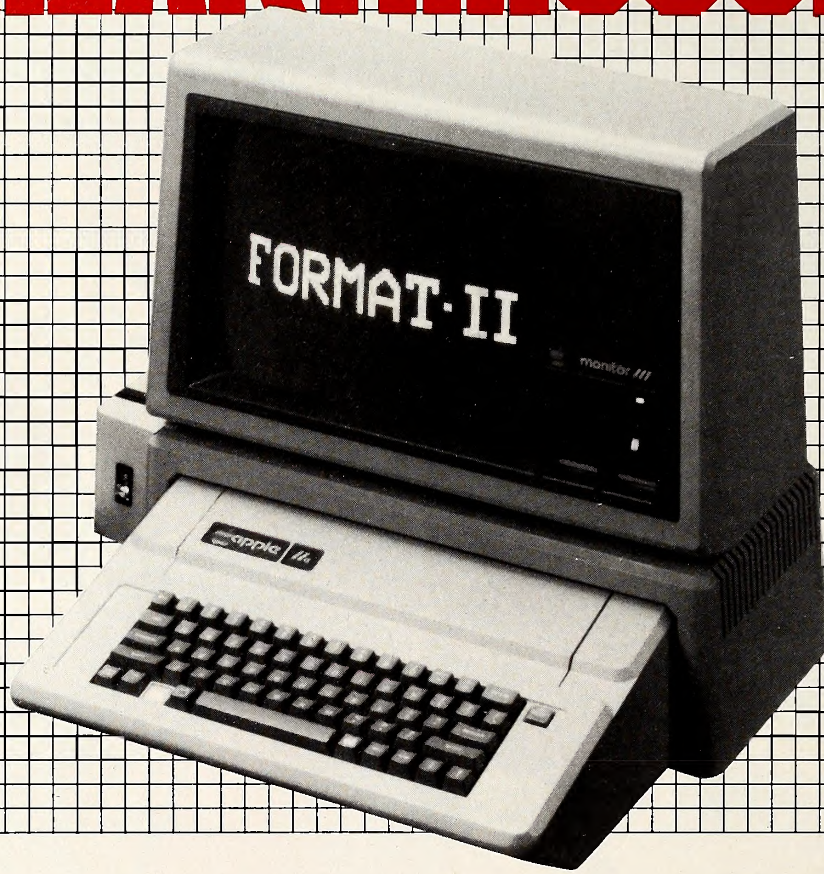
This second in a series of daring and entertaining animated adventures takes Mac Steele into the mysterious and hostile Himilayas of Tibet.

One Disk, Two Sides

Suggested Retail: $39.95

ULTRASOFT INCORPORATED 24001 SE 103rd St., Issaquah, WA 98027 (206) 392 1353

WORD PROCESSING BREAKTHROUGH!



Format-II, Enhanced Version. $150!

No other word processing program can compare in power, ease of use and value. Our new Enhanced Version retains all the features of the original Format-II and adds more:

Copies of reviews available upon request.

Flexibility. Format-II, Enhanced Version recognizes and supports all features of Apple //e, Apple II + and Franklin computers. All popular 80 column cards are supported.

Increased storage. Up to 50 pages of text can be stored on each disk.

Standard Files. Text files are standard DOS 3.3 and can be used with most popular spelling and communications programs.

Hard Disks. Format-II, Enhanced Version is one of the few leading word processors that can be placed onto hard disk drives.

Editor. Format-II, Enhanced Version retains the same single keystroke editor. No other editor is as fast or easy to learn.

What you see is what you get. Text is displayed on the screen exactly as it will print out—paragraphs, underlining, justification, page breaks.

All printers are supported (including proportional space justification).

Mailing List/Database is built in; is fast and extensive.

See your local dealer or order direct from Kensington Microware, Ltd.
919 Third Avenue, New York City 10022.
(212) 486-7707 telex. 236200 KEN UR.

Note: Format II—Enhanced Version supports Apple //e, Apple II +, and Franklin Computers. All popular 80 column cards are supported including Apple Computer's 2 new 80 column text display cards, Videx, Smarterm, Vision 80 and Sup'R'Term, Full View 80, Magnum 80.

Format II and Format II—Enhanced Version are trademarks of Kensington Microware Ltd. Apple and the Apple logo are registered trademarks of Apple Computer, Inc. Franklin/Franklin Computer. Videx/Videx Inc, Smarterm/Advanced Logic Systems, Vision 80/Vista Computer, Full View 80/Bit 3 Corp., Magnum 80/Microtek.

KENSINGTON MICROWARE

BudgeCo, 428 Pala Ave., Piedmont, CA 94611. $29.95. *5/81.*

Repton. Thompson, Kaluzniacki. The *ne plus ultra* of planet-defending, in the *Defender* style, plus. Top flight all the way. Sirius, 10364 Rockingham Dr., Sacramento, CA 95827. $39.95. *1/83.*

Seafox. A good sub-versus-convoy home-arcader. Variety of vessels, bouncing torpedoes, refueling dolphins, and intelligent depth charges. Broderbund, 1938 4th St., San Rafael, CA 94901. $29.95. *11/82.*

Serpentine. Hypnotic snake-chase maze game. Clean action, thrills, hairy escapes. Recommended. Broderbund, 1938 4th St., San Rafael, CA 94901. $34.95. *10/82.*

Sheila. Fitzgerald. Highly adventure-flavored, five-level, real-time maze game with weapons, commands, and spells—acquired with increasing point totals. H.A.L. Labs, 4074 Midland Rd., #23, Riverside, CA 92505. $23. *7/82.*

Snack Attack. Illowsky. Three-maze eat-'em-up; starts at any of five speed levels. Nonfattening. DataMost, 8943 Fullbright Ave., Chatsworth, CA 91311. $29.95. *1/82.*

● **Sneakers.** Turmell. Many-layered shoot-'em-up; one of the best. Stomping sneakers and other creatures require varying techniques. Fun. Sirius, 10364 Rockingham Dr., Sacramento, CA 95827. $29.95. *9/81.*

Star Blazer. Suzuki. Bomb-run game with five levels, minutely exact animation, and style to burn. A joy. Broderbund, 1938 4th St., San Rafael, CA 94901. $31.95. *4/82.*

● **Super Invader.** Hata. Progenitor of home arcades. Still good hi-res, still a challenge. *Softalk* readers' Most Popular Program of 1978-80. Astar Intl., through California Pacific, 1615 5th St., Davis, CA 95616, and Creative Computing, 39 E. Hanover Ave., Morris Plains, NJ 07960. $19.95.

Super Taxman 2. Fitzgerald. Pac up your troubles! Bigger, more complex version of the most perfect extant rendition of a certain arcade game. H.A.L. Labs, 4074 Midland Rd., #23, Riverside, CA 92505. $25. *1/83.*

Thunderbombs. Becklund. You'll need two sets of eyes, hands, and reflexes to survive this one. Your cloudship is under bilateral attack, and it's just you and your bilateral lightning torpedoes. Penguin, 830 4th Ave., Geneva, IL 60134. $19.95.

Wavy Navy. McAuley. Galaxy shooting game brought down to sea level in bright, cartoon-style hi-res. No aliens raining on player's patrol boat; just kamikaze pilots, bombers, and missiles. Shoot them, or it's "P.T. blown home." Good, fun game. Sirius, 10364 Rockingham Dr., Sacramento, CA 95827. $34.95. *2/83.*

● **Wayout.** Exciting 3-D maze that moves in perspective as you play. Map displayed at all times. Lots of angles and Cleptangles. Separate version for IIe. Exquisite motion animation is breakthrough. Sirius, 10364 Rockingham Dr., Sacramento, CA 95827. $39.95. *10/82.*

Home Education

Algebra 1-4. Sets of learning units progressing from algebraic rules and definitions to graphing and inequalities. Individualized teaching styles to fit everyone's needs. Good for adults wanting to overcome math anxiety as well as for schoolkids. Edu-Ware, Box 22222, Agoura, CA 91301. $39.95 each.

Apple Logo. Papert. Custom version (by its inventor) of turtle graphics language. First-rate educational tool. Great kid-friendly documentation. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $175.

Arcademic Skill Builders in Math. *Alien Addition, Alligator Mix, Demolition Division, Dragon Mix, Meteor Multiplication,* and *Minus Mission.* Arcade action blended with addition, subtraction, multiplication, and division problems. Shooting correct answers to problems gets rid of pesky attackers. Choose speed, difficulty levels, game length. Developmental Learning Materials, One DLM Park, Allen, TX 75002. $29.95 each.

Bumble Plot. Colorful musical introduction to concepts of graphing and point plotting. Teaches positive and negative numbers. Learning Co., 4370 Alpine Rd., Portola Valley, CA 94025. $60. *1/83.*

Compu-Read. Set of programs develops speed and retention in reading. Stresses character and word recognition, comprehension. Edu-Ware, Box 22222, Agoura, CA 91301. $29.95.

Computer Literacy: A Hands-On Approach. Luehrmann, Peckham. Textbook, disk, and teacher's guide package introducing students to the world of computers and basic programming. McGraw-Hill, 1221 Ave. of the Americas, New York, NY 10020. $23.97.

Cross Clues. Compete with another player to uncover hidden crossword puzzle words; computer supplies clues. SRA, 155 N. Wacker Dr., Chicago, IL 60606. $35.

CyberLogo. Woodhead. Logo learning package introduces computers, uses imaginary school and playground settings to teach kids language with fun. Includes off-computer activities for reinforcement. By *Wizardry* author. Only Logo for 48K Apples. Cybertronics Intl., 999 Mount Kemble Ave., Morristown, NJ 07960. $99.95.

Delta Drawing. Kids can make colorful drawings by using single-key commands. No special talent needed; this one develops programs that create complex graphics. Spinnaker, 215 1st St., Cambridge, MA 02142. $59.95. *11/82.*

Dragon's Keep. Graphics adventure in which youngsters find and free imprisoned animals. Written for second-grade-level readers; requires the touch of a key, no typing, to execute actions. Encouraging and rewarding. All upbeat. Sunnyside Soft, 5815 E. Parkside, Fresno, CA 93727. $34.95. *2/83.*

Early Games for Young Children. Paulson. Basic training in numbers, letters, Apple keyboard for children ages two to seven with no adult supervision. Has a neat little drawing program. Counterpoint Software, #140, Shelard Plaza North, Minneapolis, MN 55426. $29.95. *11/82.*

Ernie's Quiz. CTW. Four games, four subjects, one disk. Image recognition, counting skills, creativity, and Muppet expertise are introduced with lots of positive feedback. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $50. *2/83.*

Facemaker. DesignWare. Exercises kids' creativity and introduces programlike command sequencing as kids create faces and link them together in animated pattern. Spinnaker Software, 215 First St., Cambridge, MA 02142. $34.95.

First Words. Wilson, Fox. Vocabulary comprehension training program using color-graphics animation and sound to teach fifty basic nouns to children ages nine months to two years. Requires Echo II speech synthesizer. Laureate Learning Systems, 1 Mill St., Burlington, VT 05401. $185.

Game Show. Guess mystery words from clues given by "celebrity" partners—no threat to Liz Montgomery. Fifteen subjects cover vocabulary, history, algebra, and more. Add topics. Computer-Advanced Ideas, 1442A Walnut St., #341M, Berkeley, CA 94709. $39.

Gertrude's Secrets. Gertrude the Goose teaches four-to-nine-year-olds shape and color relationships. Solve logic puzzles, create shapes. The Learning Co., 4370 Alpine Rd., Portola Valley, CA 94025. $75. *2/83.*

Instant Zoo. CTW. Identify animals, test perception and reaction, match and decode words. Word editor lets you create your own word lists. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $50.

Juggles' Rainbow. Pre-reading tots can create colorful pictures by using the keyboard. Learning Co., 4370 Alpine Rd., Portola Valley, CA 94025. $45.

Knowledge Bowl. Home version of that TV college quiz show. Test knowledge of humanities, social sciences, and hard sciences in more than thirty programs. Play alone or compete with companion. Well done. Academic Hallmarks, Box 998, Durango, CO 81301. $27 each.

Krell Logo. Concentrates on underlying principles of Logo; sections on assembly language interfaces and music creation, plus *Alice in Logoland* tutorial. Krell, 1320 Stony Brook Rd., Stony Brook, NY 11790. $89.95. *7/82.*

Magic Crayon. Clark. Keystroke command draws pictures in lo-res. Saves pictures to disk. Option for sound; class rosters can be maintained. C & C Software, 5713 Kentford Circle, Wichita, KS 67220. $35.

• **MasterType.** Zweig. Learn to type by playing a game; simple and ingenious. IIe version teaches new keyboard. Lightning, Box 11725, Palo Alto, CA 94306. $39.95. *4/81.*

Math Blaster. Davidson, Eckert. Elementary-school-level training in four basic math functions. Options to create lessons; several levels of difficulty for various ages. Human cannonball arcade game for each function. Davidson & Associates, 6069 Groveoak Pl., #12, Rancho Palos Verdes, CA 90274. $49.95.

The Medalists Series. Elementary and junior high level puzzles about continents, presidents, and states. Disk are modified and hold fifty students' records. Students buy clues for points according to difficulty, then use them to identify the subject. The *Create* disk lets teachers devise *Medalist* competitions for the topic of their choice. Hartley Courseware, Box 431, Dimondale, MI 48821. $39.95.

Micro Mother Groose. Rhyme-related games with progressive levels, music, animation. Simple enough for tots to operate it. Free poster and stickers, too. Software Productions, 2357 Southway Dr., Box 21341, Columbus, OH 43221. $39.95. *2/83.*

Mix and Match. CTW. Create mixed-up Muppets and teach the Apple about animals. Logic and word-guessing games. Add your own word lists. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $50. *2/83.*

Multiploy. Coletta. From answer base, blast menacing arithmetic problem ships out of the sky. Ranks and scores kept for tracking progress. Reston Publishing, 11480 Sunset Hills Rd., Reston, VA 22090. $19.95.

• **The New Step by Step.** Software and audio tape team up to teach Basic programming painlessly. Graphics, animation, sound effects, and workbook. Superior. Program Design, 11 Idar Ct., Greenwich, CT 06830. $79.95. *7/82.*

Punctuation Skills: Commas. Covers all uses of the comma. **Punctuation Skills: Endmarks.** Covers semicolons, colons, exclamation points, and periods. Milton Bradley, 111 Maple St., Springfield, MA 01105. Each, $49.95.

Report Card. Ringuette. Grading system lets teachers weight different activities. No commands to learn. Sensible, 6619 Perham Dr., W. Bloomfield, MI 48033. $59.95.

Rocky's Boots. Rascally raccoon helps children build logical thinking and computer understanding. Construct machines of logical gates in convolutions of thickening complexity. Music and sound effects add to fun. The Learning Co., 4370 Alpine Rd., Portola Valley, CA 94025. $75. *2/83.*

SAT English I. Designed to help high school students prepare for college entrance exam. Covers verbal half of test; learn by mistakes. Micro Lab, 2310 Skokie Valley Rd., Highland Park, IL 60035. $30. *11/81.*

Snooper Troops. Snyder. Ongoing hi-res mystery series in form of educational games. Highly structured; excellent fourth through eighth-grade educational tool. Fun for adults, too. Spinnaker Software, 215 1st St., Cambridge, MA 02142. $44.95 each. *9/82.*

Spelling Bee Games. Hi-res games strengthen hand-eye coordination, memory, motor skills. Word lists include shapes, animals, more. Edu-Ware, Box 22222, Agoura, CA 91301. $29.95.

Spotlight. CTW. Simple geometry for preteens. Games involve number estimation and angles of reflection. Good and fun. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $50.

Steps to Advanced Reading. Courseware stressing comprehension and speed. Programmed reading lessons and computer stories on four disks; supplemental reading book. Tests and answer sheets included. Creative Curriculum, 15632 Producer Lane, Huntington Beach, CA 92649. $99.

Stickybear. Hefter, Worthington. Animated early education programs. In *Stickybear ABC*, moving pictures with sound represent letters. In *Stickybear Numbers*, groups of moving objects teach numbers and simple arithmetic. Ages three through six. Xerox Education/Weekly Reader, 245 Long Hill Rd., Middletown, CT 06457. $39.95 each.

Story Machine. Helps develop positive attitude toward writing and ability to write correctly. Words come to life when sentence typed is acted out on-screen. Kids five to nine love to type "The tree ran down the street" and see it do so. Spinnaker Software, 215 1st St., Cambridge, MA 02142. $34.95.

Supermath II. Tests students' skills in addition, subtraction, multiplication, and division. Difficulty levels change automatically according to responses. Interacts with this company's *Class Records.* Educational Systems Software, 23720 El Toro Rd., #C, Box E, El Toro, CA 92630. $39.95.

Tic Tac Show. Teaches facts and concepts about the world in general. Solo or double play; add topics. Computer-Advanced Ideas, 1442A Walnut St., #341M, Berkeley, CA 94709. $39.95.

Typing Tutor. Ainsworth, Baker. Four levels of proficiency; individualized drills created with time-response monitoring. Microsoft, 10700 Northup Wy., Bellevue, WA 98004. $24.95.

The Visible Computer: 6502. Hi-res simulation teaches machine language programming by illustrating inside of working 6502 microprocessor. Software Masters, 3330 Hillcroft, #BB, Houston, TX 77057. $49.95.

Vocabulary Skills: Subtext Clues. Develops vocabulary through context, contrast, educated guesses, and examples. **Vocabulary Skills: Prefixes, Suffixes, Roots.** Includes concepts, prefix and suffix tutors, and word building. Milton Bradley, 111 Maple St., Springfield, MA 01105. Each, $49.95.

Word Attack! Davidson, Eckert. Builds vocabulary through multiple-choice quiz, sentence-completion exercises, and arcade game. Nine levels of word difficulty. Davidson, 6069 Groveoak Pl., #2, Rancho Palos Verdes, CA 90274. $49.95.

Wordrace. Timed dictionary game. Pick correct definition out of six choices. Three levels, 2,000 words and definitions. Don't Ask, 2265 Westwood Blvd., #B-150, Los Angeles, CA 90064. $24.95.

Strategy

Bomb Alley. Grigsby, Billings. Detailed re-creation of 1942 Mediterranean naval and air war, including critical supply problems. Full scenario and two short scenarios. Strategic Simulations, 465 Fairchild Dr., #108, Mountain View, CA 94043. $59.95. *3/83.*

Casino. Five hi-res games, Vegas-style: blackjack, baccarat, keno, poker, and roulette. DataMost, 8943 Fullbright Ave., Chatsworth, CA 91311. $39.95. *10/82.*

• **Castle Wolfenstein.** Warner. First game to fuse successfully strategy, home-arcade fantasy. Escape from Nazi stronghold with secret plans. Room layout changes with each new game. Enemy speaks, in German. Muse, 347 N. Charles St., Baltimore, MD 21201. $29.95. *10/81.*

Chess 7.0. Atkin. A loving piece of programming; neither too slow nor too easy. Plays a mean end game. Tops yet. Odesta, 930 Pitner, Evanston, IL 60202. $49.95. *1/83.*

Computer Ambush. Williger. Gutty soldier-to-soldier street fighting in World War II France. Latest version is forty times faster than the original, which was one of best games ever created for Apple, except for slowness. Strategic Simulations, 883 Stierlin Rd., A-200, Mountain View, CA 94043. $59.95.

• **Computer Baseball.** Merro, Avery. Simulates individual player abilities from the teams of thirteen famous World Series. Enter and play teams of your own creation. Strategic Simulations, 465 Fairchild Dr., #108, Mountain View, CA 94043. $39.95. *9/81.*

Computer Bismarck. Lyon, Billings, Cook. Take charge of British ships and aircraft against German U-boats to sink the mighty German battleship. This edition plays faster than the original. Strategic Simulations, 883 Stierlin Rd., A-200, Mountain View, CA 94043. $59.95.

Epidemic! Faber. Fight epidemic caused by virus-bearing meteorites striking Earth in different geographic areas; weapons are interferon, vaccines, radiation treatment, and possibly a nuclear alternative. Different. Strategic Simulations, 465 Fairchild Dr., #108, Mountain View, CA 94043. $34.95. *3/83.*

• **Flight Simulator.** Artwick. Uses aerodynamic equations, airfoil characteristics for realistic takeoff, flight, and landing. Two years on Top Thirty. SubLogic, 713 Edgebrook Dr., Champaign, IL 61820. $33.50.

Flip Out. Huskey. Drop marbles through top of maze, activating traps to free your marbles and trap your opponent's. Sirius, 10364 Rockingham Dr., Sacramento, CA 95827. $29.95. *4/83.*

Germany 1985. Keating. NATO forces tangle with Soviet troops in West Germany in the first act of SSI's *When Superpowers Collide* saga—accompanied by *RDF 1985*, *Norway 1985*, and *Baltic 1985*. Includes rulebook necessary for play of the other acts. Strategic Simulations, 465 Fairchild Dr., #108, Mountain View, CA 94043. $59.95. Others, $34.95. *4/83.*

Gin Rummy. Carpet. Play against computer. Hi-res hand can be arranged. Knocking allowed. Computer plays pretty well. DataMost, 8943 Fullbright Ave., Chatsworth, CA 91311. $29.95. *6/82.*

Go. Erwin. Classic Oriental territory game in hi-res. Surround your opponents before they surround you. Play in solitaire or bihuman mode. Hayden, 600 Suffolk St., Lowell, MA 01853. $34.95.

Hi-Res Computer Golf. Aronoff. A masterpiece; requires judgment, strategy, and visual acuity. One of the few computer sports simulations that requires dexterity. Avant-Garde, Box 30160, Eugene, OR 97403. $29.95. *2/82.*

• **Microgammon II.** Program for play, practice, improvement of backgammon skills. Pretty good competition. Softape, 5547 Satsuma Ave., North Hollywood, CA 91601. $19.95. *2/81.*

Millionaire. Executive stock-market-simulation game played via reports, graphs, options, and volume indicators. Blue Chip, 19537 Wells Dr., Tarzana, CA 91356. $79.95.

Pro Poker. Allen. Hi-res 8-handed poker tutorial between just you and your Apple in kibitz mode. Plays 300 hands per hour; California poker club rules. Quality, 6660 Reseda Blvd., #105, Reseda, CA 91335. $39.95.

Rendezvous. Huntress. Space shuttle simulation in 3-D, created by senior scientist at JPL. Orbit earth, match orbit, and dock with space station. Authentic, demanding. Edu-Ware, Box 22222, Agoura, CA 91301. $39.95. *7/82.*

The Road to Gettysburg. Murray. Replace generals Lee and Meade, calling the shots in this famous Civil War re-creation. Strategic Simulations, 883 Stierlin Rd., A-200, Mountain View, CA 94043. $59.95.

• **RobotWar.** Warner. Strategy game with battling robots is teaching device for programming. Muse, 347 N. Charles St., Baltimore, MD 21201. $39.95. *1/81.*

• **Sargon II.** Spracklen, Spracklen. Computer chess game with seven levels of play. Hayden, 50 Essex St., Rochelle Park, NJ 07662. $34.95.

Utility

Amper Magic. Nacon. Attaches machine-language routines to Applesoft programs. No knowledge of machine language necessary. Anthro-Digital, 103 Bartlett Ave., Pittsfield, MA 01201. $75.

Apple-Cillin. Hardware diagnostic tests for all RAM and ROM, plug-in cards, cp registers, disks; nine video test patterns. XPS, 323 York Rd., Carlisle, PA 17013. $49.95.

Apple Mechanic. Kersey. Multiple utility disk with shape editor, custom typefonts, byte rewriter, and tricks to facilitate music, text, and hi-res generation. Beagle Bros, 4315 Sierra Vista, San Diego, CA 92103. $29.50. *9/82.*

Apple Pascal. Structured operating system featuring enhancements of color graphics, sound generation, and Apple's I/O features. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $495.

Audex. Collection of utilities to create, edit, and play back sounds; in Basic and assembly language. Sirius, 10364 Rockingham Dr., Sacramento, CA 95827. $29.95.

Bag of Tricks. Worth, Lechner. Four utility programs for dumping and examining raw tracks, sector editing, reformatting tracks, and repairing damaged catalogs. Indispensable. Quality Software, 6660 Reseda Blvd., #105, Reseda, CA 91335. $39.95.

Bug Byter. Screen-oriented mnemonic debugging tool with resident assembler and disassembler. Displays contents of accumulator, X and Y registers. Computer-Advanced Ideas, 1442A Walnut St., #431, Berkeley, CA 94709. $47.50.

DOS Boss. Kersey, Cassidy. Utility to change DOS commands; customize catalog. Good ideas and witty presentation. Beagle Bros, 4315 Sierra Vista, San Diego, CA 92103. $24. *10/81.*

DOS Tool Kit. Excellent utility package; Apple II assembler-editor system and Applesoft toolkit. Edit, assemble machine language programs; write, edit Basic programs. Simplifies graphics, includes character generator. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $75. *10/81.*

Einstein Compiler. Goodrow, Einstein. Translates Applesoft programs into machine language for runtime up to 20 times faster. Supports all graphics modes, defined functions, and DOS commands. Einstein, 11340 W. Olympic Blvd., Los Angeles, CA 90064. $119.95.

Flex Text. Simonsen. Adds graphics to text and vice versa; prints variable-width text with no hardware. Beagle Bros, 4315 Sierra Vista, San Diego, CA 92103. $29.50.

Global Program Line Editor. Enhanced version of *Program Line Editor* with programmable cursor and listing control. Edit line by line or by range of lines and search for strings. Synergistic, 830 N. Riverside Dr., #201, Renton, WA 98055. $60.

Hands-On Basic Programming. Kamins, Bennett. Workbook and disk teach Basic programming, other basic knowledge of the Apple. User frustration deliberately omitted from this tutorial. Edu-Ware, Box 22222, Agoura, CA 91301. $79.

IDS. An integrated development system allowing screen form I/O techniques, more convenient access to disk files, and print-line formatting. R. R. Michaels, Box 565, Leesburg, VA 22075. $85.

Lisa 2.5. Hyde. Longtime popular assembler with extended mnemonics and more than thirty opcodes. Sierra On-Line, Sierra On-Line Building, Coarsegold, CA 93614. $79.95.

Merlin. Does assembly language programming with dozen editing commands and 28 pseudo-ops. Southwestern Data, 10761-E Woodside Ave., Santee, CA 92071. $64.95.

MUD. *Master Utility Disk* for aid in program maintenance and repairs of fatal errors. Different versions for II, II Plus, and IIe. WM Enterprises, 9348 Santa Monica Blvd., #101, Beverly Hills, CA 90210. $69.95.

ORCA/M. Object relocatable code assembler for micros. Macro language features; linker produces executable binary files. Co-resident screen editor and system disk sector editor. Hayden, 50 Essex St., Rochelle Park, NJ 07662. Introductory, $99.95.

ProntoDOS. Weishaar. High-speed disk utility cuts about two-thirds of the time off bload and save functions. Compatible with all DOS commands; frees up to 15 extra sectors per disk. Beagle Bros, 4315 Sierra Vista, San Diego, CA 92103. $29.50.

Sphinx. Software giving single-pass encryption beyond 10 to the 400th power. Crane Hill, Box 273, Gonzalez, FL 32560. $37.50.

• **Super Disk Copy III.** Hartley. Easy-to-use menu-driven software utility; correct file sizes, undelete, free DOS tracks, more. Sensible, 6619 Perham Dr., W. Bloomfield, MI 48033. $30. *10/81.*

TASC. Peak, Howard. Applesoft compiler; user controls locations of three memory compartments. Microsoft, 10700 Northup Wy., Bellevue, WA 98004. $150. *9/81.*

Type Faces. Printing enhancement tool for dot-matrix printers; fifteen hi-res character fonts available. Alpha, 12 New England Executive Park, Burlington, MA 01803. $125.

Utility City. Kersey. Twenty-one utilities on one disk. Beagle Bros, 4315 Sierra Vista, San Diego, CA 92103. $29.50.

Word Processing

Apple Writer II and IIe. Includes WPL, word processing language. Additional functions menu; continuing features and functions menu; continuous readout of character count and length. *IIe* has shift, shift-lock, and tab, four-arrow cursor control, and delete key; data files compatible with *II*. Apple, 20525 Mariani Ave., Cupertino, CA 95014. *II*, $150; *IIe*, $195.

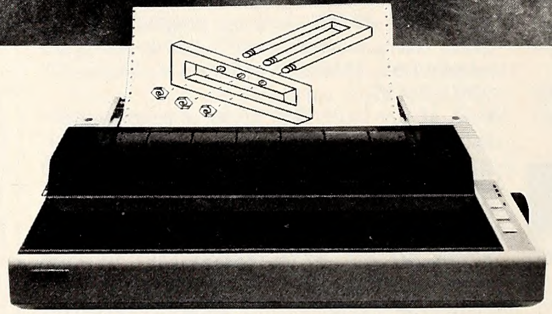
Bank Street Writer. Kusmiak, Bank Street College of Education. Designed for use by whole family. Universal search and replace, word wrap are standard. U&lc without hardware. On-disk tutorial. Takes advantage of memory, keyboard on IIe, if you have one. Broderbund, 1938 4th St., San Rafael, CA 94901. $69.95. *2/83.*

Executive Secretary. Editing, printing, and form letters, plus mail merge and electronic mail system. SofSys, 4306 Upton Ave. S., Minneapolis, MN 55410. $250.

Magic Window II. 40, 70 (in hi-res), or 80 columns in this expanded version. Compatible with Pascal 80-column. With user-tailored, fast menu; underlining; global search and replace. IIe version uses all 64K, more if you have it. Artsci, 5547 Satsuma Ave., North Hollywood, CA 91601. $149.95.

Pie Writer. Business processor allows 9,999 pages. Word deletion, auto indent, spooling, and type-ahead buffer. Hayden, 50 Essex St., Rochelle Park, NJ 07662. $149.95.

ScreenWriter II. Kidwell, Schmoyer. No extra hardware for u&lc, 70-column display, printer spooling. Edits Basic, text, and binary files; complete search



ACTUAL APPLE II PRINTOUT USING GRAPPLER AND EPSON MX100

**COMPLETELY REDESIGNED.
NOW, THE GRAPPLER +.**

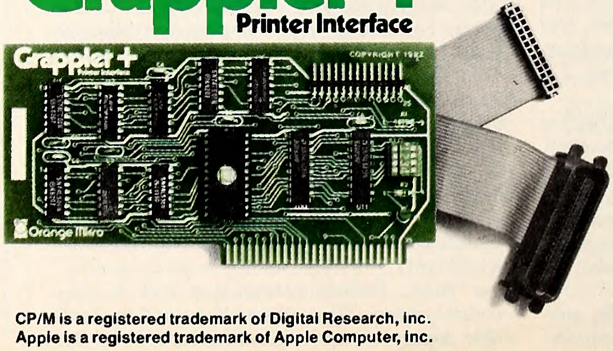
The original Grappler was the first graphics interface to give you hi-res screen dumps from your keyboard. The new Grappler + with *Dual Hi-Res Graphics* adds flexibility with a side-by-side graphics printout of page 1 and page 2.

The Grappler + can now be used with the Apple® Dot Matrix, the Okidata 84, and is Apple III compatible.* In addition, the IDS Grappler + is currently available with color capability, including color graphics screen dumps.

UP TO 64K BUFFER OPTION
An optional Bufferboard can now be added to all existing Grappler and Grappler + interfaces. See your Apple Dealer for details.

*Requires additional software driver.
**Requires graphics upgrade.

With The Grappler+™ Printer Interface



CP/M is a registered trademark of Digital Research, Inc.
Apple is a registered trademark of Apple Computer, Inc.

NOW AVAILABLE THE BUFFERED GRAPPLER +

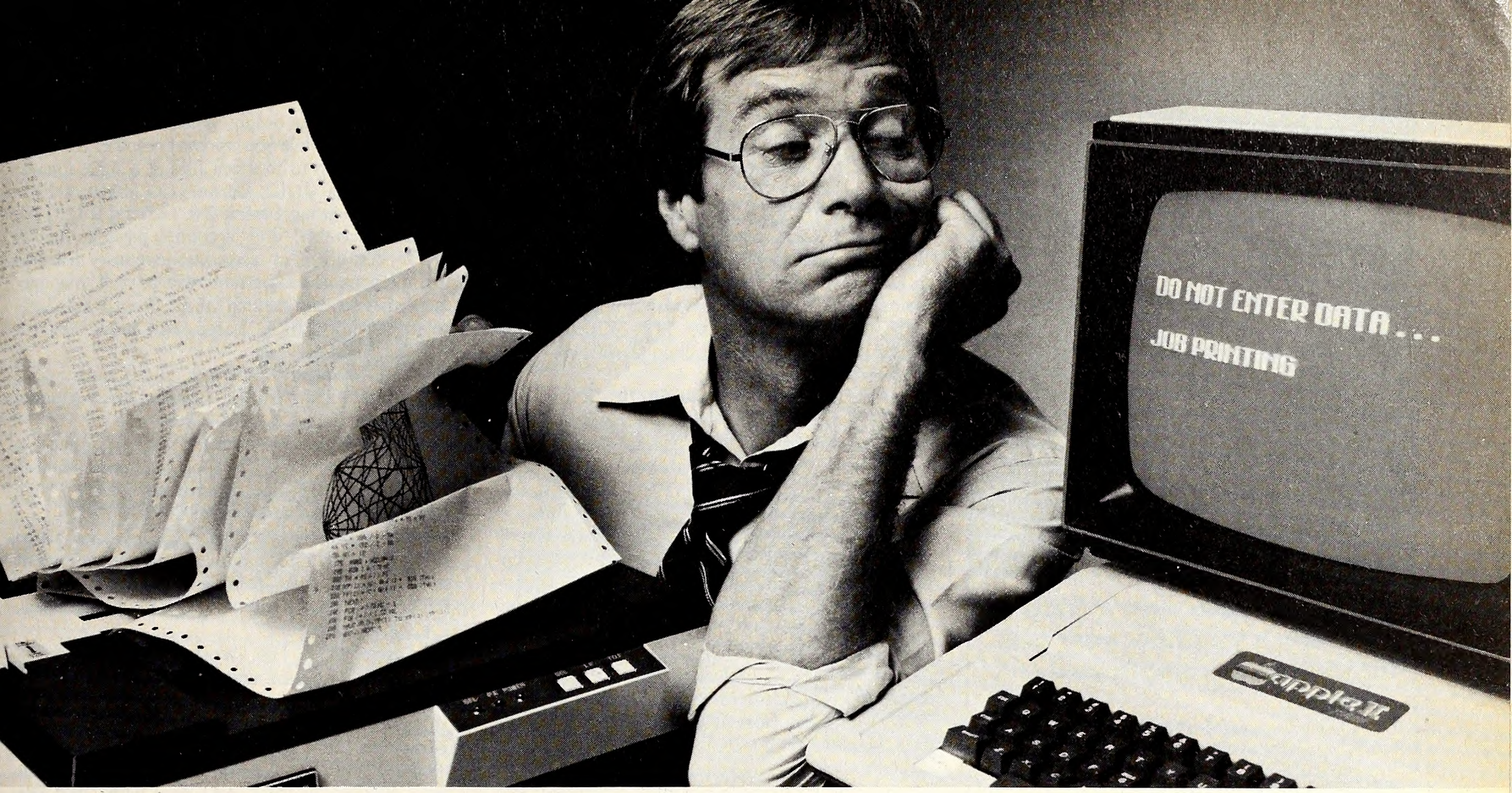
The best of both worlds...the Buffered Grappler +

All of the popular Grappler + features with the time-saving benefits of a printer buffer.

- 16K of Buffer
- Expandable to 64K
- Interfaces with all popular dot matrix printers

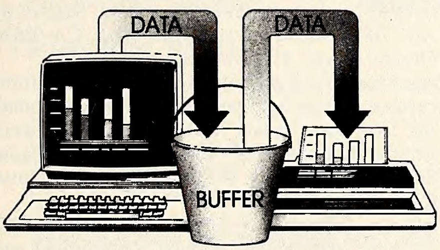
Make the most of your Apple and printer, with the Grappler + or the Buffered Grappler +.

Orange Micro inc.
1400 N. Lakeview Ave.,
Anaheim, CA 92807 U.S.A.
(714) 779-2772 Telex: 183511 CSMA
Foreign Dealer Inquiries Welcome



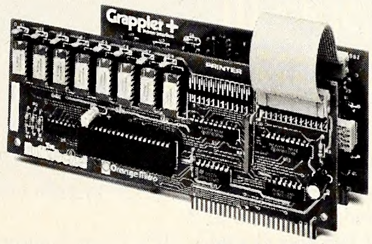
If your printer uses your Apple® more than you do, you need The Bufferboard.™

If your Apple is locked into the "PRINT" mode so much that you've taken up solitaire to kill the boredom, you need a buffer. And if your computer is the Apple II or III, the only buffer for you is The Bufferboard. Expandable to 64K of storage, The Bufferboard stores an instantaneous **bucketful** of print data from your computer. Then it feeds the data to your printer at its own printing rate. Your Apple is set free from driving your printer and is ready for more data from you.



Take your existing interface—and buffer it!

Only The Bufferboard has a simple Interface-Docking System. No bulky boxes or expensive power supplies are needed because The Bufferboard fits right into your Apple—and **docks** onto your existing printer interface. The result is convenient



and economical buffering of most popular printer interfaces, including the Grappler+™ interface, Epson interface, and Apple printer interface. Thirty seconds and a single hook-up are all you need to end the printer waiting game forever.

Up to 20 letter-size pages stored at a time.

The Bufferboard comes standard with 16K, and is expandable to 32K or 64K of buffering capacity with the addition of memory chips. This "bucket" will hold up to 20 pages of a print job, allowing you freedom to use your Apple.

The Bufferboard—designed exclusively for the Apple Computer.

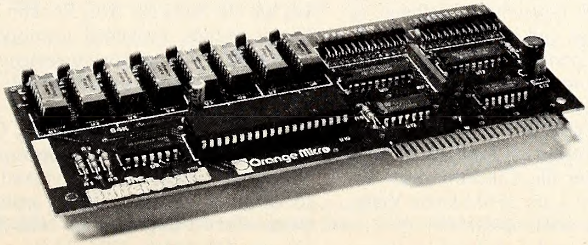
Specifications:
• Versions for Grappler+ interface, Epson interface, Apple interface, and other popular printer interfaces • 16K buffer standard • Upgradeable to 32K or 64K • Automatic memory configuration • Automatic self test • Includes interface docking cable.

The Bufferboard is made by Orange Micro, Inc.; the same people who brought you the popular Grappler+ printer interface. Both the Grappler+ and The Bufferboard are now available at your local Apple dealer.

Apple is a registered trademark of Apple, Inc.
Epson is a registered trademark of Epson America, Inc.

Orange Micro inc.

1400 N. Lakeview, Anaheim, CA 92807
U.S.A. (714) 779-2772
TELEX: TX 183511 CSMA



and replace. IIe version uses 80 columns, u&lc, shift key, and all available memory. Sierra On-Line, Sierra On-Line Building, Coarsegold, CA 93614. $129.95. *1/83*.

Sensible Speller. Spell-checking program sports listable 85,000 words, extensible up to 110,000 words. Recognizes contractions, gives word counts, word incidence, number of unique words. Clear documentation and simplicity of operation. Works with many word processors' files. Best of breed. Sensible, 6619 Perham Dr., W. Bloomfield, MI 48033. $125. *1/82*.

Super-Text Home/Office (40/56/70). Zaron. Get 40, 56, or 70 columns without hardware. Design character sets. Basics of text editing. Character-oriented, floating-cursor edit with add, change, print, and preview modes. Muse, 347 N. Charles St., Baltimore, MD 21201. $99.

Super-Text Professional (40/80). Automatic 80-column, u&lc on equipped IIe; with appropriate equipment on II Plus. On-screen formatting and help reference guides. Muse, 347 N. Charles St., Baltimore, MD 21201. $99.

Videx Preboot Apple Writer. 80-column display for *AppleWriter II* with u&lc input from keyboard. Enhancer II and Videoterm compatibility. Videx, 897 N.W. Grant St., Corvallis, OR 97330. $19.

Word Handler II. Elekman. Simple program with straightforward documentation. Allows folded paper printout for two-sided printing. 80-column with the IIe. Silicon Valley Systems, 1625 El Camino Real, #4, Belmont, CA 94002. $199. *11/82*.

WordStar. Screen-oriented, integrated word processing system in CP/M. Z-80. MicroPro, 33 San Pablo Ave., San Rafael, CA 94903. $495.

Zardax. Philips. Highly recommended. Single program includes supersimple use of powerful word processing features. Considerable extras including communication by modem. Good 80-column facility with board, automatic in IIe version. Computer Solutions, Box 397, Mount Gravatt, Queensland, Australia. In the U.S.: Action-Research Northwest, 11442 Marine View Dr. S.W., Seattle, WA 98146. $295. *Zip-Comm* modem program. $80. *11/82*.

Apple III

Access III. Communications program for time sharing and standalone tasks; gives access to remote information services, minis, and mainframes. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $150.

Apple Business Basic. High-level structured programming language. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $125.

Apple III Business Graphics. BPS. General-purpose graphics program draws line graphs, bar graphs in three formats, overlays, and pie charts in 16 colors. Continuous or discrete data; curve-fitting capabilities. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $175.

Apple III Pascal. Program preparer with editor, compiler, disassembler, linker, filer, system library. Features cursor control, text modeling, formatting. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $250.

Apple Writer III. Lutus. Uses WPL (word processing language) to automate text manipulation and document creation. Adjusts print format during printing; translates from typewriter shorthand to English or other language and back again. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $225.

Catalyst. Allows boot from hard disk; transfers all programs to ProFile. Quark Engineering, 1433 Williams, #1102, Denver, CO 80218. $149.

Data Manager III. Expansion of *Data Factory* allowing 32,000 records per file. Custom screen display and printing. Micro Lab, 2310 Skokie Valley Rd., Highland Park, IL 60035. $750.

Discourse. Spooler to be used with hard or floppy disk drive. Printer output goes to disk, then from disk to printer while you use the computer for other tasks. Holds up to fourteen files at a time. Quark, 1433 Williams, #1102, Denver, CO 80218. $125.

Hardisk Accounting Series, 2.0. General ledger, accounts receivable, and accounts payable handle 32,776 customers or accounts; inventory features five methods of evaluation. Also payroll, management analysis, and mailing labels. Great Plains Software, 123 N. 15th St., Fargo, ND 58102. $395 to $595 per module.

Mail List Manager. Generates, stores, sorts, edits, and prints mailing list files. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $150.

Micro/Terminal. Gives access to any in-house or remote database; set up and log only once. Built-in editor or edit off-line. Microcom, 1400A Providence Hwy., Norwood, MA 02062. $99.95.

PFS:File (formerly *Personal Filing System*). Page. Form-oriented information management system stores and retrieves up to 32,000 entries. Software Publishing, 1901 Landings Dr., Mountain View, CA 94043. $175.

PFS:Graph. Chin, Hill. Works alone or interfaces with PFS databases and *VisiCalc* files. Produces bar, line, and pie charts, merging data from several sources. Software Publishing, 1901 Landings Dr., Mountain View, CA 94043. $175.

PFS:Report. Page. Generates reports; sorts, calculates, and manipulates data filed with *PFS:File*. Software Publishing, 1901 Landings Dr., Mountain View, CA 94043. $125.

Pick That Tune. Swearingen. Up to 10 players bid on least notes to I.D. any melody in Pop, Country/Western, Children, and TV categories. 16 variations. Additional categories available separately. Swearingen Software, 6312 W. Little York, #197, Houston, TX 77088. $29.95.

Quick File III. Personal index card or filing system. 15 fields; file as long as disk allows; can be put on ProFile. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $100.

Senior Analyst III. Corporate planning tool for developing budgets, forecasts, financial models without programming; automatically formats reports and documents and assumptions in plain English. Apple, 20525 Mariani Ave., Cupertino, CA 95014. $300. *4/83*.

State of the Art General Ledger and **Business Modules.** Standalone interfaceable modules for 12 accounting periods. General ledger can handle 470 accounts, 100 transactions before updating files. Modules for budget and financial reporting, accounts receivable/payable, and inventory control. State of the Art, 3183A Airway Ave., Costa Mesa, CA 92626. General ledger, $595; modules, $495.

VersaForm. Landau. State-of-the-art business forms processor. Does invoicing, purchasing orders, mailing lists, client billing. Powerful, complex, worth getting to know. Hard-disk-compatible. Applied Software Technology, 14128 Capri Dr., Los Gatos, CA 95030. $495. *8/82*.

VisiCalc Advanced Version. For corporatewide modeling applications; develop sophisticated templates to be filled in by novice users. On-screen help, IRR and calendar functions, macro facility, variable column widths, locked cell values, and hidden cell contents. VisiCorp, 2895 Zanker Rd., San Jose, CA 95134. $400.

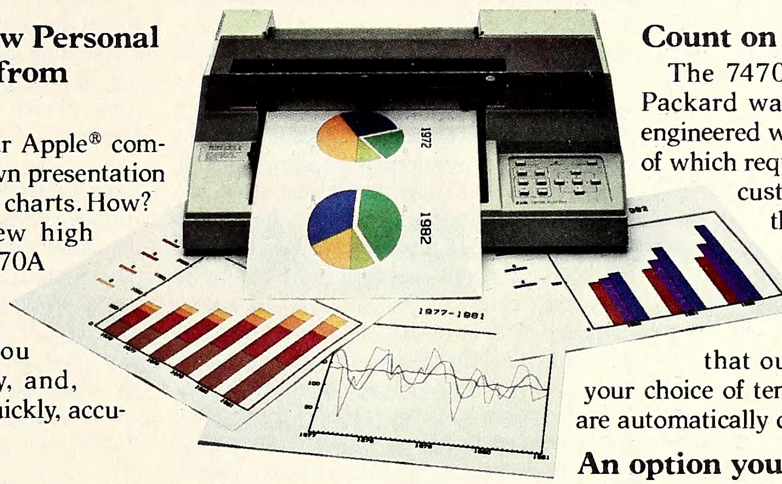
VisiCalc III. Software Arts, Bricklin, Frankston. Just like it sounds; expanded memory, u&lc, 80 columns. Four-way cursor movement. VisiCorp, 2895 Zanker Rd., San Jose, CA 95134. $250.

VisiSchedule. Critical path PERT scheduler. VisiCorp, 2895 Zanker Rd., San Jose, CA 95134. $300.

Word Juggler. Gill. Word processor uses expanded memory. Printout can be reviewed on-screen prior to printing; multiple copies printed of selected pages. Quark Engineering, 1433 Williams, #1102, Denver, CO 80218. $295. *12/82*.

One Apple® and $1,575 can make a lot of pies.

And charts. And graphs.



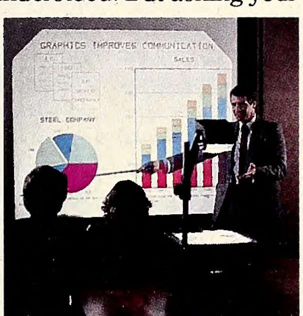
Introducing the New Personal Computer Plotter from Hewlett-Packard

Now you can use your Apple® computer to generate your own presentation charts, graphs, and pie charts. How? Simply add on the new high quality, low cost HP 7470A Personal Computer Plotter.

The 7470A helps you save time, save money, and, lets you communicate quickly, accurately and effectively.

The eye is faster.

Data, when visualized graphically, becomes information fast. Charts and bar graphs can make any presentation clearer and more readily understood. But asking your staff to produce the graphics for your next presentation doesn't ensure accuracy or artistic talent. And going to outside suppliers can be costly. Combined with your Apple® computer, the new HP 7470A plotter does the communicating for you. Quickly. Logically. And with off-the-shelf software.



Fast and pretty.

The 7470A gives you high plotting speed with excellent line quality... faster than any competitive small plotter. On top of all that, it comes in an attractive design package that looks nice on your desk. And it does it for only $1,575. (U.S.A. domestic suggested retail price.)

Count on it.

The 7470A is built the Hewlett-Packard way. To last. Designed and engineered with only a few parts, none of which require adjustment. And with customized integrated circuits that ensure reliability.

Pen Pals.

The HP 7470A has two single-pen stables that output multi-color plots in your choice of ten coordinated colors. Pens are automatically capped and stored.

An option you'll want, too.

For only $95, you can also get a 17057 Overhead Transparency Kit that turns your plots into transparencies for overhead projectors. For "I need it tomorrow at 9:00 A.M!" meetings, it's a necessity.

Start plotting your next presentation today. Clip and mail the coupon below. Now.

Mail the coupon below and we'll send you—absolutely free—a sample plot, a more detailed brochure, and a sample overhead transparency.

Then... stop in at your nearest Hewlett-Packard Dealer. See the HP 7470A in action. Once you see it demonstrated you'll find a hundred ways to make your own Apple® pies. And charts. And graphs.

When performance must be measured by results

HEWLETT PACKARD

Seeing is believing. Send me a sample plot, an overhead transparency, and more detailed information.

Name ______________________ Title ______________________

Company ______________________

Address ______________________

City, State & Zip ______________________

Phone Number (   ) ______________________

Send to: Hewlett-Packard, 16399 W. Bernardo Drive, San Diego, CA 92127—Attn: Nancy Carter

112X2 ST5

OPEN DISCUSSION

The Most Personal

Apple produced the first open microcomputer system. This was more than a hacker's system because it allowed anyone to plug in new hardware. The new Apple IIe is even more open, because now it has a sixty-pin computer slot in it. This is the main improvement. Every signal of any consequence is available to the new sixty-pin slot. What this means is that Apple has now introduced the first microcomputer that cannot be outdated. When the thirty-two-bit processors arrive, and the big changes come about, you will be able to plug them into your IIe. The 6502 can then be used for I/O or graphics.

What we need is to get Dr. Jeppson to write an emulation of the IIe for the Apple III. He is perhaps the only one with enough guts to get the job done.

Apple has now introduced the first truly personal computer, the Lisa. If Apple understands the market that this superb product should be targeted for, Lisa will sell like hot cakes. The reasons are simple. Now one manager, with a Lisa, can do the work of several managers who must use a manual system. The economics are also simple. Cut office labor costs by using the Lisa; more payroll cuts are needed all the time. The Lisa's standard features are so many that it's a bargain, unless one is ignorant of what any system approximating the Lisa would cost.

Give me $50 million and I guarantee in two years to deliver not another Lisa but a desk-top computer that can handle and sort the sum of human knowledge!

Kevin Everett FitzMaurice, Council Bluffs, IA

Well Done

As a new Apple owner with kids, I was concerned that they would only be able to find arcade-style games to play, which were shoot-'em-ups and other forms of violence. Almost immediately I discovered the game *Pie Man* at my local computer store; I was first drawn to it by the cartoonish nature of the cover art. How wonderful it was for my kids and me that the contents continued that theme. Others had told me that the only way arcade-type games could really be exciting was if they involved firing lasers, burning down buildings, shooting down parachutists, or the like. *Pie Man* is none of these—no one is ever harmed and yet the action is enjoyable and exciting.

David Erlich, Palo Alto, CA

Old Faithful

After several years of faithful operation, my old Hayes Micromodem II stopped working in my Integer Apple II. I stuffed it in a box along with a brief note describing the problem and shipped it to the company in Georgia. What a wonderful surprise when a larger box was shortly returned to me containing a brand-new 1983 version of the famous Hayes Micromodem II.

Still more wonders. Upon opening the invoice, I found that it was sent without charge. What more can a customer ask for? Needless to say, my future communication needs, both hardware and software, will be satisfied with Hayes microcomputer products.

Arthur H. Ude, Stoddard, NH

IIe Too!

I have been using *Apple Writer II* and *Quick File II* for about a month now and find these two software packages to be excellent. As a nurse anesthesia educator I have been using *Apple Writer II* for preparation of lecture notes with the greatest of ease and hope to use *Quick File II* for filing banks of test questions for present and future classes to allow for selectivity in printing and formatting tests. However, *Quick File II* may not be the most efficient package for test-authoring available. If readers involved in educational uses of the Apple IIe are aware of test-writing software that could help me, I would appreciate hearing from them.

I am also interested in finding software for keeping gradebooks, attendance records, and comprehensive scheduling of class and clinical time equitably. Please help!

Understanding that the Apple IIe is a relatively new machine, I have not expected an avalanche of software tailored for the IIe's enhancements. However, I hope that *Softalk* will soon include a separate listing of IIe-specific software in Fastalk. I am most interested in applications that will utilize the full two banks of 64K available with the extended memory eighty-column card I now own. For example, I would very much like to see a comprehensive review of *The Incredible Jack*.

Another article that might be appreciated would be a tutorial covering the new Apple Dot-Matrix Printer that would include accessing the array of printing fonts available through *Apple Writer II*.

Stephen C. Smith, Houston, TX

Easy To Be Hard

If there are any Corvus hard-disk users that either have *Apple Writer IIe* or are considering purchasing it, consider this. Though not mentioned anywhere in the manual, it is possible to use *Apple Writer IIe* with the Corvus. All you have to do is type C while it is booting, and *Apple Writer IIe* will respond with "Enable Corvus in Slot 6 ? (Y/N)." Simply respond with a Y, and you are up and running on the hard disk.

And to anyone who is looking for a *great* assembler, I highly recommend *Merlin*, from Southwestern Data Systems. Though it requires a 16K card, it is by far the simplest, while still the most sophisticated, assembler I have ever used on the Apple. It also includes *Sourceror*, an extremely easy-to-use disassembler.

Jeff Jewell, Kennwick, WA

Happiness Is

Softalk is a magazine for users and, hopefully, manufacturers alike. I have several gripes, ideas, and suggestions for the manufacturers that would make people like me, a user, very happy.

Last week I bought some items from Videx and from Apple. The products worked fine, but when it came time to fill out the registration forms the easy stuff stopped. Finding the serial number on the Videoterm was easy—I think. I still don't know if I got the right number. On the Apple letter-quality printer, I had to tilt it on its side to read the numbers. But the worst thing was on the Apple Super Serial Card. There are numbers all over the card, but not one of them looks like a serial number. When I first bought my Apple I got a NEC display with it. What does NEC do? It provides a separate card with the serial numbers on it. This makes it easy to fill out the registration form, and I had a serial number that I could put away with the instruction manual. So much for the gripes.

I have compiled a list of things that my friends and I think would be nice (really, a necessity) to have on all software. The first thing is a fast-loading DOS of some kind. Most manufacturers put this in with their copy protection. Another thing would make games playable on Apples that have the Enhancer II installed.

Don't use the escape key for anything—the Enhancer usually waits for something to follow so the program hangs. The simple solution would be to use a different key for pause, like return. While we are on the subject of pausing games, let's have a pause that waits a few seconds before the game restarts. Trying to restart a game while holding onto a joystick and keeping your eyes on the screen all at once is very difficult. Why not just put a time delay of five seconds after the restart key has been pressed? The last suggestion is that high scores should be recorded onto the disk after the game has been played. The high-score system on *Serpentine* is a perfect example of what I am talking about.

If manufacturers would listen to these few suggestions, it would make computers and computer games more enjoyable.
Timo Bruck, Long Beach, CA

Emulation Indication
I would urge all writers and publishers writing for the Apple II to indicate specifically whether or not the program will run on the Apple III in emulation mode. There are many Apple III owners who frequently put away *VisiCalc* in favor of a good old game. Special thanks to those software publishing houses that already indicate this, such as Dakin-5, Infocom, and Blue Chip Software.
Stephen M. Dorman, Clarkston, WA

Cry If You Want To
I loaned my filing program, *PFS*, to a friend for him to view. He returned it to me at a party. I didn't know what to do with it, so I put it on the kitchen cupboard. At the end of the party, when I went to get the program, to my dismay I found that ice cream had been spilled all over it. When I took it home and tried to boot it up, I found that the disk was all gummed in and couldn't turn in the drive. I thought that all was lost but decided to try opening the plastic case (after all, the program was shot anyway).

I took out the plastic disk and rinsed it off with water and carefully blotted it with some toilet tissue. Then I took out the soft, now sticky, lining, cleaned the plastic disk case, and dried it off. I carefully put the disk back into the case and taped it shut. By that time the disk had fingerprints and stains on it. To my amazement the program booted up beautifully and has been working well ever since.
Tim Anderson, Logan, UT

Bucking the Averages
My Apple II Plus (48K) is connected to a C. Itoh printer (model 8510) through a Grappler card, and I'm afraid that the printer is the most underutilized part of my system. This printer has many features, many capabilities, but I've found it difficult to use, because in my opinion the printer manual is not written for the average user.
D. G. Thomas, Erie, PA

The Orange Oval
I have recently acquired the third in a series of PFS programs—*PFS:Graph*. They have all been excellent, with the latter exceptionally useful. However, the full capabilities have not been exercised because I mistakenly purchased a product from Orange Micro—the Grappler+ printer interface. This interface creates ellipsoid pie charts instead of round ones. I have tried other boards at my local dealer and all work properly.
Phillip P. Brown, Nashville, TN

The manufacturer responds:

PFS:Graph was rewritten by Software Publishing Corporation to provide compatibility with the Grappler+ last year. When this was done, the Grappler+ driver did not account for aspect-ratio variance. The original printer driver was written to interface an Epson printer with an Epson interface card. The hi-res screen-dump distortion was corrected by Software Publishing Corporation for this particular combination.

The Grappler+ does not actually cause the distortion of the hi-res picture. Each printer brand will have a different distortion ratio because graphics resolution values vary from manufacturer to manufacturer. This distortion ratio can be corrected only within the software.

The technical staff has informed me that it is aware of the distortion, and that future versions of *PFS:Graph* will adjust the aspect ratio for an output more closely resembling the video display.
Bob Mickey, technical manager, Orange Micro, Anaheim, CA

Preservation of Vital Assets
The piracy of software is a business within a business. As soon as a piece of software hits the market—that is, the open market and not the computer stores—the product has been "broken" and has begun its trek around the country—in some cases, around the world.

Software manufacturers appear to take the brute-force approach when dealing with this particular problem. They protect, reprotect, and double-protect their disks to such a point that some manufacturers' programs will not boot on my Apple disk drives. This protection hurts both the manufacturer and the end user.

The manufacturer has to pay people to handle the phone and mail complaints arising from the protection schemes. This extra customer support amounts to added costs for the manufacturer, which are passed on to the consumer in the form of higher software costs. The consumer has to face the problem of disk exchange from the local, "friendly" computer store, or possibly face the long wait and cost of replacing the disk via the mail. This experience can ultimately lead to consumer resentment of a particular software manufacturer, and that resentment can be transmitted like a disease, until it destroys one of a company's vital assets—goodwill.

Take a hard look at the copy-protection policies. If manufacturers must protect their disks, they should make sure that any protection technique they employ will be compatible with standard Apple equipment. If they find that a copy-protection technique is incompatible with some standard Apple equipment, they should correct the problem and provide the parties who have defective disks with a copy of the new release. A company that follows this course of action will engender good feeling among the consumers and subsequently increase its goodwill.
David B. Martin, Fort Worth, TX

Conclusions—On and Off the Mark
A valuable and continuing contribution to the use of personal computers would be honest reactions and evaluations of software from those who've purchased and tried to use it.

When I bought my first Apple II Plus, I bought a copy of *EasyWriter Professional* as my word processor. After easily learning *VisiCalc*, *Information Master*, and *DB Master*, I approached *EasyWriter*. To put it bluntly—I failed, as did my daughter, who has a degree in computer science from a leading university. In frustration we traded it in for *WordStar*, which, though much more expensive and requiring considerably more hardware, was at least a usable word processing program.

After becoming proficient in additional programs, I decided to give the original *EasyWriter* a second try. With a year of experience behind me, I did get the original *EasyWriter* up and running, but it is one of the most foolishly designed programs I've yet experienced. The manual is a disaster, and would be better placed in the area of fantasy games complete with hidden caverns, misdirections, and all the ploys found in *Adventure* and other such games.

I recently obtained *ScreenWriter II*; once one gets through a somewhat ponderous manual, it works very well. *ScreenWriter II* is particularly attractive in that it allows users to see the page format without an eighty-column board. (It displays up to seventy columns.) Compared to the *EasyWriter* products distributed by Information Unlimited, it is infinitely superior.

One thing that disturbs me about *Softalk* is the continued listing of *EasyWriter* in the Fastalk column. Why do you continue to do so, knowing that it performs miserably? Had I purchased the program on the basis of the recommendation in Fastalk, I would conclude that these recommendations were bought and paid for.
Robert J. Levine, East Brunswick, NJ

Frankliner's Fidelity
It is my understanding that your subscription policy provides a free year of *Softalk* to all new owners of Apple computers. You state in each issue that *Softalk* "is totally independent of Apple Computer."

The Franklin Ace 1000, which I recently purchased, is Apple II software and hardware compatible. Virtually all programs running on an Apple II will run on the Franklin. Likewise, virtually all Apple II articles, advertisements, and programs contained in the pages of *Softalk* are totally compatible with the Franklin. Why, then, are first-year subscriptions not available on the same basis to Franklin owners? I realize that it would make little sense to give the magazine to all 6502-based computer owners, but owners of Apple "clones" would get the same benefit as Apple owners.

Perhaps you don't know where or even how to "draw the line." May I suggest that a program be developed that would fairly test for architectural and software similarity. Any computer that can run the program and produce meaningful output, unassisted by patches, should qualify for the free subscription. If you are really unrelated to Apple, then why leave out this segment of your market?

I have been a well-satisfied reader of *Softalk* for two years, but only recently purchased my first computer.

I will continue buying the magazine if you do not give me the free subscription; just like an Apple owner, I need it.
Neal M. Rosen, East Brunswick, NJ

Send Out the Clones
When the Apple "clone" machines, Orange II, Franklin Ace, and Pineapple, first came on the market, my initial reaction was to scream bloody murder. Arrest them, drag the perpetrators off to court, and sentence them to drawing hi-res pictures with a keyboard for twenty years. Here I paid a king's ransom for my Apple (bought from a dealer), and these companies are making a fortune by charging less than what I paid for a machine that does the same thing. The non-Apple owners are snickering up their sleeves while their "Franklinsteins" sport extras like built-in fans, numeric keyboards, volume control, and multijoystick capability. Apple software runs equally well on theirs, and they didn't have to mortgage the house and sell the dog to afford it. Well, they'll be sorry when Mr. Jobs and the Wizard of Woz take these companies to court; meanwhile, I'll just pretend they don't exist and refuse to acknowledge that the clone owners have any kind of computer at all!

But when you think about it, the only major difference between the Apple and any other 6502-based microcomputer is the software in ROM. A quick look in the back of the *Apple II Reference Manual* reveals listings of the Monitor and the Autostart ROM. Hardly a way to keep a secret and very helpful for advanced programmers to learn more about the system. The question arises as to whether or not these companies are illegally undermining the success of Apple Computer, or just building machines that follow an industry "standard" set by Apple for a 6502 microcomputer, allowing use of the already flooded software market for Apple programs. A "hardware pirate," by my definition, is one who copies a device in design, right down to the silk-screened pc board. These companies have designed their own machines, with desirable improvements, and sold them for far less than Apple has. Let's face it, Apple has always charged more than it absolutely had to for its equipment. A good example of this is the decrease in price of the Apple II disk drive, after the "compatible drives" hit the market.

I don't see the harm in giving Apple Computer some long-needed honest competition. We will all benefit from it in the long run. I bought my Apple in June of 1980 and began my subscription to *Softalk* the same month. Apple was and still is the Cadillac of microcomputers. The quality control on the Apple II Plus is outstanding and dealer service (rarely required) is typically prompt. If I had to do it over again I might buy another Apple, but only if I was in a position to afford it.
Bruce D. Youmans, Utica, NY

Slow Down!
I absolutely loved your March issue, particularly all the reviews and part 2 of "Apple on the Phone." However, *Softalk* still has one big drawback—Fastalk. It has to be the silliest concept ever! It wastes about seven pages of valuable space and serves no useful purpose whatsoever. I would love to know how many people actually read Fastalk and use it for one reason or another. Do you want to know why I hate Fastalk? It's always wrong somewhere, someplace. For instance, since when are the S.A.G.A. adventures designated classics? They're fairly new and are merely spruced-up versions of the original Scott Adams adventures. I also feel that *Hi-Res Adventure #2* is a classic, after being in the Top Thirty for over a year, as well as *Zork I*, for continuing to be in the Adventure 5 list and the Top Thirty. And what about *DB Master*? Or *Ultima*? Or *WordStar*? Or *DOS Tool Kit*? Or *LISA 2.5*? All of these programs are definite classics and should be labeled so. And since when is *Crossword Magic* a classic? I have never even heard of it or seen it in the Top Thirty. And what about *Wayout*? If it can be a classic, then so can *Choplifter*, *Snack Attack* (which should be), *Swashbuckler*, and many others.
Peter T. Clark, Sacramento, CA

Of Stellar Modifications
To Greg Tibbetts: First, let me say how much I enjoy SoftCard Symposium. It certainly has helped me to understand and appreciate the capabilities and potential of Microsoft's Z-80 card. I initially bought the card to allow access to CP/M-based word processing (*WordStar*) and found out there's a whole new and powerful world wrapped up and plugged into slot 4! Along with learning CP/M from books and help from your column, I am also digging into the internals of the *WordStar* program and realizing that there's a lot of variables that can be manipulated to customize the program to suit anyone's particular hardware setup.

I guess the reason why I am writing is to ask for a little help. I am sitting here with an 80K Apple, Z-80 card, *WordStar* (3.01P), and an NEC PC-8023. The printer is not being utilized to its full capacity. There are features that the

WordStar program is not calling and will not unless I load DDT.COM and have at the internal workings. I've read the *WordStar* manual and will admit that I really don't understand how to install scrollup and scrolldown, nor backspace, nor a host of other little goodies that would be nice to peek and tweak this particular configuration.

How about showing the power of DDT.-COM and other commands, as well as providing us with a few vivid examples by using *WordStar* as the guinea pig?
Chris Stearn, Mayaguez, Puerto Rico

Greg Tibbetts responds:

One of the most difficult things for a contributing editor such as myself is the inability to be all things to all people. Your particular concern in this case is the *WordStar* word processor. It's certainly a valid concern and a widely used program. Among the thousands of other readers of the column, however, exist several hundred other legitimate and worthwhile concerns. It is not within my power to address each person's specific area of need. The best I can hope to do is act as a sort of guide through the programs and material shipped with SoftCard and with CP/M.

Where possible during discussions on some of these programs, I do try to incorporate simple examples with wide application that, unlike *WordStar*, will not create more questions than they answer. Each reader's need for *WordStar* modifications, even assuming that all readers have and use it, would be quite different, requiring considerable customization and explanation. To attempt even a light touch on this would be a book in and of itself. In fact, several books exist on the subject of *WordStar*, and most of these don't attempt to deal with the customization of that product. To deal with this in the few pages allowed me in the column would be unfair to the product, to the readers, and all in all would produce a less-than-desirable result.

I hope this explains, at least in part, why I do not choose such topics as *WordStar* for example material. As to where one could go to get such information, you might seriously consider contacting MicroPro's technical-support section for the names of publications that discuss *WordStar* modifications. Personally, I have never needed to go beyond the *WordStar* manual and MicroPro's customization notes, although I do remember spending considerable time reading and rereading the same paragraphs before it became totally clear. In future columns, when and if the actual operating system information has been thoroughly discussed, I will probably begin to take some of these subjects on.
Greg Tibbetts, Santa Barbara, CA

A Real Nowhere File
Doug Carlston suggested the following program line that enables users to catalog during a load or save by asking for a file with no name.

```
25 VTAB 23: IF A$ = "" THEN PRINT
   CHR$(4);"CATALOG":PRINT:PRINT
   :GOTO 20
```

Why doesn't changing "" to CHR$(13) or CHR$(32) do the same?
J. C. Gobins, Jr., Phoenix, AZ

Doug Carlston responds:

Input treats both of these characters in a special manner. If you hit return (or control-M, which is the same thing), this is a signal to input to stop accepting input. Therefore:

```
10 INPUT A$: IF A$ = CHR$(13) THEN
   PRINT "AHA!"
20 GOTO 10
```

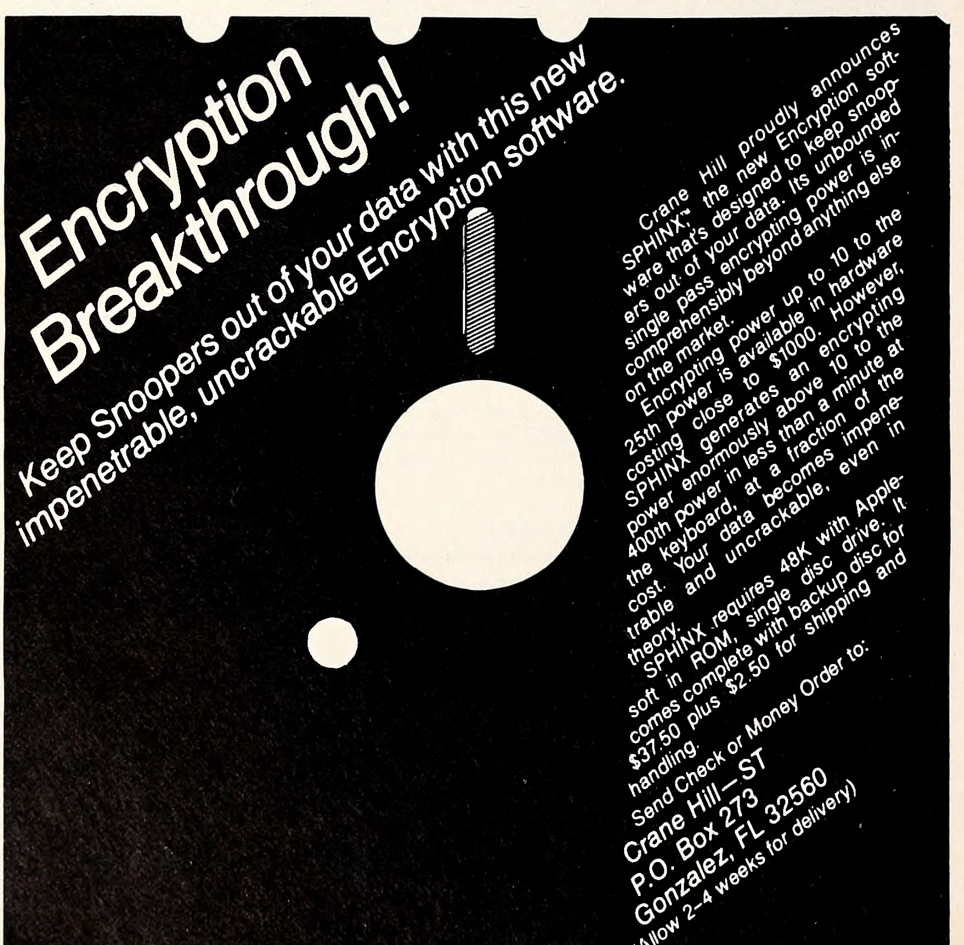
You will never be able to get an "AHA!" out of your machine, since the return you press will always be taken as a signal that input is finished. Likewise with CHR$(32), the space:

```
10 INPUT A$: PRINT LEN(A$);" "A$: GOTO
   10
```

Try entering the following:

```
HELLO (then press return)
  HELLO (then press return)
    HELLO (then press return)
```

Leading spaces are ignored by an input statement. If you had tried most other keys (one not treated in a special manner by input), your approach would have worked.
Doug Carlston, San Rafael, CA



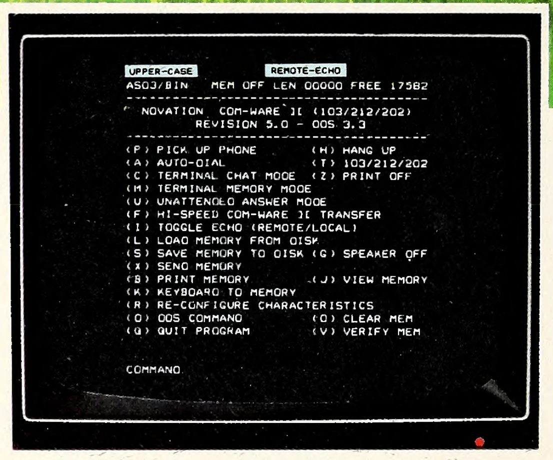
Teen Dream
I think Bill Budge is a gorgeous hunk! Why don't you call him up for an interview some time, and pretend you're going to talk about his new *Pinball Construction Set*—but instead just take lots of pictures. You could publish the pictures and pretend you forgot to publish the text—and in the next issue publish an erratum, explaining that, golly, you left the text of the interview lying around somewhere but couldn't remember just where. Then, if Mr. Budge got upset, you'd go over to his house, pretend to re-interview him, but actually take more pictures.

I think Doug Carlston is cute, too, but he's older. (I'm only 17.)
Poppi Kosak, Novato, CA

SoftGraph Glitch
I have been following David Durkee's SoftGraph series and have been avidly entering his programs each month. When my March issue arrived, I entered the *Pie Chart* program that appeared in it.

It worked like a charm, except for one minor error. No matter which column of data you select to graph, the labels printed are the labels for the first column. In other words, if you elect to graph the data in the third column, the graph will be correct, but the labels will have the data from the first column. The error is in statement 740. The first command on that line should be changed from $HF = 2$ to read $HF = PR(2) + 1$.
Norman L. Kushnick, Baltimore, MD

Apple®-Cat™ II



The Cat system lets you slip something into your Apple II no other modem offers—a complete range of speeds from 110, 300 and 202 half-duplex—to full duplex 212

Either way, you have state-of-the-art LSI technology. And it means you can start right off with the most advanced system available. Or you can trim your investment, yet always have the option to move up at any step with absolutely no compromises in quality.

Com-Ware™ software is part of the package.

Five minutes after you've booted up the Novation Com-Ware you'll have a good notion of what it's like to work with the best, most accurate, most convenient personal communication system designed for your Apple.

It makes all of the moves you need to work with another computer, swap programs, access data, whatever.

And it's simple to operate. Just follow the menu. No programming, no fussing. It's all there.

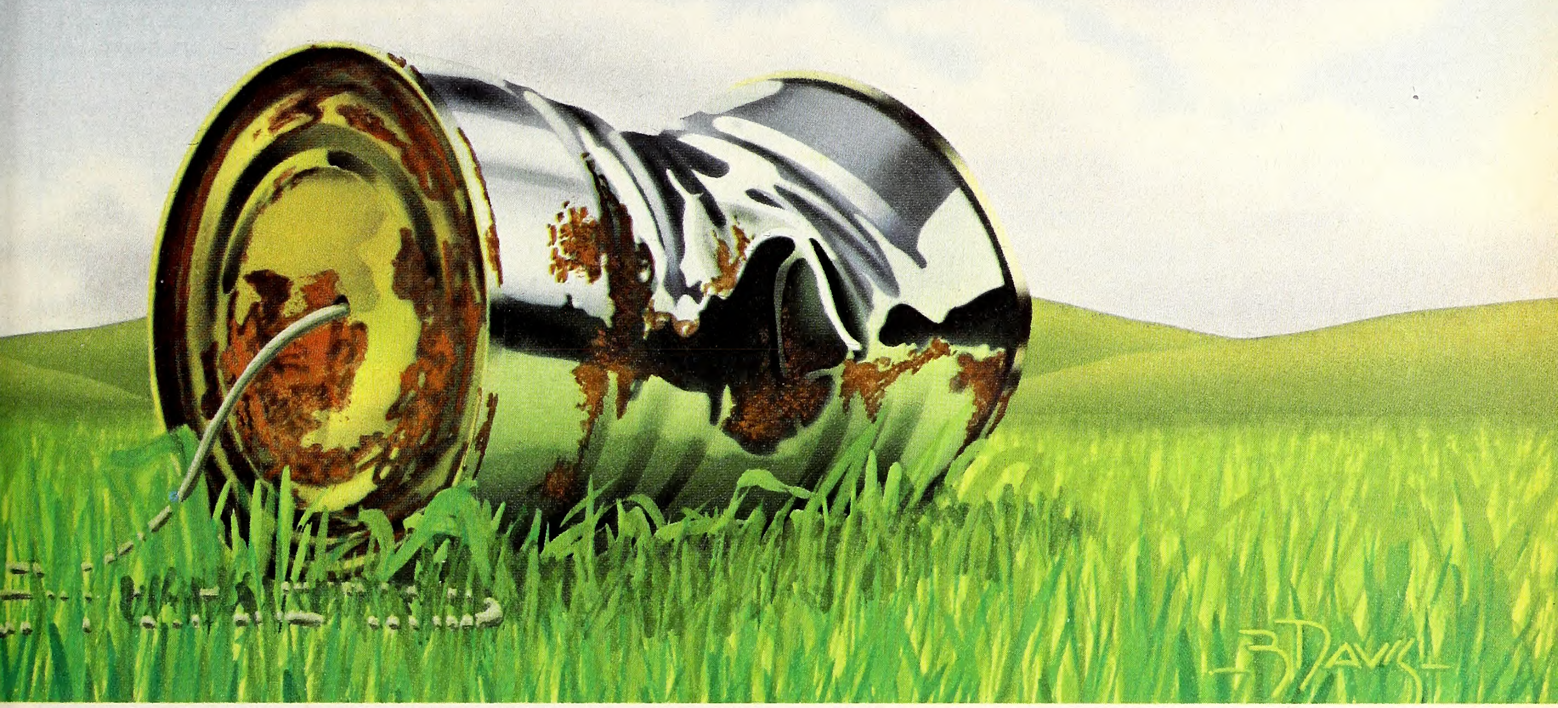
New—telephone directory.

Our engineers have done it again. They've expanded our Com-Ware. Now included: a time-saving directory of 26 telephone numbers with terminal configurations all selected and stored for auto dialing. Handy.

And some nice extras. Print-out during communication, a non-destructive memory mode, a changeable "welcome" message for automatic answer. There's more.

If you have an Apple-Cat II and our earlier Com-Ware (4.4 or earlier), you should really have the new one (5.0). As always, it's free.

It's the modem/communication system you grow into, not out of.



Just send us your old diskette and we'll send you the new one.

It can make all the moves you want.

Start with 110, 300, 202 half-duplex. *Or*—add 212 full duplex and move data four times faster with accurate, block by block verification. / Automatically send and receive anytime—including the middle of the night when line charges are lowest. / Use 80 or 40 column format. / Set for local or remote echo. And more.

Two more reasons to move right now.

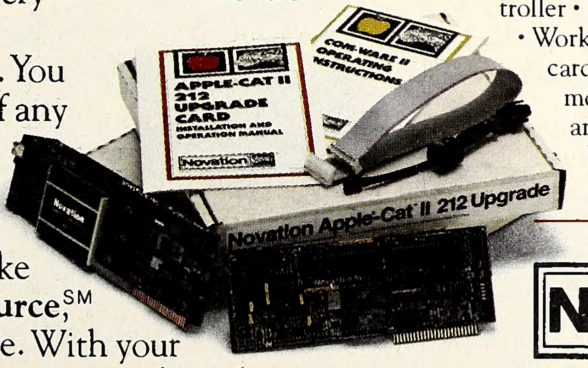
Packed with every Apple-Cat II is a list of options. You get your choice of any one—and save up to $40.

Also, if you've ever wanted to take a look at **The Source**,SM this is your chance. With your Apple-Cat II you get a sampler subscription offer. It's limited, but it gives you a taste.

They're at your dealer.

He has all of the details. See him now.

Easy installation.
In less than 10 minutes, you're talking to the world.



More features than any other modem.

• Full range of communication baud rates—up to 1200 (Bell System 100, 202 or 212 series compatible) • Full or half duplex operation • Complete Com-Ware system on a single diskette. Also, wide assortment of excellent software available from other sources • All automatic functions—auto dial (pulse or Touch Tone), redial, auto answer and disconnect • It's a telephone with speaker monitor—switch between data and voice. For regular use, it's a handy intelligent phone with auto-dial • Touch Tone receiver • Built-in BSR X-10 Controller • Remote control for external cassette tape recorder • Works with other Apple parallel or serial printer interface cards • Constant status display on screen • Binary or text modes • Single card installation for Apple Cat II and an additional card for the 212 upgrade • FCC certified built-in phone line interface (PLI) Module • New full Duplex 212 option.

Novation

Novation, Inc.
18664 Oxnard Street
Tarzana, CA 91356

(800) 423-5419 • In California: (213) 996-5060

David Durkee responds:

The "labels-in-two-columns" option was designed to allow labels of larger than eight characters, in which case the second column must be the one read for the second half of the label, as shown in the upper-left screen shot on page 76 of the March SoftGraph. Making Mr. Kushnick's correction will give you a different use for this parameter. The second column of the data will be the numbers that are graphed, which is useful as well, but it eliminates the possibility of using more than one column for labels, as was originally intended. This means that categories such as "Development," as shown in the screen shots, would have to be abbreviated to "Devlpmnt," or something similar.

In short, you can have one of these options or the other, but not both. Incorporating both options would be possible, but it would require rewriting the parameter-input label-reading routines.

Also, several readers have called to report a problem with SoftGraph. When *Bar/Line Chart* gets to the point where it is supposed to label a graph, it goes haywire and draws horizontal and vertical lines all over the screen. Those familiar with the operations of shape tables will instantly recognize what's wrong: The scale hasn't been set to 1, so it has remained at its default value of 0, which is the equivalent of 256.

This was mysterious because the scale setting was supposed to be in the hello program (at the end of the February article), and, in fact, is there on the disk. Not so in the magazine. To correct this fault, change line 10 of the hello program to read *TEXT : HOME : SCALE= 1: ROT= 0.* Those who purchased SoftGraph on disk escaped the problem.

If you want to see the kind of tribulations your ambitious counterparts who typed in the program went through, boot the disk, quit from the menu, type *scale=0* and *run*. Then run *Bar/Line Chart* as you normally would.
David Durkee, Burbank, CA

Jabbertalky

In the March *Softalk*, Allen Munro makes a reference to *Alice in Wonderland* in which he claims the Mad Hatter says that words mean what he wants them to mean.

Though very apt, it was Humpty Dumpty who made the remark, and it's from *Through the Looking Glass.* " 'When I use a word,' Humpty Dumpty said, in a rather scornful tone, 'it means just what I choose it to mean—neither more nor less.' "
Timothy King, Monmouth, OR

An Isolated Case?

After one year with Apple and *Softalk,* I now appreciate the value of your publication. I would like to suggest, though, that writers provide a brief bibliography at the end of articles when it applies. These references need not exceed five items. This would greatly facilitate the spread of knowledge on the more obscure topics. The problem in my geographic location is a lack of computer stores or computer books in bookstores. An occasional mention of quality refer-

ences or reviews of Apple-related books new to the market would be helpful.
Dan Moody, Winterport, ME

Poking the Byte out of the Bag

Yes, I am game. I, of course, am writing in response to the responses to Mike Mahone's original reset problem. I myself remember a problem I had not at all dissimilar to Mahone's a year ago. I was writing a program for my junior high school, and I wanted it to be totally seventh-grader-proof. This was before I belonged to Mini'app'les, and before I ever picked up an issue of *Softalk*. I called every computer store and computer person I knew, and the best answer I got was, "Flip the switch on the encoder board so the user must press control-reset." This did no good because I needed a software solution. I looked for hours before I found page 37 of the *Apple II Reference Manual*. I have been intrigued by the answer ever since.

Whenever someone presses reset, the Apple takes a look at what is in locations 1010 ($3F2) and 1011 ($3F3). It takes the LSB out of 1010 and the MSB out of 1011 and jumps to that location. To find what the address is, type

```
PEEK (1010) + PEEK (1011) * 256.
```

According to the manual, it is supposed to contain $E003. I found that to be true most of the time. Sometimes it has different addresses, but they all seem to be some kind of DOS rehook location.

If that were the case alone, the reset key would not be very difficult to understand. But there's more. When Apple came out with the Autostart ROM, it had to have some way of knowing whether the Apple had just been turned on or not so it could carry out the proper code to boot itself. (This is otherwise known as doing a cold start.) This is where 1012 ($3F4) comes in. It is affectionately known as the "power-up byte." The tricky Apple system Monitor checks to see whether 1012 ($3F4) contains the exclusive-or of $A5 (165) and whatever is contained in 1011. In Basic terms, that's peek (1011). Normally, this is $E0 (224). (Remember $E003?) Therefore, if 1011 contains $E0, as it should, then

```
$E0 (11100000)=224
$A5 (10100101)=165
------------------
EOR=$45 (01000101)=69
```

If 1012 agrees with 1011 by containing the EOR of $A5 and the contents of 1011, then reset will jump to the location specified in 1010 and 1011. If 1011 and 1012 do not agree with each other, the Apple thinks someone has just turned it on, and it reboots. Simple as that. I hope that is clear to you.

For those of you who don't wish to figure out the exclusive-or after you have changed 1010 and 1011 to suit your needs, the system Monitor has a routine to do just that for you. It can be accessed as follows: *call 64367*, in Basic; or *call −1169*; or from assembly language *JSR FB6F*.

Therefore, if you want your Apple to jump to a specific location, do the following:

```
POKE 1010, ADDRS − ( INT ( ADDRS / 256)
* 256) : POKE 1011, INT ( ADDRS / 256 )
: CALL −1169
```

where ADDRS is the address you want it to jump to. If, however, you simply want it to reboot, just set 1012 to a value that does not agree with 1011. (Meaning, of course, that it does not have the EOR of $A5 and the byte contained in 1011.) Try this:

```
POKE 1012,X
```

where X is any value that is not the exclusive-or of $A5 and peek (1011). Zero usually works fine for starters.

Now that you understand how the Apple treats the reset key, you can make reset do a jump to any location in memory. You can make it go to any machine language routine of your creation, so long as it can find its way back. (Remember, when reset is pressed, your Apple does a JMP to the location specified, *not* a JSR!) The addresses I like to use as a value for ADDRS previously mentioned are 54630 ($D566), which is equivalent to RUN, and 976, which reconnects DOS.

I like to use the last one most of all because it doesn't appear as if reset is acting any differently than normal, but it is actually reconnecting DOS. (Don't you hate it when you try to save a file you have been working on and it gives you an unappreciative syntax error?)

There are many routines I would like to use, but I can't because they end in an RTS. For instance, try using 42350 ($A56E), which prints a catalog when called. It will work, up to the point when it tries to get back to Applesoft. Instead, it will continue executing whatever is at the top of the system stack, which could mean disaster. Try it sometime (that is, when you don't have anything important in memory). I would *like* to use −3100, which displays page one hi-res graphics, but that also ends in an RTS.

In this not-so-brief letter (my English teacher, who is also a computer buff, calls it an "article"), I hope I have opened up some interesting possibilities using the reset key. If you still need to know more, try (and I mean try) to read pages 36 and 37 of the *Apple II Reference Manual*. Also, even though I can't say I have read it, an article called "Trapping the Reset Key" appeared in *Nibble*, volume 2, number 5. I did, however, read a letter responding to that article in *Nibble*, volume 3, number 7, that made some "vital modifications" to the program that would trap reset with an onerr goto statement. If anyone wants to know more, I'll be glad to try to help; I always like to have more people to exchange information with.
Loren Ryter, Minnetonka, MN

Meeow!

"Pokes 'n' Boots" (March Open Discussion) was great! I found it unsuccessful, though, in a security program I'm developing. With *poke 1010,102* and *poke 1011,213* plus *call 64367*, the program restarted as expected. Then, when I punched in the correct code, every time thereafter reset ran the hello program again. How can I undo the pokes so I can program in peace?
Robert Buschel, Hollywood, FL

COM-STAR F/T

Tractor Friction Printer

only $299

FREE
Box of printer paper and demo tape with purchase.

- Lowest price quality tractor friction printer in the U.S.A. • Fast 80 characters per second
- 40, 46, 66, 80, 96, or 132 characters per line spacing • Prints labels, letters, graphs, and tables
- List your programs • Print out data from modem services

Deluxe COMSTAR F/T PRINTER — $299.00

The Comstar is an excellent addition to any micro-computer system. (interfaces are available for Apple, VIC-20, Commodore-64, Pet, Atari 400 and 800, and Hewlett Packard) At only $299. the Comstar gives you print quality and features found only on printers costing twice as much. Compare these features.

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

If you want more try —

Premium Quality COMSTAR F/T SUPER-10" PRINTER — $389.00

More Features Than MX-80 For $250 Less

For $389.00 you get all of the features of the Comstar plus 10" carriage, 100 cps, 9 x 9 dot matrix with double strike capability for 18 x 18 dotmatrix. High resolution bit image (120 x 144 dot matrix), underlining, backspacing, 2.3K buffer, left and right margin settings, true lower descenders, with super and subscripts, and prints standard, italic, Block Graphics, special characters, plus 2K of user definable characters. For the ultimate in price performance the Comstar F/T Super 10" leads the pack!

80 COLUMN PRINTER $199

Super silent operation, 60 CPS, prints Hi-resolution graphics and block graphics, expanded character set, exceptionally clear characters, fantastic print quality, uses inexpensive thermal roll paper!

Double Immediate Replacement Warranty

We have doubled the normal 90 day warranty to 180 days. Therefore if your printer fails within "180 days" from the date of purchase you simply send your printer to us via United Parcel Service, prepaid. We will IMMEDIATELY send you a replacement printer at no charge via United Parcel Service, prepaid. This warranty, once again, proves that WE LOVE OUR CUSTOMERS!

15 DAY FREE TRIAL

OTHER OPTIONS

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

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

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

COMSTAR F/T

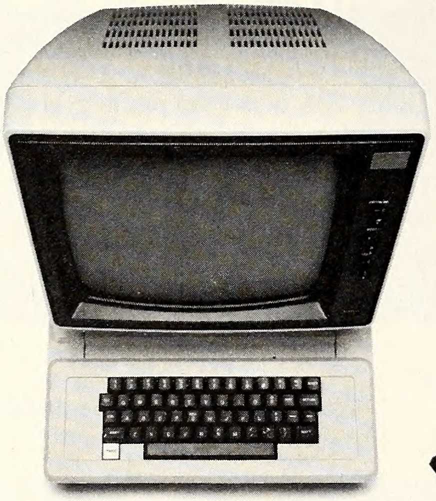
ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijk
lmn opqrstuvwxyz 1234567890
ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890

SUPER-10" ABCDEFGHIJKLMNOPQRSTUVWXYZ
ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890

IF YOU'RE CONFUSED PERSONAL COMPUTER,

At this moment, there are no less than 50 personal computers on the market. And more are being introduced every day.

On one hand, having all those options is a good thing. On the other, it can make picking the right one pretty difficult.



Computers come in two parts. You have to buy both.

We'd like to help. So here are a few suggestions about how to buy the computer that's right for you.

Computers come in two parts.

One part is the "hardware," which is the machinery itself. The other is the "software," or a program, as it's sometimes called.

Software is the part that tells the computer what to do, the way a driver tells a car what to do.

Without software, a computer can't do anything.

And vice versa.

You have to buy both.

Buy the software first.

Since the reason you're buying a computer is to get the capability the software gives you (remember, it's the software that knows how to get things done), it makes good sense to pick the software first.

Start by making a list of the things you want to use the computer for. It can include almost anything—any kind of inventory, filing, accounting, graphics, reporting, record-keeping, analysis—you name it and there's probably a software program that does it.

Next, take the list into a computer store and ask the salesperson to give you a demonstration of the program, or programs, that will do the things you want.

Even though you'll need a computer for the software demonstration, keep in mind the computer is just a vehicle. The software is the driver. And once you've decided on the software, picking out the rest of the computer system will be much easier.

The simpler the better.

Look for software that's easy to learn, easy to use, and that does the job in the simplest way possible.

Good personal software should be, as the computer people say, "friendly." Meaning that it helps you do what you have to do without getting in the way.

Meaning there are no complicated routines to follow to perform a simple task. And no programming language to learn.

Some people, however, will tell you that software has to be complicated to be powerful.

Nothing could be farther from the truth.

Because in order for a program to appear simple to you on the outside, it has to be extremely complex on the inside.

I'd like to comment on Matt Offenbacher's *poke 50,(any number from 0 to 254)* mentioned in "Pokes 'n' Boots." (How about a section in *Softalk* devoted to these obscure peeks, pokes, and calls?) Location 50 is the video mask byte. If it has 255 then normal output; if it has 127 then flashing output; if it has 63 then inverse output. Any other number will cause garbage output.
Mike Davis, Pensacola, FL

Friends of Pascal
I think I can help with the question in the March Open Discussion about a Pascal user group. The UCSD Pascal System Users' Society, or USUS (pronounced "use us"), is a nonprofit organization, independent of all vendors, that was created to promote and influence the development, education, and exchange of the UCSD p-System. USUS periodically holds meetings around the country and publishes a quarterly newsletter that serves as a forum for its members. USUS also supports a software exchange library from which members can obtain software for a nominal reproduction charge. Individual memberships are $20, payable to USUS, Box 1148, La Jolla, CA 92038.
Alex Kleider, Sioux City, IA

To Hobart Cable: The current address of the Pascal Users' Group appears to be 2903 Huntington Road, Cleveland, OH 44120. The company has been "moving around" lately but I have a return postage-paid envelope from it with this address.

To Thomas Batson: I have both versions of Pascal that you mention—Apple's and the Softech version for the Apple. I haven't used the Softech version very much. In fact, it is currently sitting unused. It represents certain enhancements to the system with regard to multitasking, screen control, and segmentation, but it has some serious problems in my opinion. First, it appears to be incompatible with many of the accessory boards for the Apple slots that so many of us are fond of. This is supposedly handled with the user-definable drivers. This, however, requires thorough knowledge of assembler as well as the internals of the board in question. Softech provides some drivers, but not too many, and this leaves a large gap. A specific example: Softech Pascal will not even boot on my Apple, let alone operate, with the Novation Apple-Cat modem installed. Calls to Softech have produced a shrugged shoulder and the comment from the company that Softech never claimed its Pascal would be compatible with all boards (but Softech never said it wouldn't be either). A call to Novation informed me that the company only advertises its board as being compatible with Apple Pascal. Sound familiar to any of your readers? Softech Pascal is more disk-dependent (swapping) than Apple's version. Unless you have enough disks on-line to leave the system files on-line at all times, it can be frustrating. Again, an example is that the filer code must remain on-line in order to do some of its operations. Not so with Apple Pascal; once you are in the filer, you can take the disk out.

In closing, I would like to recommend that Thomas Batson get a copy of *Apple Orchard*, volume 3, number 2 (May-June 1982). This has a nice article by Dr. Wo on the various Pascal versions. I would have preferred that it be a little more critical in its review, but it is still a good, interesting article.
Clinton L. Collier, Walnut Creek, CA

Translator Needed
I have a data storage and retrieval program that runs in Pascal. I do not otherwise use Pascal. I do a great deal of word processing with *Apple Writer II*. It has dawned on me that I won't be able to make full use of my stored data until I can load it out of my Pascal text files into my *Apple Writer II* text files.

I understand there is a translator program that will turn Pascal text files into DOS text files but that it needs a Pascal operating system, which I don't have. Does anyone have any advice?

I suppose another solution to the problem would be to go to word processing in Pascal. I've had no success so far in finding out how to use Pascal as a word processor. Can anyone help?
John D. Ayer, Davis, CA

Upgrade References
I would like to offer sources to make the good old original *Apple Writer* program a more

modern and useful text-editing system. The original program was very simple to use and could be copied to any disk. This made it easy to use one disk for letters, another for reports, and so on, without having to switch disks or use two drives. One shortcoming was that non-ASCII letter codes were used. This made it impossible to enter printer commands for print size, underlining, and the like. Another disadvantage was that everything was displayed in upper case, making it difficult to read. Capital letters displayed in inverse did not help avoid capitalization errors.

The following articles can upgrade your old *Apple Writer* program. For lower-case display and standard ASCII code files, see the February 1981 issue of *Creative Computing*. A Dan Paymar or similar lower-case adapter is required to display lower case. The June 1982 issue tells how to incorporate printer control codes into *Apple Writer* files. For example, all I have to do is enter control-I and E to insert a file called Text.E containing the printer code for elite printing. Another article in the same issue tells how to preview your file on-screen just as it will print.

For entering lower case into Basic programs, see the April 1982 *Call —A.P.P.L.E.* To wire in the shift key, see the May 1982 *Call —A.P.P.L.E.* I had to figure out for myself how to add this shift-key modification to the *Apple Writer* program. It works great, except for special characters that normally use the shift key in *Apple Writer*. With my own shift-key mods these characters cannot be entered from the keyboard but can be kept in a file and inserted like printer control codes. In the same manner you could also print any graphic symbols available in your printer, such as Greek letters.

With lower-case display, use of the shift key, standard ASCII coding, and printer control codes entered into files, I think that my good old modified *Apple Writer* is the most versatile and easy-to-use text-editing program I could find. It uses the same keys for editing as Applesoft uses. I laugh when I look at the complicated manuals of other word processing programs. I would never buy a copy-protected program (other than a game) because no program does everything the way I want it to.

David Efflandt, Elgin, IL

Gettin' Cookin'

I am interested in hearing about anyone's experiences with software designed for the maintenance of a recipe file, as well as other cookbook-related functions. I am already aware of *Micro Cookbook* from Virtual Combinatics, but I'd like to hear about some other products as well.

Nancy Stanger, Chatham, NJ

Dialing Data for Doc

I have been reading with interest the "Apple on the Phone" articles, and now I have a question perhaps someone could answer. Are there any independent on-line databases? In particular, I'm interested in those relating to medicine.

W. V. Cuthrell, Portsmouth, VA

Community Conscious

Our school uses only Apple computers, and we find them very reliable. We are pleased with the amount of materials available for it. The users' group that we participate in is for all types of machines. Do any readers have any information that would help us become better organized, or activities that we could offer to the community? Our long-range objective is to offer a computer fair so that nonusers and potential buyers can see what a micro can do.

Mary Ann Emerick, Bancroft, IA

Major Announcement

The 175th Medical Brigade, California Army National Guard, has formed a clearing house for military users of personal computers. The unit has established an Apple II software library of military applications, as well as a database of personnel and units who are using any personal computer. Anyone interested may contact me, the clearing house project officer.

Major Jack L. Espinal, 3250 Meadowview Road, Sacramento, CA 95832

At Long Last—His Theory Proved

I had always theorized that there exists a parallel, mirror-image universe occupying the same time and space as our own. For years, I have searched for some thread of evidence to confirm this theory. I have spent hours combing through reams of paper in such prestigious publications as *Scientific American*, *Science '83*, and *Science News*, but to no avail. Discouraged about my failure to confirm my theory, I turned to such mundane activities as playing *Choplifter* and reading *Softalk*. After working my way through the March articles on Apples and lasers, and Apples and art, and Apples and skating rinks, I got to another article on word processing. Lord knows there are enough articles on word processors, so I simply turned past it to the *Softalk* fiction section.

Then it hit me. Hurriedly turning back to the *Pie Writer* article—I saw it. There it was: proof of my theory on page 236. Through some fluke, here was photographic evidence proving the existence of a mirror-image universe. Through some imbalance in the silver ion stabilizer photographic process, or possibly due to a small rotating black hole near the photographer, here it was: a perfect picture of Peter Trafton's mirror-image, antimatter self in living color! Not only was he inverted, so was his Apple, with the power light on the right and everything reversed. (I wonder what kind of keyboard notation they use? YTREWQ?) Not only this, but every book on the shelf behind Peter Trafton reads from right to left, like Hebrew. A great discovery such as this should not go unnoticed. It could even make it into the *National Enquirer*!

I would like greatly to discuss the procedure for making this photograph with the photographer. There could very well be a Nobel Prize in this discovery. Who knows? A discovery of this magnitude may be the best thing since sliced bread!

Scott Sanders, Birmingham, AL

SOFTALK CLASSIFIED ADVERTISING

THE REMINDER CALENDAR

Better than an electronic appointment book; everyone from businessmen to housewives will enjoy using it everyday! Send $24.95 to P.O. Box 5701, Forest Pk. Branch, Dayton, Ohio 45405 (add $1 for C.O.D.).

HO$$

Thoroughbred Handicapping Tutorial
Learn computer race handicapping methods. Five Complete Systems, track, speed, jockey, etc. factors. 200+ page handbook includes Applesoft documentation & disk. User Modeling Coefficients to meet YOUR needs. Menu driven, Apple II+, DOS 3.3. $89 incl. tax. TOUT Company, 360 S. Gordon, Pomona, CA 91766.

SAVE AT SOFT WAREHOUSE

We offer the lowest prices on business, education, and entertainment software. All major brands are available. Write for our free price list.
SOFT WAREHOUSE
P.O. Box 153
West Islip, NY 11795

VISICALC III USERS

DIF SORT: Program will sort a *VisiCalc* DIF file by rows or columns. Sort the worksheet in ascending or descending order. Menu-driven. Includes sample files and tutorial manual. Also a "SOS" commands sub-menu helps you manage disk space, filenames, and sub-directories. Listable. Requires Business Basic. $50. NC Sales, 1111 Archwood Dr., #318, Olympia, WA 98502.

FREE SOFTWARE

Join The Big Red Apple Club, a national Apple user's group with benefits including monthly newsletter and a large library of free software. Annual membership $12. Sample newsletter $1. BIG RED APPLE CLUB, 1301 N. 19th, Norfolk, NE 68701; (402) 379-3531.

$$LOW LOW PRICE$$

Call or send for our free catalog of hardware and software. Satisfaction guaranteed. Software, Etc., 16037 Kelly Road, Suite 1, Fort Myers, FL 33907; (813) 433-2200.

APPLE JOYSTICK $26.95

$59.95 too much money? We thought so. Try the DDS joystick. Full range inputs, two buttons. Satisfaction guaranteed. Ask your Apple dealer or order direct. Add $2 shipping. Deger Data Systems, 1001 E. Walnut St., Goldsboro, NC 27530; (919) 734-7183; NC tax 4%.

PRINTERS-MONITORS-SOFTWARE

Amdek, NEC, Okidata, USI, etc. Dealer inquiries welcome. Best prices anywhere—compare. Call or write ARGONAUT DISTRIBUTING, 1104 Buchanan, Antioch, CA 94509; (415) 778-2595.

SOFTWARE DEVELOPERS

You've spent many nights developing your program. Don't let a poorly written user's manual hamper your chances for success. For free information on our editing services write: TECH-ed, 902 Wilson Blvd., Mishawaka, IN 46545.

COMPUTER ROAD ATLAS

Start your next trip with a computer listing showing the best ... ROADSEARCH computes the shortes
Prints the route
Contains 372 U
road miles. DO
guarantee. $3
Check/Visa/MC
2235M, Columb

EXTREM

Authorized Ra
software. We
The
7603 E.

SUPER

Request free
grams at unb
reduced pho
on software.
time; or wri

MOR

Now turn
chine just
difference is that
lose any money playing. Requires either ... ple II+ with 48K or an Apple IIe. $19.95. Jersey Shore Software, P.O. Box 751, Barnegat, NJ

AT LAST! A BETTER WAY TO BUY SOFTWARE!

The Problem: Even with ads and reviews, it's hard to judge if a program will be enjoyable. *The Solution*: Join CompuFun—a unique club that enables you to evaluate software *before* you buy! STOP WASTING YOUR SOFTWARE DOLLAR! Send today for *free* membership kit. CompuFun, Dept. S-5, 5753G E. Santa Ana Canyon Rd., Anaheim Hills, CA 92807.

FLIGHT PLANNER by Jerry Kennedy. You supply point of departure, destination, and altitude, and *Flight Planner* computes your route of flight using VORs. Computes RNAV routes too! The computer-designed routing ensures the shortest distance VOR routing for your flight no matter what altitude. See the review in the February issue of *Flying* magazine. Only $59.95 from Illinois Computer Mart, Route 8, Sweet's Corner Plaza, Carbondale, IL 62901. Phone (618) 529-2983.

WOLFENSTEIN PLAYERS

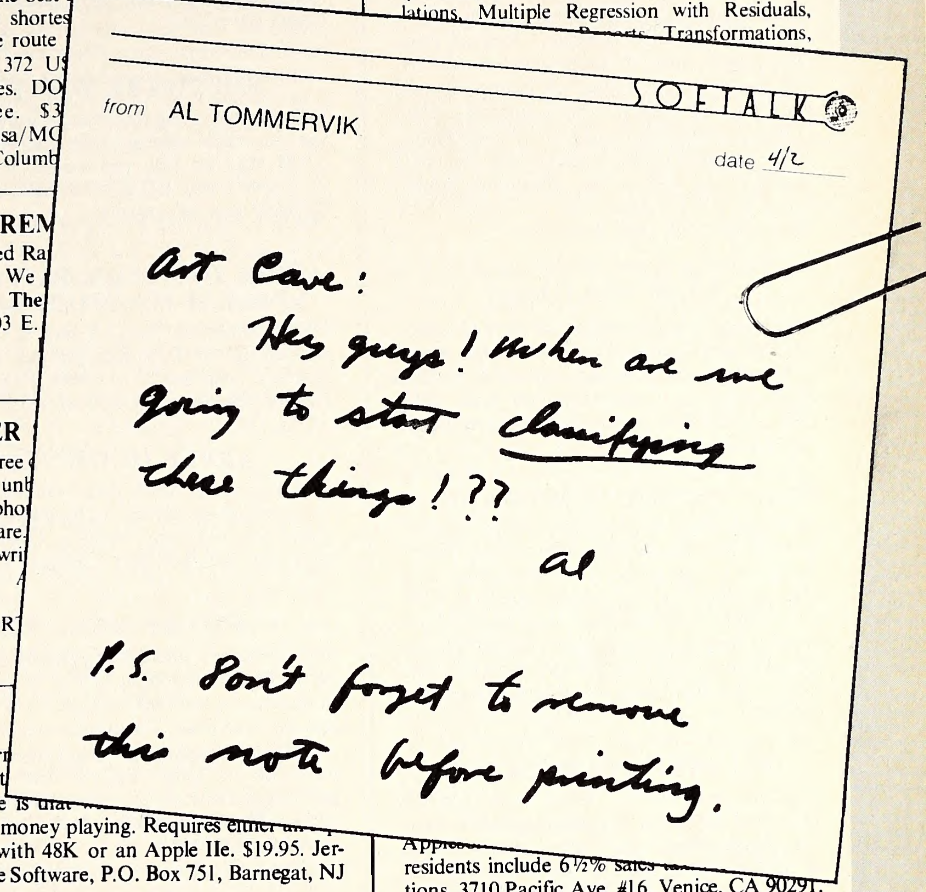
Escape the tedium of aimless wandering! With this foolproof, detailed map of the castle, you will know *exactly* where you are. Create new castles as much as you like—this map *always* works! Send $4 to The Originality Office, Suite 246, 251 Baldwin Ave., San Mateo, CA 94401.

THE ELEGANT GAME PORT

Extender. World famous Zero Insertion Force socket, woodgrain finish, non-slip base. Not an unfinished board. Postage paid $19.95. Rigney, P.O. Box 74, Chelan, WA 98816.

A-STAT 79.6C

Comprehensive statistical analysis system. Frequencies, Crosstabs, T-Tests, ANOVA, Correlations, Multiple Regression with Residuals, ... Reports, Transformations,



Applesoft ...
residents include 6½% sales ...
tions, 3710 Pacific Ave. #16, Venice, CA 90291.

BIG PROFITS TRADING stocks and stock options. Small investment with low risk. Sophisticated money management system. Write free details. Softsystems, 2005 W. Balboa Blvd., Suite 309F, Newport Beach, CA 92663.

LOW SOFTWARE PRICES!

Check out our fantastic prices. Write for our **FREE** price list! KERR SOFTWARE, Box 5301-ST, Long Beach, CA 90805. Or call (213) 428-8193.

FREE PASCAL NEWSLETTER

Tips, techniques & info on Apple Pascal. Also describes LOW-cost utilities & fun programs (with SOURCE). Kingdom Computer Concepts, POB182, St. Johnsbury Ctr., VT 05863.

"SCREENWRITER II MADE EASY"

Learn to use *Screenwriter II* in one sitting with my "plain English" booklet. Send $5.95 postpaid (check or M.O.) to J. Mandell, Dept. MA, P.O. Box 7063, Charlottesville, VA 22906.

APPLE GAME DISK EXCHANGE

Adventure games are great until solved. Arcade games can become stale. Now exchange your unwanted games for ones you would like to play. WRITE FOR INFORMATION or SEND your original manufacturer's game disk, documentation, a list of five games for us to make your exchange from, and $5 to:
National Home Computer Game Exchange
P.O. Box 14135, Columbus, Ohio 43214

ASSET MGMT. & SUMMARY II: NOW CORRELATES DATA AND PRINTS IRS FORM 4562. Complete, compact, flexible, yet easy to use Asset Management System with USER-controlled filing. Being used by CPA's and bus. Unlimited capacity. Does all types Depr., Amor, Inv. Cr., Disposition, Recap. Inv. Cr., Cap G/L, etc. Prints Sch, listings & summary for both Tax and Acctg purposes. $240. Low-cost updates provided if tax laws change. Now shipping 83 Edit. Apple II, 48K, IIe, or III (Inter), 2 3.3 disk drives, 80-column printer. SALBA SOFTWARE, 206 E. Cypress, Elmwood, IL 61529; (309) 742-8123.

TANK PLATOON!

Each player controls personnel carriers, tanks, infantry and leaders. Find the enemy; attack with artillery, machine guns, anti-tank missiles; dig a foxhole, run out of ammo; panic. 48K, DOS 3.3, $29. Dataworks, Box 236, Deep River, CT 06417.

REAL ESTATE SOFTWARE including property management, home purchase, loan amortization, ACRS/depreciation, tax deferred exchange, property sales analysis, loan sales/purchase, income property analysis, APR loan analysis, construction cost/profit, and property listings/comparables. Visa/MC/Amex (213) 372-9419. REALTY SOFTWARE COMPANY, 1116 8th Street, Manhattan Beach, CA 90266.

MANAGE YOUR DATA!!!

Organize your data to YOUR needs with this formatted data entry & report generation package. Menu/help-driven system allows custom screen/reports, or use included forms: mag. article, program lib., order entry, address/label. Supports search, sort & math operations. Tutorial manual. AppleSoft/ROM/48K/DOS 3.3. $49.95, check or M.O. to: R. Gooding, 19 Grist Mill Rd., Acton, MA 01720.

Softalk's classified advertising section offers a considerably less expensive way than display advertising to reach tens of thousands of Apple owners.

Classified advertising space is available at the rate of $10 per line for the first ten lines, with a five line minimum. Each line over ten lines is $25 per line. Ad copy should be received no later than the 10th of the second month prior to the cover date of the issue in which you want the ad to appear. Payment must accompany ad copy.

The publisher reserves the right to reject any advertising that he feels is not in keeping with the publication's standards.

Heads will be set in 10 point bold face, all capitals only. Italics are available for body text only; please underline the portions you would like italicized.

The body text of the ad will hold roughly 45 characters per line. Spaces between words are counted as one character. Heads will hold roughly 24 characters per line, with spaces between words counted as one character. Please indicate if you would like the head centered or run into the text.

Please write or call for additional information.

Softalk Classified Advertising
11160 McCormick
Box 60
North Hollywood, California 91603
Attention: Linda McGuire
213-980-5074

Lowest S/W & H/W Prices

No overhead produces the lowest possible prices ever. No corner is left uncut. Write today for free catalog. Softmail, 3061 Skipper St., San Diego, CA 92123.

SOFTWARE WANTED

Do you have a microcomputer software package, insurance related, that you have developed? May we help you market it? Call INSURANCE MICRO SOFTWARE, INC. (800) 835-0099 and let's discuss it.

APPLE III USERS, RUN MORE APPLE II JOYSTICK GAMES

Cable connecting ports A and B to joystick (Cursor III or TG) does the job. Also adds graphics. Send $60.00 to Cable, P.O. Box 316, Newton Highlands, MA 02161; (617) 965-3948.

APPLE III OWNERS

Join the original Apple IIIrs. Nationwide Apple III users' group. Software, hardware, VC templates, and newsletter. Write to Original Apple IIIrs, P.O. Box 813, San Francisco, CA 94101.

SMART MENU III

Use this highly interactive program with your Apple III Business Basic system to quickly run programs, list text files, and select any subdirectory on your disks. The program is very handy with floppies and truly works wonders with a hard disk. No longer will you need to remember (or type) obscure program pathnames. Send $21 to: The Third Source, 3649 Chevy Chase, Houston, TX 77019.

THE PREDICTOR

Predict winners for the U.S. Football League. By Jim Jasper, author of *Basic Betting* and *Sports Betting*. $39.95. Req. Apple II+/IIe, 1 disk drive. See your local Apple dealer or contact Pickam Software at 14411 Vanowen St., Ste. 209, Van Nuys, CA 91405; (213) 994-7944.

THE CHEAP ASSEMBLER

An APPLE assembler with: ten-command text editor, two-pass RAM/disk-based assembly, free field programming, interactive operation, tutorial manual, and demonstration routines on disk. Apple II+, 48K, DOS 3.3 required. Send $20 + $4 P&H to: Thunder Software, P.O. Box 31501, Houston, Texas 77231.

E-Z PILOT by Earl Keyser

Write your own school lessons or entertaining programs with the easiest of all computer languages—E-Z PILOT. Add colorful graphics, sound, and large, colored letters for emphasis. Anyone can do it! Requires 48K and DOS 3.3. Comes on a copyable diskette with a thorough manual for only $51.45. TECK Associates, Dept. S, P.O. Box 8732, White Bear Lake, MN 55110.

PRINTER FONT PREBOOT

Select any of the possible print options for your dot-matrix printer with a few keystrokes, then boot word processor, *PFS*, *VisiCalc*, etc. No more "PRINT CHR$" from BASIC. Comes on a fast-boot disk. Specify Epson, NEC/C.Itoh, or Apple. Send $15 check or MO postpaid to Steve Lemke, P.O. Box 30696, Santa Barbara, CA 93105.

STOCK DATA 1 YR ON DISK

Last 260+ trading days hi, lo, close, vol. on Apple DOS 3.3 disk. Send list of desired stocks and $5.50 plus $3.75 per stock to: Numerical Products, P.O. Box 6281, Torrance, CA 90504.

WORD PROCESSOR

Guess the price: upper/lower-case without hardware, word wrap, search/replace, preview of printed version, block text movement, page numbering, right justification, file linking, alternative (non-QWERTY) keyboards. $200? WRONG!! The Cheapsuit Word Processor is $30. Apple II, 48K, DOS 3.3. *Not* copy-protected! Cheapsuit Software, Dept. ST-2, Box 923, Iowa City, IA 52244.

OPTION INVESTING

Maximize your profit. APPLE, TRS-80, IBM. Compares investments, graphs. $125. M/C, Visa. Free brochure. OPTIONS-80, Box 471-R, Concord, MA 01742.

THE SOFTWARE HOUSE

Selling software for the Apple Computer: TG, Stoneware, Muse, On-Line, Sirius, SubLogic, Budge, Ashton-Tate, Broderbund, Hayes, Lotus, Silicon Valley, Videx, VisiCorp & many MORE!! Low Low Prices!! Free price sheet. M/C & Visa accepted. Write: 411 Rices Mill Rd., Wyncote, PA 19095; (215) 885-6151.

PRESCRIPTIVE MATH

A program that prescribes, revises, and records the changing needs of your child. Sample disk $10. Quantity discounts to schools.

Ed-U-Comp Corporation
246 South Ocean Avenue
Freeport, New York 11520

PERSIMMON SOFTWARE has an increasing variety of educational and home utility programs. None over $35; some only $7.50 on disk. SASE for catalog. 502 C. Savannah St., Greensboro, NC 27406. Author's inquiries welcome.

STOCK MARKET SOFTWARE

Increase your profits with OPTIONX, the ultimate option analysis program for the Apple computer. Price $145.00 including a 65-page manual. In CA add 6% tax. Send for free brochure describing OPTIONX and other sophisticated investment software.

CRAWFORD DATA SYSTEMS
350 N. Lantana, Box 3000-561C
Camarillo, CA 93011

APPLE KEYBOARD EXTENDER

Don't be tied down to your desk—put your keyboard where you want it. In minutes you can remove your Apple II keyboard and put it into a separate enclosure. Only tool required is Phillips screwdriver. Kit includes *6' retractable cable*, new enclosure, and all hardware. Introductory offer: $95 inc. tax and shipping. Send check or money order to: Darsey Co., 842 Lori Ave., Sunnyvale, CA 94086.

WARGAMERS!!

New assistance software to aid play of popular board wargames. Programs add fun & enjoyment; eliminates bookwork & die rolling. *Send now for a free catalog.* TRILOBYTE SOFTWARE, 7820 Exeter St., New Orleans, LA 70126; (504) 733-3195.

COMPUTER ROAD ATLAS

Start your next trip with a computer listing showing the best route. ROADSEARCH computes the shortest practical route and more. Prints the route with miles, time, and gallons. Contains 372 USA/Canada cities and 69,000 road miles. DOS 3.3. Ten day money-back guarantee. $34.95. (plus $1.50 S/H). Check/Visa/MC. Columbia Software, Box 2235M, Columbia, MD 21045; (301) 997-3100.

A-STAT 79.6C

Comprehensive statistical analysis system. Frequencies, Crosstabs, T-Tests, ANOVA, Correlations, Multiple Regression with Residuals, Data Entry, Sort, Reports, Transformations, Merging, Aggregation; Apple Plot, File Cabinet, and VisiCorp DIF Interfaces. For the Apple since 1979. $175.00 complete from: Rosen Grandon Associates, 7807 Whittier St., Tampa, FL 33617; (813) 985-4911.

EXTREMELY LOW PRICES

Authorized Rana drive dealer. Extensive Apple software. We pay shipping. Write for catalog.

The Computer Workshop
7603 E. Firestone Blvd., Suite 155
Downey, CA 90241
(213) 864-5564

SUPER SOFTWARE SAVINGS

Request free catalog. Wide variety of Apple programs at unbeatable prices. Take advantage of reduced phone rates after 5 p.m. Call for quote on software. Calls taken until 10 p.m. central time; or write.

APOGEE SOFTWARE
P.O. BOX 71
MORTON GROVE, IL 60053-0071
(312) 729-4821

THE GENERAL MANAGER

User group now forming. . . .
Paul Stocks
201 W. Collins
Oxnard, California 93030

INCREDIBLY LOW SOFTWARE PRICES!

Equally low prices for hardware also available. Send for catalog, $1, refundable on first order. Spin-tronics, 2490 Channing Way, Ste. 503, Berkeley, CA 94704; (415) 843-2743.

INSTANT SHIFT-KEY MOD

Tired of hitting ESC or a CTRL key to get a capital on your word processor or 80-column card? Ever thought of using the shift key for capitals? This easy-to-install modification comes with instructions. Send $15 check or MO postpaid to Steve Lemke, P.O. Box 30696, Santa Barbara, CA 93105.

MEDICATION SAFETY

Home Educational Programs answer hundreds of questions about how drugs can be taken safely and effectively. Input specific medications for potential interactions with food, alcohol, or other drugs. Order: "Consumer" Drug Watcher. Applesoft/disk/48K, $39.95 (CA res add 6%).

MEDICAL WATCH SOFTWARE
1620 Ensenada Dr., Modesto, California 95355

* GOLF HANDICAPPING * USPGA RULES

Complete handicap system ideal for pro shops, golf clubs, and personal use. Will calculate new handicap for an individual or the entire club. Program written with the assistance of a professional golfer. Complete with disk and instructions. Requires 48K Apple II or II Plus w/ 1 or 2 drives and printer. C C Computing, Box 432, Simsbury, CT 06070; (203) 658-7375.

AMATEUR RADIO OPERATORS

The RADCOM PLUS+ PACKAGE consists of a quality TU INTERFACE that installs in the Apple, connects to your rig, and includes SUPER-RATT, the most advanced, feature packed software ever developed for sending/receiving RTTY and MORSE CODE, with SECALL, RBBS, and much more. Detailed information from: ALEX M. MASSIMO—AF6W, 4041 41st. St., San Diego, CA 92105.

PSYCHOLOGICAL PSOFTWARE

Intimacy, The Art of Communications, Personality Profile, Assertiveness Training, Stress Management, and The Dream Machine. For more information about these outstanding and reasonably priced programs, write to: PSYCHOLOGICAL PSOFTWARE, 4757 Sun Valley Rd., Del Mar, CA 92014.

VIDEO POKER

Now turn your Apple into a video poker machine just like the ones in the casinos. The only difference is that with our version you'll never lose any money playing. Requires either an Apple II+ with 48K or an Apple IIe. $19.95. Jersey Shore Software, P.O. Box 751, Barnegat, NJ 08005.

WIZARDRY PLAYERS

Modify your characters with this utility program. Resurrect the dead or lost. Alter characteristics, spells, hit, experience & gold points. CHEAT! Make yourself a 1000th level super hero for Knight of Diamonds. Disk, 48K & Applesoft. Includes maps. $20.00. California residents include 6½% sales tax. ARS Publications, 3710 Pacific Ave. #16, Venice, CA 90291.

MINUTE MANUAL FOR APPLE WRITER II

Step-by-step instructions for basic & advanced procedures, & COMPLETE EPSON PRINTING COMMANDS. Perfect for HOME, SCHOOL, OFFICE. 100 pages. $5.95 + $1 S/H (5% MD tax). (301) 995-1166. AFFORDABLE PRINTERS—send for LOW price list. MinuteWare, P.O. Box 2392, Columbia, MD 21045.

HELLER SOFTWARE

AUDIO SPECTRUM DISPLAY . . . . $19.95. *See your music in a color graphics display!* Add $2 shipping. Pa. res. 6% tax. C-157, 4500 Londonderry Rd., Hbg., PA 17109; (717) 652-6655.

WE CAN'T AFFORD A BIG AD

Because we're keeping our overhead low so you'll get the *cheapest* software prices. Write for our free catalog: Alligator Enterprises, 1105 Alameda, Austin, TX 78704; or call (512) 443-2621.

3M SCOTCH DISKS...$21

Guaranteed 100% error-free, reinforced centers. Box of 10 for Apple . . . $21 + $2 shipping. Cactus Computer, 39 Carriage Pl., Urbana, IL 61801. 5% tax in IL. COD $3 extra. Order now!

ULTIMA II character editor and mapper. Boost characteristics, get possessions, map dungeons, towers, countryside. Disk 48K and Applesoft $18. Ont. add 7%. Amazing Software, 625 Wellington, London, Ont., Can., N6A 3R8.

CAN LEARNING AND FUN MIX?

See for yourself. Our new catalog clearly describes over 73 critically selected games and simulations that take a more creative and exciting approach to learning than more typical programs. And we stand behind that claim 100% with a *30-day-return guarantee.* Go beyond dull drill-and-practice and mindless eye/hand reflex-testers to find the innovative educational software you've been looking for. $1 covers first-class postage & handling. K-12 MicroMedia, P.O. Box 561, Valley Cottage, NY 10989.

ADVENTURERS BEWARE!

If WIZARDRY left you in awe, you will be aMAZEd by the KNIGHT OF DIAMONDS. Prepare for the unknown. Behold! The maps you need are available. Complete sets. $10 ppd. Ca. residents add 6½% sales tax. S&S Services, 2477 Kingfield Way, San Jose, CA 95124.

Pascal/CPM/DOS FILE TRANSFER UTILITIES

Six programs on a partitioned 5¼" diskette to transfer and reformat files among the CP/M, DOS, and Pascal operating systems. $45.

CLOCKWARE. Machine code subroutines for Pascal support of Prometheus and other clock/calendars. $25. MC/Visa/check or write for info: RCM SOFTWARE, 815 Friendship Drive, New Concord, OH 43762.

dBASE II USERS

Business software for under $50. Example Gen. Accounting Package includes check register, sales ledger, and financial statements $49.95. Payroll $49.95. Mail List $34.95. Dealer inquiries welcome. Call or write for prices. ARGONAUT DISTRIBUTING, 1104 Buchanan, Antioch, CA 94509; (415) 778-2595.

MACRO-TREND

New totally automated commodity trading system for the Apple II+. Generates entry points, exit points, reversals, and protective stops. Daily telephone data updates from Commodity Systems, Inc. One year program lease is $2,500. Test system with our demo diskette. To obtain demo diskette and complete documentation send $20 to: Steven Bollt, 7420 Westlake Terrace, Suite 1509, Bethesda, Maryland 20817.

FULL COLOR PRINTOUT!!

Printouts of HI-RES or LO-RES pictures, on white 20 lb. paper. Regular size (4.75" x 3.25"): $1.50; double size (9.5" x 6.5"): $3.00. Add $1.00 per disk shipping. Send pictures on disk (include your phone number) to: Douglas M. Smith, 135 N. Harvey, Oak Park, IL 60302; (312) 383-7209 (evenings).

exec Syntauri:

sounds of success

BY DENNIS BRISKIN

If Syntauri Corporation were just another high-tech company, its people would have some bright ideas, an Apple-related product, and the business skills to sell it at a profit. While that's all true, something beyond brains and greed comes through when Syntaurians talk about their work.

Call it belief, passion, juice, or commitment.

Listen to founder and president Ellen Lapham: "When I discovered the Syntauri synthesizer I said, 'Wow! This is what I want to do with my life.' It's fun because your heart and your guts are engaged, as well as your mind."

Cofounder and software consultant Scott Gibbs says, "Everyone connected with Syntauri is cause-oriented." (As in true believer.) "There is never enough time to do all we want. Both Ellen and I are that way." Congratulations are in order. Gibbs and Lapham were recently married.

The Original Syn. The attitude that this is more than just a job seems spread throughout the company.

"I don't work for money," says marketing director Ilana Wiedhopf. "I work for productivity and creativity. Since age twelve I wanted to give something to the world. When I realized God gave me life, I asked myself, 'What can I do with it?' There's a lot of that in Syntauri.

"I believe in the product," she continues, referring to the Apple-driven, digital music synthesizer called the alphaSyntauri. The name is a play on the double star Alpha Centauri (our closest stellar neighbor) in the constellation Centaurus.

"I can transfer how I feel about it to a broad number of people. That comes from inside. I have to preach and spread the word." She laughs self-consciously. "Yes, it's evangelism. Sometimes I get embarrassed. I start pounding."

Senior software writer Steve Leonard was among the first musicians to buy the alphaSyntauri. At Lapham's urging he came north from Los Angeles to the company's home in Palo Alto, California, to help take the synthesizer from "an experiment for hobbyists to a performance instrument for professional musicians.

"I love working in an art form with expressive people," Leonard says. "There's a fulfillment that is part of the magic of Syntauri. All the people in the company make it special. I feel lucky to have fallen into it."

Lenore Wolgelenter, who has just left Syntauri after more than three years with the company, was the first person Ellen Lapham hired. "I enjoy art in a multitude of forms," says Wolgelenter. "I'm an artistic *voyeur*. Syntauri gave me the outlet to be involved in a creative process.

"The product is sexy. It's fun."

Do You Believe in Syn? Sexy? Fun? Magic? True belief? Evangelism? What *are* these people talking about? Beyond the joy of music, they work, and play, in a new realm that treats sound as digitized data to be created, manipulated, stored, and retrieved lightning fast.

"This is an orchestra in a keyboard," says Gibbs, who is part of the team of programmer/musicians that create the alphaSyntauri's power and flexibility. Their top-of-the-line software lets you record one sound over another (called sound on sound) and play up to eight "instruments" at the same time (called split keyboard). "Put them together in the right way," Gibbs explains, "and you have a multitrack system."

Opposite page, top to bottom: Scott Gibbs, cofounder and software consultant; Ilana Wiedhopf, marketing director; Julie Roybal, operations director; and Robin Jigour, director of product development. This page: Ellen Lapham, president and cofounder.

Syntauri calls it *Metatrak*. With another program, called *Composer's Assistant*, the Apple prints out in score form what you have just composed.

Robin Jigour, Syntauri's director of product development, speaks proudly of what he helped create. "We have a sixteen-track recorder with our *Metatrak* software," he says. "It allows you to set up any track with any instrument, play it back at any volume, and listen to it while playing on top of it. When you're finished you're not stuck with the piece as you recorded it. You can go back and change any of the instruments any way you want. This versatility and flexibility of the system is what attracts most professional musicians."

Perhaps the best-known professional to endorse the alphaSyntauri is Herbie Hancock, jazz keyboardist and synthesist. Jigour calls him a musician's musician.

"If you can satisfy the professional market," Jigour adds, "you can derive from that and satisfy the personal market."

Syntauri's personal market includes music students in schools and colleges, as well as families at home. Developing strategies to reach those prospects is the job of recently hired marketing consultant David Archambault, who sees Syntauri facing a potential market as large as every home with a piano or organ.

"The desire for a musical instrument in the home hasn't changed," he says. "It's still there, as strong as ever. Music is the next major area of computer exploitation. We'll see an increasing demand for a computer-based musical instrument as a home-entertainment device."

Taking Aim. Archambault describes Syntauri's targeted home user with almost statistical precision.

"Our home market is the young male professional, aged thirty to forty-five, married with young kids. He's already bought himself an Apple but hasn't yet purchased an organ or piano. Now the kids are growing up and he wants to teach them music."

Buying a computer, disk drives, printer, video monitor, amplifier, and speakers, plus a keyboard, seems like a hefty investment for a child's musical experience. To the question, why not just get a piano, Lapham says: "What *is* a musical experience? I can show you with a couple of spoons and a pie tin that we can make terrific music together. Anything creative with sound can be called musical.

"Our focus is to *broaden* the musical experience. When I was a kid we had pianos, and all I cared about was exploring the sound capability with them, making thunderstorms or putting pencils and nails along the strings and listening to the weird effects. I also found that that wasn't done. Little girls don't go around making thunder sounds on pianos. Instead you study classical piano, usually from a little old piano teacher who turns you off.

"Everyone has had turn-off experiences in music because of the limitations of the instrument. An alphaSyntauri doesn't impose these structures because it's got so much programmability. You can go off with it and do what you want. You can explore. We are opening up what it is that's a musical experience."

The theme of opening to greater creative possibilities runs through the Syntauri group. "I think people are not satisfied with a closed set of capabilities," says Wiedhopf. Part of her job as marketing director is to know what people want.

"Syntauri gives them more variety than a piano. People say, 'Why should I buy a piano when this can play a piano and the flute and make a wa-wa sound and I can design any sound I want and compose?' People want music in their lives; the more creative they can be with it, the more they like it."

Garage Sale. While on the subject of creativity, remember the story of how a couple of young guys at Hewlett-Packard built a microcomputer in their garage and offered it to their corporate employer? And the employer didn't want it so they went out and started their own company?

Syntauri's company history includes an analogue of that story. It begins with Charlie Kellner, a programmer/musician in Oregon. Kellner is fuzzy on the dates, but sometime in 1975 or 1976 he and a couple of friends in Salem built a synthesizer from an old organ keyboard, using hardware and logic only.

"I have always underestimated the power of software," he says.

In January 1979, Kellner came south to California to write educational software. He couldn't afford to move his piano with him and wanted a substitute. While working in Silicon Valley he learned of Mountain Hardware (later to become Mountain Computer), which produced a music synthesizer card. With an organ keyboard, an interface card, several thousand connections, and an Apple computer, he had synthesized music again.

"The first software was primitive," Kellner says. "I was doing everything the hard way. But it worked out to a design similar to what Syntauri is using now."

Kellner first offered his computer-driven digital music synthesizer, which he named alphaSyntauri, to his corporate employer, but they turned it down. The company? Apple Computer.

"It would have been a good product for the company," says Kellner, who still does programming and research for Apple. "After a long decision-making process, though, they decided they weren't in the music business."

So Kellner introduced it on his own. He first showed the alphaSyntauri synthesizer at the West Coast Computer Faire in April 1979. Later that year it appeared at shows in Chicago (Consumer Electronics Show) and New York (National Computer Conference). "It was the hit of the show," he adds matter-of-factly. At the New York show Ellen Lapham approached him about taking the product to market.

"I had to decide whether to do it myself or go elsewhere," Kellner says. "There were too many possibilities at Apple for me to leave. And I didn't want to spend my life building synthesizers. Ellen and Scott convinced me that they had the talent and the know-how to make it successful. They were my best opportunity."

Says Scott Gibbs: "Ellen was the person who said, 'This thing will go.' She always believed that it would take off. It took more convincing for me."

Syncing Up. Lapham and Gibbs bought the rights to Kellner's synthesizer and the names alphaSyntauri and Syntauri Corporation, which they established in California. Kellner says their first agreement, written

on a napkin, was informal and based on mutual trust.

Although his original idea was to sell Lapham and Gibbs all rights to the product, Kellner found himself involved in the early work of refining the synthesizer. At that time he took royalties in lieu of wages. Now the royalties have stopped.

"I have no regrets," says Kellner, who is no longer involved with Syntauri. "Ellen and Scott have done an incredible job with it. They took the synthesizer farther than I could have.

"The people at Syntauri are doing a very professional job," he adds. "The new software has features that can't be matched even by a $20,000 synthesizer. It's way beyond my original concept. It's becoming 90 percent of what anyone would want to do with a synthesizer."

Kellner also says the alpha's progress is a matter of musicianship. "Robin and Scott are both performers. They can add to software development what I couldn't. They know what to do with a synthesizer. It's the kind of experience programmers don't have. That's what's making Syntauri so successful."

How successful has Syntauri become since it first started shipping product in December 1980? Lapham and her staff won't say exactly, but she admits to 1982 sales in the $1 million to $2 million range. Although the company has factors in its favor (an innovative, artistic, timely, software-based product promoted to diverse markets by bright, high-energy people), it is still a small start-up (only fourteen employees) facing a number of challenges.

Variable Barriers. New products, no matter how good, require educating both consumers and dealers, as well as overcoming the barriers of resistance. According to Wiedhopf, the barriers vary with the market.

"In the professional market," she says, "they know a lot about music, but they don't feel comfortable with the computer. In the education market, they love the fact that it has a computer. They love *everything* it has. The barrier is budget. In the home market, it's also economics, although as we ride the trend of increasing computer awareness we will do fine."

Syntauri faces a more serious challenge in establishing a high-quality dealer network. They have only seventy-five U.S. dealers, with another fifteen overseas. Their goal is to double that number in 1983 and eventually peak at four hundred.

"The biggest problem we have now is finding dealers sophisticated enough to understand the computers and the music aspects," says Archambault. Syntauri began by offering its product at computer stores.

"The computer people are still learning their *own* business," he says, explaining why the alphaSyntauri sells better in music stores. "They aren't sophisticated enough yet to see music as the next major area. Some do, and as our dealers they do very well."

Musical Sales. Since it now sees its product as fundamentally musical, Syntauri has adjusted its sales strategy toward various music outlets. While the majority of its dealers are still computer stores, the highest sales volumes come from music dealers. These include the now-struggling piano and organ store (Archambault says organ sales in North America fell 42 percent in 1982), the high-volume music store, and the average pro-sound store.

"Music is the most personal home entertainment medium," he says. "The prime dealer increasingly is the store with someone on staff who has a personal interest in the product, someone who has taught himself about computers and can see the way the business is going. They spend the time with the customer. People still need to have it demonstrated to them, even though they are accepting the technology."

Syntauri's sales growth may have to be pushed through the bottleneck of its ability to expand a supportable dealer network. "Much the way the dealer has to give personal support to the end user, we have to give personal support to the dealers," Archambault explains. Syntauri's dealer support also includes cooperative advertising, local seminars, and new point-of-purchase materials.

While the keyboard synthesizer and its software have been well-reviewed, Syntauri people talk honestly about the need for product improvements.

Operations director Julie Roybal says their biggest manufacturing problem is a return rate of 5 percent on the music synthesizer boards,

which are made for Syntauri by Mountain Computer Company of Scotts Valley, California.

"It's a complex unit and the quality control is not very good," she says. Keyboards, pc boards, and cables are subcontracted for manufacture to a local electronics company. Roybal estimates the return rate at 1 per 100. "I would like to see that improved to 1 in 500," she says, although she adds that the quality improvement might not be worth the extra time and expense.

Scott Gibbs has the precision of a mathematician (which he is) and the perfectionism of a classically trained musician (which he also is). He is not satisfied with the sounds from the alphaSyntauri.

"I'm a purist in that sense," he says. "The sounds simply aren't good enough. The organ sounds are superb. Steve Leonard did those. The trumpet sounds are not good enough. The strings are definitely not good enough. A little postprocessing has to be done to make them sound really snappy.

"I think the boards in this system, without any more money being poured into them except for software development, can make fantastic sounds. There are people in the company who don't agree with me, but I'm sure of it. I'm working on getting the last ounce of quality out of these boards."

A Delicate Balance. For Robin Jigour and his team of software writers, the problem is to keep a balance in meeting the needs of Syntauri's three basic markets: professionals, students, and home users.

"We could go into one market and hit every key feature and need," Jigour says, "but that could mean abandoning others. The real challenge is satisfying all of them."

Jigour says the company also faces the problem of finding qualified engineers for its special product. "We need engineers with computer hardware and software experience and a strong background in music synthesis or music theory. It's a very select breed.

"Now it's much easier because Syntauri is so well-known for what it does that people have been coming to us. In the earlier days we had to do the hunting. We were lucky to get a guy like Steve Leonard."

Ellen Lapham and her team candidly admit to having made mistakes and learned along the way.

"I would have gotten more capital in sooner," says Lapham. "And I should have gotten my board of directors together sooner. Now I have a board that is absolutely colossal. I wish I had this crew a year ago."

Heavy Hitters. Besides Lapham as chair, the Syntauri board includes Julie Roybal's husband, Phil Roybal, manager of European marketing for Apple Computer; Tom Skornia, venture capitalist and former vice president of and general counsel for Advanced Micro Devices; Herbie Hancock; and Sam Bernstein, a former vice president of sales at Commodore Computer.

"We should have been more efficient with our promotional dollars," Lapham adds. "And I would have loved to have hired a few people sooner, so we would have a broader software product line today. That's the dilemma of a start-up company. You have to focus."

"I would say the major mistake was thinking of the synthesizer as a set of features rather than a set of benefits," says Wiedhopf. " 'The programs are written in Applesoft,' for example.

"That type of thinking is related to our going to computer dealers first rather than music dealers. It's a common error in this industry. But one day you wake up and you hear consumers saying, 'What do you *mean*? I don't care about that.' Anyone marketing a high-tech product to consumers should forget what is inside or how it works. Tell them *why* they should buy it. What it will do for them."

Scott Gibbs ponders the question of mistakes. "Did I make any mistakes?" he asks himself in a whisper. "No, I don't think I made any mistakes."

What did he learn?

"Be prepared to kill yourself in the process. I have given it my all, although I have a lot more left." He describes writing software as harrowing. "The concentration is the most exhausting thing I have ever experienced. The only way I can finish something is by working fourteen hours straight. Then I collapse for a day. I'm always thinking about something. It's like an irritation until the problem is solved."

Lapham is clear about what she has learned. "Understand what cash flow really means. Cash flow is the bottom line for any start-up company. I don't have to worry about return on investment yet, but I sure have to worry about whether I can pay my bills. To have controls in place for cash-flow management is very important.

"You can't afford mistakes or fuzzy thinking in a start-up," she says, "especially in high-tech types where the technology is moving—or you are moving it—faster than anyone else. Managing in an environment of constant newness means you have to hire people who are good at self-managing.

"The hardest thing in dealing with a start-up is the people part. With technology, if you throw enough time and money at a problem you will probably solve it. But getting the people who can pull that off, providing the internal working relationships that have them feeling satisfied and happy and feeling good about being a team, that's the trick.

"It's important that lines of communication are established and maintained, that there's trust between people and a certain amount of professional respect. You get that by having communication. My job is to be sensitive to when it breaks down and provide the structure for communication to occur."

The Lapham Guide to Success. "Hire the absolutely best people you can early on in the game; people who are not only motivated but who are as experienced as you can get. You don't necessarily have to get the heavyweights in the industry. You need people who have functional skills and a certain amount of self-confidence. I have a team of terrific people who are highly motivated, with high energy, who are willing to take the personal risk of tackling something they haven't done before."

Lapham points to Lenore Wolgelenter, her first employee, as one who had the courage to risk doing what is new and different.

"She's not a musician, but she went out and did the demos at all the Applefests. She's not a computer programmer, never held a soldering iron, no 'techy' skills whatsoever. But good 'people' skills and high energy.

"We have what everyone feels is a high-tech product. But it's a people product. It's as high-tech as a piano or as low-tech as a kazoo, depending on what you want to do with it."

The Syntauri people all seem to love putting in long hours. Typically, they work ten to twelve hours per day, six days a week. Some longer.

"I'm always thinking about something Syntauri-related," says Gibbs. "It never stops, except for short vacations or when I get sick."

"It's with me every day," Steve Leonard says. "Even when I'm watching TV. That's how I problem-solve. It's the nature of the industry. Also, there's competition. It's what you have to do to make it the best."

Wiedhopf says she doesn't count the hours. "My dream is to be able to work forty a week. Sometimes it's eighteen hours per day. Sometimes when I burn out it can be five." To deal with the stress she swims an hour a day, spends two or three hours per day with her teen-age daughter, or turns to music, preferably Bach fugues on violin. When it gets really bad she turns to country music.

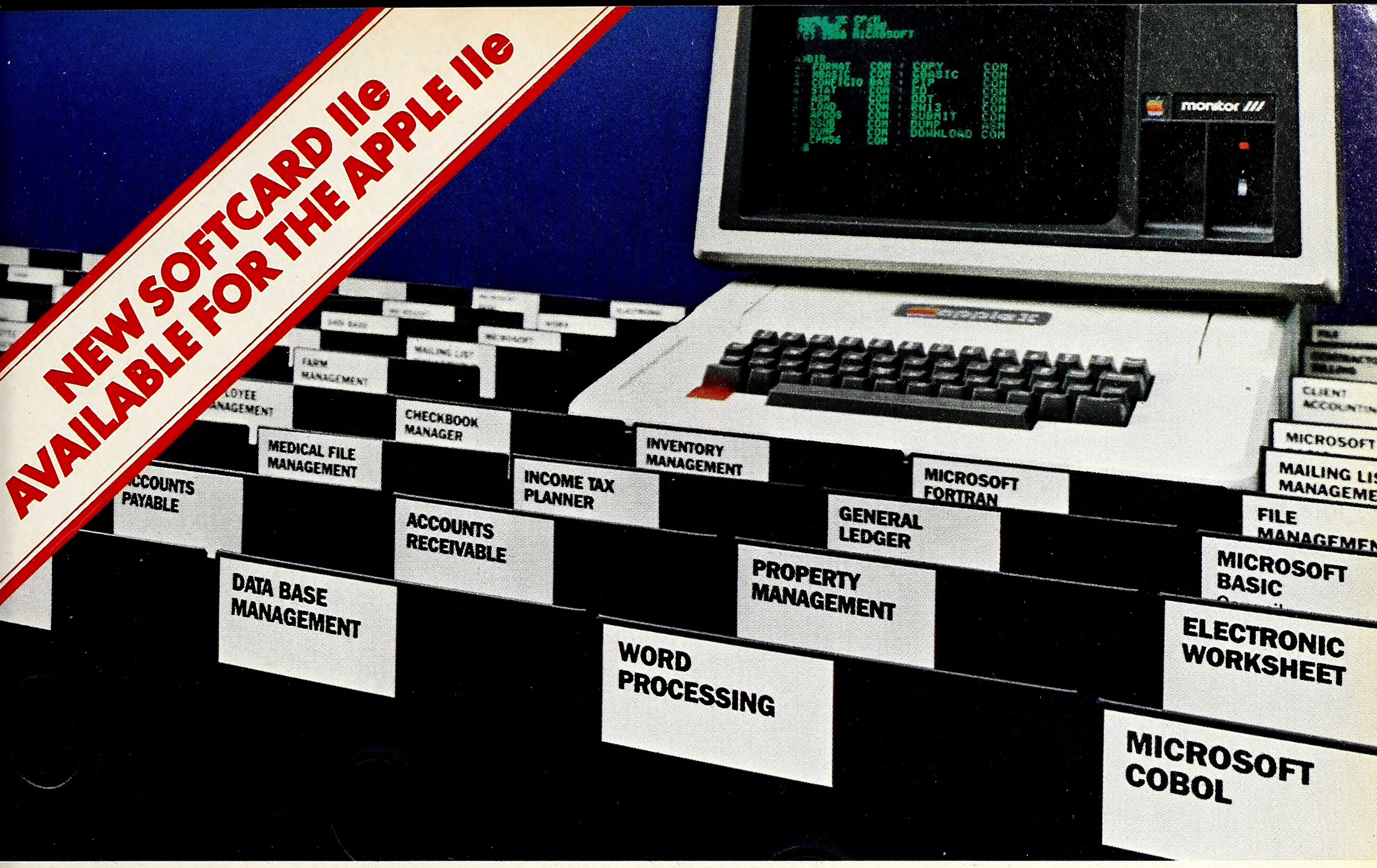
Lapham and Julie Roybal deal with the stress through the relaxation of massage. "Your body and your brain are tied together," Lapham says. "The massage gives me a chance to concentrate on relaxing my body. Countering stress also means putting things in perspective and dealing with them one by one."

Roybal puts the long, hard hours in a different perspective. "It's not the number of hours you spend. It's how you feel about what you're doing. This has been the most dynamic, eye-opening, emotionally demanding experience of my life. I have become much more attuned to my own strengths and weaknesses, and much more sensitive to others."

Command Performance. It all comes back to those intangibles; what Lapham calls the "psychic satisfactions" of managing a small, innovative company.

"I have a gut belief in what we're doing," Lapham says. "I can't relate on a day-to-day basis with what Herbie Hancock does, because I don't have Herbie's musical skills—his 'chops,' as they say.

"But I can relate to every kid who is taking music lessons. All of us are musicians at heart. The synthesizer lets me be a kid with my own product. I think that's great. It's a lot more fun than business-systems software." ⌧

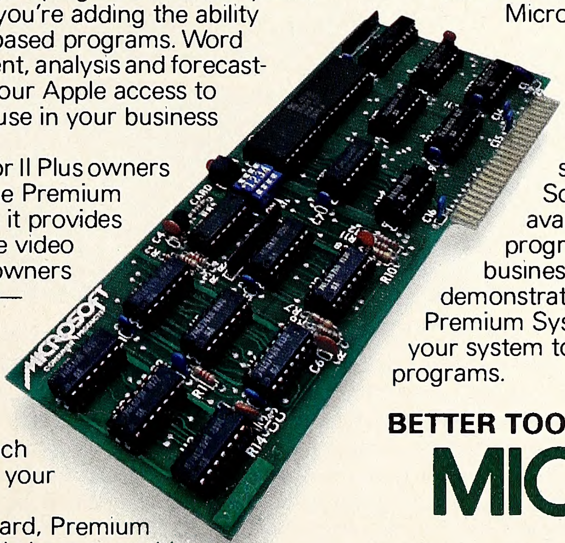


Microsoft SoftCard systems introduce your Apple to thousands of new programs.

More powerful Apples. When you plug a SoftCard™ system into the Apple® II, II Plus, or IIe, you're adding the ability to run thousands of CP/M®-80-based programs. Word processing, data-base management, analysis and forecasting programs—SoftCard gives your Apple access to thousands of software tools for use in your business or home.

Premium capabilities. Apple II or II Plus owners who want even more can add the Premium System. In addition to CP/M-80, it provides 80-column upper-and-lower case video and a 16K RAMCard. Apple IIe owners can have all this on a single card—the Premium SoftCard IIe.

Two computers in one. Any of these systems turns your Apple into two computers. One that runs Apple software and another that runs CP/M-80. Which means you'll double the utility of your computer.



Complete solutions. The SoftCard, Premium System, and Premium SoftCard IIe have everything you need. Easy-to-install circuit boards. The CP/M-80 operating system. Microsoft® BASIC. And the utilities you need to manage CP/M-80 files.

Why Microsoft? Microsoft was the first microcomputer software manufacturer. The very first. Today, Microsoft software is running on well over a million computers, worldwide. With a reputation for dependability and consistent product enhancement.

Ask your dealer. Ask about the superior application programs the SoftCard and Premium System make available to your Apple—high quality programs for almost every area of home, business and professional use. Then ask for a demonstration of the complete SoftCard, Premium System, or SoftCard IIe. And introduce your system to some of those thousands of new programs.

BETTER TOOLS FOR MICROCOMPUTERS

MICROSOFT™

MICROSOFT CORPORATION
10700 NORTHUP WAY
BELLEVUE, WASHINGTON 98004

Get through the best of times and the worst of times in no time.

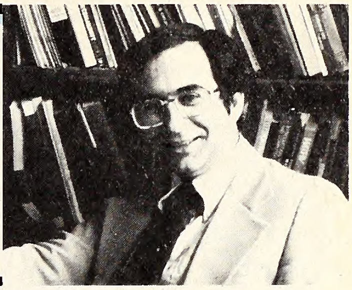
With the BPI Speed Reading system, you can learn to read 1,000 words per minute in 9 easy lessons. Or learn to speed scan up to 2,000 wpm. It's all possible with the BPI Speed Reading system, the most sophisticated speed reading method available anywhere. For use with Apple II hardware, BPI software allows you to learn speed reading at your own pace on your own home computer. Designed by university educators and expert programmers, the BPI program can accommodate as many as 20 readers, with record keeping capabilities that can display each reader's progress in graph or print form for up to 50 previous sessions. Featured are an optional comprehension check and effectiveness measure. And an exclusive feature allows you to select your own reading material. BPI even offers a disk library of various subjects, available at a nominal charge. The BPI Speed Reading system. At home. On your own time. Because at 1,000 wpm, you could have been through this ad a long time ago.

BPI SYSTEMS™

3423 GUADALUPE / AUSTIN, TX 78705 / 512-454-2801

Mind Your Business

BY PETER OLIVIERI



May is one of the nicest months of the year. Here's hoping that as you read this you're relaxing somewhere and enjoying some fine weather. After all, what could be more pleasant than soaking up some sun, sipping a refreshing drink, and reading the most recent issue?

Looks Are Everything. When it comes to printed output, looks are, indeed, everything. Often, a report with quality content fails to get the attention it deserves because it doesn't "look professional." And often, a report of lesser quality wins an audience solely as a result of its attractive appearance. It stands to reason, then, that whatever you can do to enhance the appearance of your material will probably increase its chances of being read and taken seriously.

Such things as type style, paper quality, and how the document is formatted are important factors in how "professional" a document looks. A good-quality daisy-wheel printer can give you very professional-looking output. In addition, by changing the print wheel, you can get various different type styles. (A similar effect can be achieved using a dot-matrix printer if that printer has certain options. For example, many Epson users have said that printing in emphasized, double-strike mode results in near-letter-quality printing. Of course, there is some sacrifice; printing speed is slowed considerably, for one thing.)

It's also possible to get different typefaces by means of software. This approach is often a bit less expensive than purchasing a letter-quality printer. With that in mind, look now at *Type Faces,* a new program from Alpha Software Corporation that allows users to incorporate some interesting options into their printed material. *Type Faces* is a powerful, easy-to-use program for printing in different fonts. It can be used to prepare business presentations, professional-quality invitations, letterhead stationery, special announcements, personalized cards, and attractive signs.

Type Faces requires 64K of memory, two disk drives, and a printer. The printers it supports include the IDS 460/Prism 80, IDS 560/Prism 132, Epson 80/100, Trendcom 200, and the Silentype. (The Epson and IDS printers must have the graphics option.) Note that *Type Faces* supports regular parallel and serial printer cards but is not compatible with "smart" printer cards such as the Grappler and Microbuffer.

The intent of this package is to allow the user to print text in a variety of fonts. The actual characters printed are about five-sixteenths of an inch high (about the equivalent of two lines of single-spaced, normal text). Thus, *Type Faces* has not been designed as a multiple font program for your word processing applications but more as a means of creating attractive signs and announcements. The results are indeed attractive. Not only are the characters clearly formed and easy to read, but the variety of characters available is quite appealing. Sixteen different type styles available are shown on this page.

The user guide is quite clear, and in no time at all you can learn how to print out text in a variety of formats. Besides being able to enter text,

This is Italic Complex
Τηισ ισ Γρεεκ Χομπλεξ
This is Script Complex
This is Roman Triplex
This is Italic Triplex
This is Italian Gothic
This is English Gothic
This is Roman Indexical
This is Italic Indexical
Τηισ ισ Γρεεκ Ινδεξιχαλ
This is Roman Simplex
Τηισ ισ Γρεεκ Σιμπλεξ
This is Script Simplex
This is Roman Duplex
This is Roman Complex

save it, and print it out in the font of your choice, you can use *Type Faces* to list all the file names in your directory, list your original file on the printer, delete a file from disk, pack the data on disk in order to take advantage of any unused space, format a blank disk, move backup copies of data disks, copy a DOS text file, and copy a CP/M text file.

The program can read files from those word processors that store data in DOS text files and it can also read files generated via *WordStar* and other CP/M word processing programs. In addition, any file created by a Pascal editor can be copied onto the *Type Faces* data disk, and later on such a file can be typed out in any face you choose.

Left Face. Besides printing the contents of your text file, you can embed format commands within the text itself in order to achieve format variations. Here are some examples: .ce—this will center the next line; .ul—this will underline the next line; and .sp 3—this will skip three lines.

The print commands are reasonably extensive and include spacing commands (spaces between lines, line heights), print enhancements (underlining, changing character sets), page organization commands (headers, footers, and page lengths), and line organization commands (justification, indenting, and so on). *Type Faces* also offers a "comprehensive" mode for the editing of files. In this mode, characters can be searched for, lines inserted and deleted, text saved, characters substituted, and so on.

For applications requiring a poster, sign, or announcement, *Type Faces* may be just right. Please note that you'll need 64K, that the product supports only certain printers, and that it's not a replacement for a letter-quality muliple-font printer for word processing.

Quick File III. Database management is an application area of great interest to most businesses. DBMS packages range from the simple to use (such as *PFS:File*) to the more complex, full-function database (such as *DB Master*). One of the database packages created for the Apple III is *Quick File III* from Apple Computer. To use the program, you need an Apple III with a second disk drive or a ProFile hard disk. A printer, while not required, is highly recommended.

Quick File III was designed for somewhat less complex applications than a program such as *DB Master* might be used for. The name is an appropriate one—*Quick File III* can be learned quickly and retrieves data quickly. To most users, these are important features. Some possible applications for this package include keeping and retrieving data on expense accounts, travel itineraries, appointments, business contacts, personnel reports, telephone lists, and so forth.

This database management system allows the user to show the information in the database on-screen in the order and format that is most appropriate to a particular application. There's a good deal of flexibility in this regard. Information can be sorted in any order, and reports can be easily prepared. Searches through the database can be made to select only information that meets certain user-defined conditions.

One of *Quick File III*'s nice features is that it takes advantage of the Apple III's special characteristics. Thus, let's suppose that the records in your database have been called to the screen. To look through the records, you simply use the arrow keys (to move left and right one character at a time), the up and down arrows (to move up and down a line), and the tab key to move to the next piece of information (that is, to the next field). Combining the open-Apple key with these keys enables the user to move backward by field, move to the beginning or the end of a file, and up or down the screen page by page. There's also a help screen that allows you to seek clarification if you don't understand something; this feature is another indication of a well-designed system.

For example, suppose you're looking at the names of the departments in which all the people in your database are employed and you wish to look at all the records of the people in a particular department. Let's further suppose that you're not sure how to use the "find" feature in *Quick File III.* Your first step would be to invoke the "help" option by pressing the open-Apple key and the letter H at the same time. You'd then be presented with a series of help screens, one of which deals with the find command. You would see the following on-screen summary (all of these commands are preceded by pressing the open-Apple key):

```
A Arrange (sort) on a category.
B Go to beginning of file.
D Delete records.
E Go to end of file.
F Find all records that contain. . . .
I Insert new records before the current record.
K Make a duplicate copy of the current record.
```

After choosing F (by pressing open-Apple F), you'd be asked to enter the information you were searching for. It would then be brought to the screen for display.

Quick File III's screen displays are designed to be as informative as possible. You always know where you are in *Quick File III,* what you are currently doing, and how to get back to the main menu; this information is displayed directly on-screen.

Sorting is also quick and easy. You simply place the cursor on the field that you wish to have sorted and press open-Apple A (for arrange). The arrangement order can be A to Z, Z to A, 0 to 9, or 9 to 0.

One of the unique features of this package is that it offers you two record layouts. (A *record* is a collection of information about one member of your database. One piece of information in a record is called a *field.*) These two record layouts are called the "multiple record layout" and the "single record layout."

As the name implies, the single record layout displays all of the fields of information about one member of your database. The multiple record layout displays the data for several records. Users may choose to have this layout display only the records they're interested in, rather than all of the records in the database. This is extremely handy, since it allows users to create different data-collection forms, different record-update forms, or different screen displays according to the particular needs of the time. It is also quite easy to rearrange a record format into a different order or to specify a different selection of fields.

Quick File III also has a report option. The two report styles available are "tables style" (in which information is shown in rows and columns and only one record can appear on one line) and "labels style" (in which information is displayed in a vertical format and data from a sin-

Smart Customers

SMART CUSTOMERS HELPED DESIGN OUR NEWEST PROGRAM. IT SORTS, LISTS, SCANS, TOTALS, SEARCHES AND DOES AMAZING TRICKS!

The story behind our new program. No sooner had we introduced our first checkbook accounting system last year, than customers began calling and writing. They said they loved the program, and then began pestering us to produce a new program.

For example, Leo de Gar Kulka of Foster City, California wrote saying: "I've tried five checkbook balancers for the Apple. Yours is best. But you need to take one more step: Make it do bookkeeping. Right now you almost have a business ledger with as many categories as you give".

Then there was the letter from Mike Merren, a neurologist in San Antonio, Texas: "I find it an excellent program and use it for my wife's small business", he wrote, adding that it could become a "cash basis" bookkeeper if it sorted cleared and uncleared checks together.

Kellar Watson, a retired druggist from Orange, California explained that he manages his children's trusts: "I use your checkbook accounting program and now find it would be almost impossible to live without it". He went on to suggest that any new program allow sorts by fiscal year.

A general contractor in Portland, Oregon suggested our new program include a sort-by-payee feature: "That way I could do job costing and file IRS 1099s directly from my checkbook".

Doyle Lee, a computer store owner said, "For small business, your checkbook accounting system is usually better than a full-blown accounting package. But your new program should include running sub-totals of category codes. With those, anyone could create customized reports".

A local CPA suggested we add new audit trail features: "Make your new program sort by amount, entry order, and uncleared checks. Make it print a "trial reconciliation" to help balance the bank statement.

He also suggested our new program create a cancelled check file: "Which leaves checks in the order which you get them back from the bank, but allows a user to find disputed checks fast".

Money Street™. It's totally new. So, we went to work and created a brand new program. It's called Money Street, and it's a checkbook accounting system like none you've ever seen. We did all the things our customers asked for, and added over 25 other features.

For example, you can sit before your computer and scan an electronic file of checks at the rate of 500 per minute. You can enter a check and instantly see a "category year-to-date total". You can find a cancelled check in a file of 1200 checks in 90 seconds.

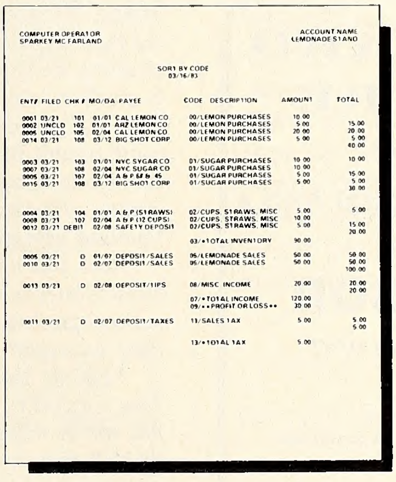
Money Street prints reconciliation summary of each balancing session. It prints a complete checkbook history showing every entry, plus the running balance each step of the way. Money Street lets you edit any entry, anytime; before, during, and after entering it into the system.

It can sort checks into tax deductible categories. It manages unlimited checking accounts; one program does it all. It's documentation includes tutorial, on-screen demo, program map, samples of all important screens, plus samples of all reports.

It allows you to code deposits, too. That means you can track income, sales, and taxable items. Mix checks and deposits into one category, and create net amounts.

Academy Award. If programmers got Academy Awards, then Don Hill deserves it for Money Street. Don zips you around the program in seconds. His error trapping simply won't allow you to make invalid entries. His prompts always tell you where you are, what you do, and how you exit. There are no dead ends, no traps, and no fancy set-ups.

Best of all, his 15 ready-to-print reports require only four key presses. He included a place for your name, your account name, and today's date. He even created a special way to "talk" to your printer. For example, you can tell an Epson to print emphasized type faces. The reports look like this:



COMPUTER OPERATOR
SPARKEY MC FARLAND

ACCOUNT NAME
LEMONADE STAND

SORT BY CODE
03/16/83

ENT#	FILED	CHK #	MO/DA	PAYEE	CODE DESCRIPTION	AMOUNT	TOTAL
0001	03/21	101	01/01	CAL LEMON CO	00/LEMON PURCHASES	10.00	
0002	UNCLD	102	01/01	ARZ LEMON CO	00/LEMON PURCHASES	5.00	15.00
0005	UNCLD	105	02/04	CAL LEMON CO	00/LEMON PURCHASES	20.00	20.00
0014	03/21	108	03/12	BIG SHOT CORP	00/LEMON PURCHASES	5.00	5.00
							40.00
0003	03/21	103	01/01	NYC SYGAR CO	01/SUGAR PURCHASES	10.00	10.00
0007	03/21	106	02/04	NYC SUGAR CO	01/SUGAR PURCHASES	10.00	
0006	03/21	107	02/04	A & P 6# & .45	01/SUGAR PURCHASES	5.00	15.00
0015	03/21	108	03/12	BIG SHOT CORP	01/SUGAR PURCHASES	5.00	5.00
							30.00
0004	03/21	104	01/01	A & P (STRAWS)	02/CUPS, STRAWS, MISC	5.00	5.00
0008	03/21	107	02/04	A & P (12 CUPS)	02/CUPS, STRAWS, MISC	10.00	
0012	03/21	DEBIT	02/08	SAFETY DEPOSIT	02/CUPS, STRAWS, MISC	5.00	15.00
							20.00
					03/*TOTAL INVENTORY	90.00	
0006	03/21	D	01/07	DEPOSIT/SALES	05/LEMONADE SALES	50.00	50.00
0010	03/21	D	02/07	DEPOSIT/SALES	05/LEMONADE SALES	50.00	50.00
							100.00
0013	03/21	D	02/08	DEPOSIT/TIPS	06/MISC. INCOME	20.00	20.00
							20.00
					07/*TOTAL INCOME	120.00	
					09/**PROFIT OR LOSS**	30.00	
0011	03/21	D	02/07	DEPOSIT/TAXES	11/SALES TAX	5.00	5.00
							5.00
					13/*TOTAL TAX	5.00	

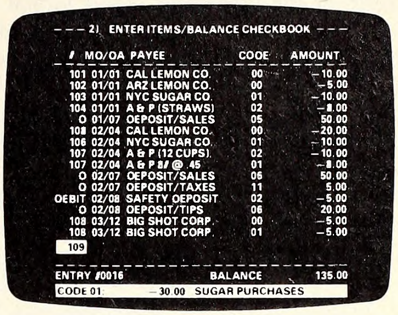
15 ready-to-print reports.

1. Month
2. Amount
3. Payee
4. Category
5. Check number
6. Deposits
7. Code dictionary
8. Single month
9. Single category
10. Entry order
11. Statement date
12. Uncleared check
13. Uncleared deposit
14. Code dictionary
15. Category totals

One $99.95 package does it all. That's because one Money Street package handles unlimited checking accounts, it works for calendar or fiscal years too. At years end, the program rolls over into the new year in less than two minutes.

Better yet, all your business and tax data is confidential. Your personal financial life is your own. It's a private matter between you and your computer.

How it works. On your computer screen, you create a facsimile of your checkbook. You see 17 items per screen, and can scroll for more. It looks like this:



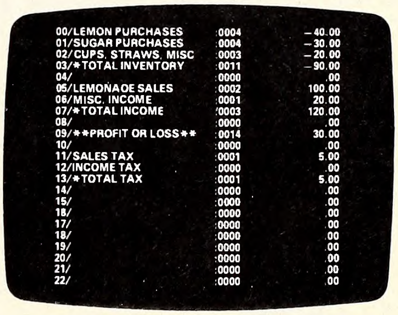
--- 2) ENTER ITEMS/BALANCE CHECKBOOK ---

#	MO/DA	PAYEE	CODE	AMOUNT
101	01/01	CAL LEMON CO.	00	−10.00
102	01/01	ARZ LEMON CO.	00	−5.00
103	01/01	NYC SUGAR CO.	01	−10.00
104	01/01	A & P (STRAWS)	02	−8.00
D	01/07	DEPOSIT/SALES	05	50.00
105	02/04	CAL LEMON CO.	00	−20.00
106	02/04	NYC SUGAR CO.	01	−10.00
107	02/04	A & P (12 CUPS)	02	−10.00
107	02/04	A & P 8/ @ .45	01	−8.00
D	02/07	DEPOSIT/SALES	06	50.00
D	02/07	DEPOSIT/TAXES	11	5.00
DEBIT	02/08	SAFETY DEPOSIT	02	−5.00
D	02/08	DEPOSIT/TIPS	06	20.00
108	03/12	BIG SHOT CORP.	00	−5.00
108	03/12	BIG SHOT CORP.	01	−5.00
109				

ENTRY #0016 BALANCE 135.00
CODE 01: −30.00 SUGAR PURCHASES

Money Street's most amazing feature is its "real time" data bank. It accumulates year-to-date totals for each of the 100 categories. You see these totals instantly. Just enter a check, and look at the bottom of the screen. The year-to-date total will flash into view with each new entry. So, you get constant up-dates as you enter data!

Here's how you set up your code categories. Press Ctrl-O and a screen-behind-the-screen instantly appears. Then just type in category names. You can add, change, or delete category names anytime without affecting data. It looks like this:



00/LEMON PURCHASES	:0004	−40.00
01/SUGAR PURCHASES	:0004	−30.00
02/CUPS, STRAWS, MISC	:0003	−20.00
03/*TOTAL INVENTORY	:0011	−90.00
04/	:0000	.00
05/LEMONADE SALES	:0002	100.00
06/MISC. INCOME	:0001	20.00
07/*TOTAL INCOME	:0003	120.00
08/	:0000	.00
09/**PROFIT OR LOSS**	:0014	30.00
10/	:0000	.00
11/SALES TAX	:0001	5.00
12/INCOME TAX	:0000	.00
13/*TOTAL TAX	:0001	5.00
14/	:0000	.00
15/	:0000	.00
16/	:0000	.00
17/	:0000	.00
18/	:0000	.00
19/	:0000	.00
20/	:0000	.00
21/	:0000	.00
22/	:0000	.00

Look again at the Dictionary, and you'll see all the year-to-date totals, plus a count of items. The count is very handy for checking on items requiring monthly payments.

Better yet, the Dictionary will let you sub-total groups of category codes. In the example above, code three totals all the categories above it. To get group totals, just start any category with an asterisk. Two asterisk creates grand totals.

Includes tutorial, map, and reports. Money Street includes Program Map, complete documentation, and samples of all reports. For Apple® II, II+, IIe, and III in emulation. Requires 3.3 DOS, 48K. Money Street works with one drive, but two are preferred. It's also okay without a printer, but you'll miss a few reports. Master Charge, Visa, COD okay. Add $2.50 on all orders for postage and packing. Back-up $10.

Money back no matter what. Why not give us a try? As with our other programs, if you aren't delighted, we'll give you a full refund (on mail order purchases from us, only).

Computer Tax Service
P.O. Box 7915
Incline Village, NV 89450
(702) 832-1001

$99.95

gle record may appear across several lines). The tables-style report format permits you to subtotal and total categories and to calculate a new "column" based on information from other columns in your table. Examples of the labels-style report format include mailing labels, index cards, recipes, and so on. In the course of designing your reports, you can delete columns, switch columns, sort information, and set the appropriate printer options. Report formats can be saved for recall later.

The documentation is well done. Examples are plentiful and you can become familiar with the features of the program very quickly. Users of WPL (the Word Processing Language from *Apple Writer III*) will be pleased to know that it's possible to transfer *Quick File III* reports to *Apple Writer III*. Thus, any such reports can be included as part of a document you've prepared using that word processor.

Here are a few other guidelines of interest.

Number of files per disk: twenty-six.

Number of categories per file: fifteen.

Length of a record: 1,140 characters (maximum).

Length of an entry: seventy-six characters (maximum).

Number of characters in a file name: twenty.

Number of characters in a field name: twenty.

As you can see, this is a fairly thorough database management package. It does have a limitation, however, that may be significant to some users. This limitation concerns the number of records allowed in a file. Assuming an average record size of seventy-five characters, you can store 250 records per file on a 128K Apple III and 550 records per file on a 256K Apple III. Note that the file size is not a function of the capacity of the disks being used. More specifically, if you have a 128K Apple III, 20,000 divided by the record size equals the number of records per file. Thus, this program is only appropriate for relatively small files. It is excellent at managing files of this size, and if your applications fit into this category, you'll probably be very pleased with *Quick File III*, its documentation, its speed of operation, and its variety of features.

B.U.G. Thanks to all you Business User Group members who stopped for a moment to write in and assist other readers. Here are some of the hints, helpful suggestions, and questions that have come in recently.

Epson Edification. One of the most frequently asked questions has concerned reference materials for the Epson. In particular, readers were interested in knowing how to use an Epson printer with *Apple Writer II*. One way to find out is by reading *Minute Manual for Apple Writer II*, published by MinuteWare (Columbia, MD) and written by Jim Pirisino.

This eighty-page manual would be useful to an *Apple Writer II* beginner, to someone who is constantly being asked to show others how to use *Apple Writer II*, and to anyone who wants to learn more about the hardware that can enhance the package. The book provides an excellent summary of the *Apple Writer II* commands and procedures and explains how they work. There's even a brief section on WPL, but don't expect this guide to help you there.

The book also contains a twenty-two-page chapter on using the features of the Epson printer with *Apple Writer II*. In addition to general instructions, the chapter includes information on condensed mode, double-width mode, condensed/double-width mode, double-strike mode, emphasized mode, italics, underlining, backspacing and overstrike, unidirectional printing, and subscripts and superscripts (in normal, emphasized, and double-strike modes). In addition, there's a section explaining how to create a glossary of all the printing commands. Finally, the Epson chapter describes how to display and print some of the special characters not shown on the keyboard.

If you're still having trouble getting your Epson to do all that it should with *Apple Writer II*, you'll certainly want to consider obtaining this guidebook.

Tracking Epson. Harding Rees of Los Gatos, California, sends in a suggestion for those who are having problems with the tractor feed mechanism on their Epson printers (as was Paul Metzker, Mind Your Business, December 1982). The problem is that the margins begin to vary after a few pages and the tractor teeth begin chewing up the paper.

The Epson manual is remiss, explains Rees, when it talks about installing the paper separator, the metal rack that is supposed to allow the paper to feed smoothly. After connecting the paper separator to the printer, it's necessary to inspect the metal rack carefully to be sure that the rack doesn't rest in contact with the top surface of the printer, thus pinching the paper between the rack and the printer surface. If the paper is pinched, with even slightly more pressure on one side than on the other, the paper will wander toward the side as it feeds through and will cause trouble by the time the fourth or fifth sheet is being printed. The solution is to bend the two shorter wires at each end of the rack down slightly, allowing the paper to slide under the rack without being squeezed between it and the printer surface.

Rees also has a question for other B.U.G. members. He'd like to know if anyone is using the Epson with *Quick File III* or *VisiCalc III*. If so, how can one get the printer to print with a single line feed when using these packages?

VisiCalc Help. Here's a call for help from Robert Krandel of San Jose, California. He wants to get a count of numbers in a column that fall within different ranges. If, for example, he had a column titled SALES that contained the dollar amount of sales for forty different salespersons, he'd like to be able to use the *VisiCalc* function COUNT to determine how many salespersons had sales between $0 and $50, how many had sales between $51 and $100, how many had sales between $101 and $151, and so on.

C'mon all you puzzle-lovers—what's the simplest solution? There's probably a solution involving the IF command, but it looks complicated. Let's all try to come up with something and meet back again soon to compare what we've worked out.

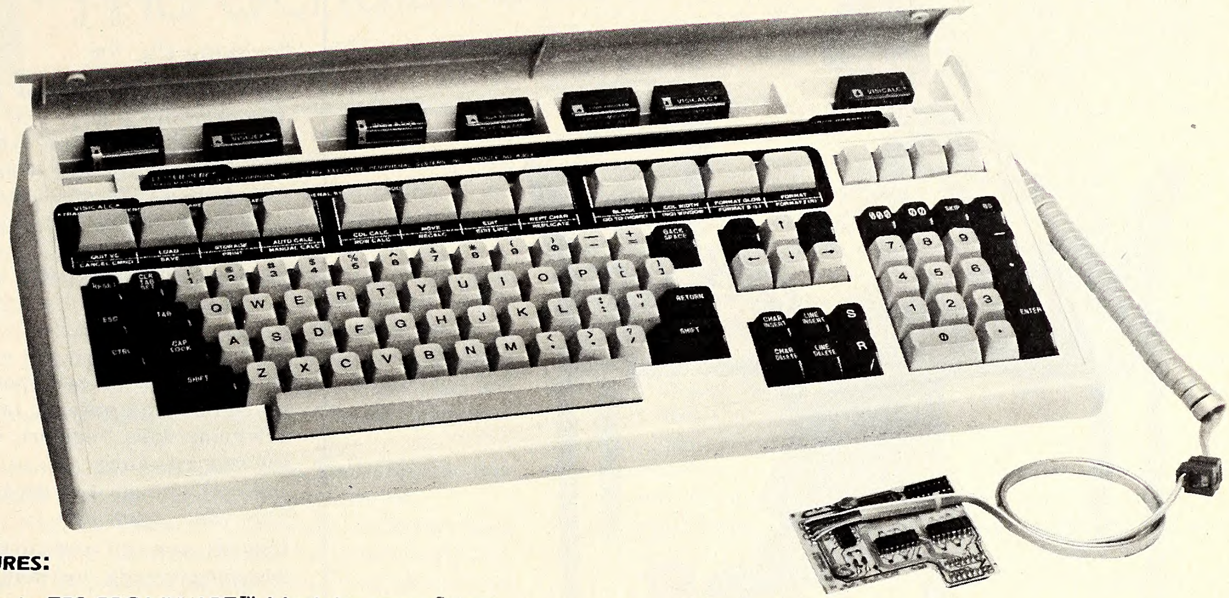
It's Almost June. Well, we've rambled over enough territory for this month. If you get the chance before next month, get in touch. Include complaints, compliments, questions, suggestions, hints for other readers, goals for B.U.G., whatever. Take care until then.

Alpha Software Corporation, 12 New England Executive Park, Burlington, MA 01803; (617) 229-2924. Apple Computer, 20525 Mariani Avenue, Cupertino, CA 95014; (800) 538-9696, (800) 662-9238. MinuteWare, Box 2392, Columbia, MD 21045; (301) 995-1166. VisiCorp, 2895 Zanker Road, San Jose, CA 95134; (408) 946-9000.

A NEW LIFE FOR YOUR APPLE II®

MOVE UP TO THE NEW GENERATION WITH AN

EPS KEYBOARD!



FEATURES:

- Plug-in EPS PROMWARE™ Modules reconfigure keyboard for popular software packages, virtually eliminating complex command sequences.
- 12 Special Function Keys give up to 48 commands for popular software packages when used with EPS PROMWARE™ Modules.
- PROMWARE Modules available for WORDSTAR*, VISICALC*, APPLEWRITER II*, SCREENWRITER II*, and other packages.
- Interface Board plugs into keyboard socket on motherboard, no I/O slot required.
- 6 ft. cord and telephone style jack provide convenient placement and portability.
- Latching cover secures extra PROMWARE™ Modules and Command Templates.
- Full Word Processing layout — convenient and time saving.
- Word Processing Edit Keys (Delete, Insert, Find, Replace).
- CAP-LOCK, Working SHIFT Key.
- Auto-Repeat on all keys.
- Full cursor control with UP and DOWN arrows improves speed in spread sheet and word processing applications.
- 21-key Numeric Pad allows easy data entry.
- Works with APPLE II+ and most older versions, most 80 column cards and printers.

TECHNICAL SPECIFICATIONS

- Full ASCII character set
- Microprocessor controlled
- Pinout matches APPLE II requirements
- Parallel output
- Power 5 Volts, 100 mA
- 19½ x 9″ x 3″
- 5½ lbs.

Ask about optional Softswitch, 20 ft. extension cord, and Burn-Your-Own-Prom Kit.

Economic ** Flexible ** Comfortable

THE COMPLETE, INTELLIGENT SOLUTION

Ideal for both Business and Personal Applications

To order, please call or write to:

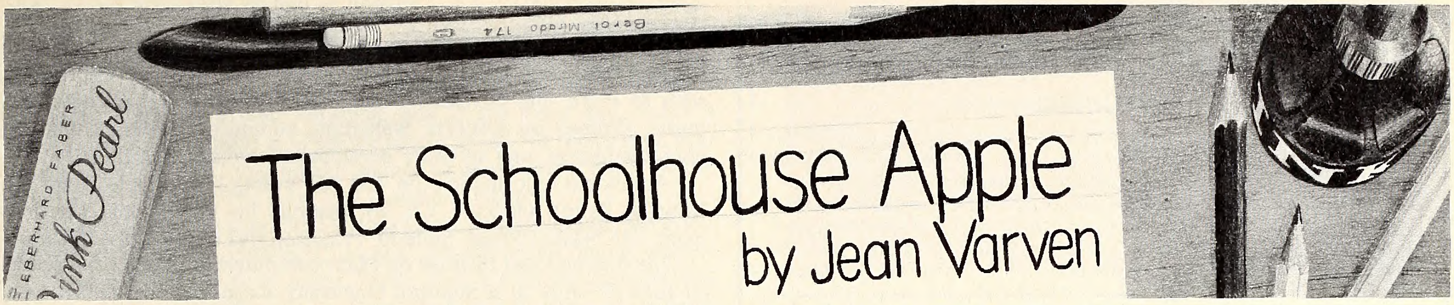
EPS

Executive Peripheral Systems, Inc.

800 San Antonio Road, Palo Alto, CA 94303 (415) 856-2822

See us at BOSTON APPLEFEST Booth A634

EPS and PROMWARE are trademarks of Executive Peripheral Systems, Inc.
*Trademarks of Micro Pro International, Personal Software, Apple Computer Inc., and On-Line Systems.



The Schoolhouse Apple

by Jean Varven

Not long ago, there weren't many programs intended especially for the very young computer user. Lately, however, new companies that specialize in creating software for young children have popped up, and established companies have responded as well. In the course of this month's column, we'll look at a few of the choices available to kids and their parents.

But First, What Else Is Happening? As summer approaches, computer users of all ages are being bombarded with educational opportunities. Conferences, workshops, institutes, and computer camps all over the country are promising an embarrassment of riches, including information, experience with software, and exposure to new ideas and applications, in environments especially conducive to the development of skills.

Let's take a few minutes now to consider a few special summer possibilities, some of which may hold particular appeal for you.

The fifth annual National Education Computing Conference takes place June 6–8 in Baltimore, Maryland. Hosted by Towson State University, the conference is the product of cooperation among nineteen scientific and professional societies involved in educational computing. Stated objectives are to present major work regarding computers in instruction, to promote interaction among educators and others at all levels, to develop and coordinate the activities of various professional organizations, and to produce a proceedings that documents the status of computers in education.

The conference will feature survey and tutorial sessions by invited speakers, presentations of contributed papers, and special panels. Topics will include the current status of the educational use of computers at the national level and projections for the future. Informal "birds of a feather" sessions will be scheduled throughout the three-day program so that attendees with common interests, problems, and goals can connect with each other. In addition, there will be vendor demos and exhibits as well as noncommercial exhibits. Preconference workshops will be held on June 5.

For more information, contact Doris K. Lidtke, NECC-83 General Chairman, Department of Mathematics and Computer Science, Towson State University, Baltimore, MD 21204.

The University of Wisconsin at Madison's Center for Education Research is holding its third annual Microcomputers in Education conference on Saturday, June 18, from 9:00 a.m. to 4:00 p.m. on the campus. Keynote speaker Nancy Grimes will talk about the development and implementation of a computer-literacy program in grades 3 through 9 in Wisconsin's Apple Valley/Rosemount School District, and other educators who use computers will also be giving presentations.

To learn more, contact the Wisconsin Center for Education Research, 1025 West Johnson Street, Madison, WI 53706.

New York City's Sheraton Centre will be the site of a one-day seminar for educators and administrators sponsored by CW Communications on Saturday, June 25. The Microcomputers in Education seminar is part of a three-day event, the Executive Microcomputer Conference and Exposition. Dr. Sylvia Charp, editor-in-chief of *T.H.E. Journal* and director of instructional systems for the Philadelphia Public School System, is Saturday's keynote speaker. In addition to Dr. Charp's presentation on the role of computers in education, there will be sessions on computer funding and acquisition, Logo, teacher training, administrative applications of computers, and educational computing at home.

For more information, or to preregister (which is suggested), contact registration manager Louise Myerow at (800) 225-4698; (617) 879-0700 in Massachusetts.

Necessary *Direction* for Computer Education: Navigational Aids of the '80s is the theme of Computers in Education '83, a conference and summer institute hosted by the University Conference Center at Rutgers University in New Jersey. Both events will focus on the impact that microcomputers and other information technologies are having on education at all levels, and both are intended to benefit teachers and administrators; book, periodical, and software publishers; and others with an interest in the educational applications of computers. According to CE director Dr. Mitchell Batoff, CE '83, cosponsored by more than a dozen nonprofit professional associations, will provide attendees with opportunities "to meet with hundreds of other microcomputer users from all levels and disciplines in education . . . and to engage in free exchange of ideas and experiences."

The conference will be held June 27–29 and will feature nine keynote presentations, plus 150 special interest sessions and short reports from experts and grassroots representatives around the country. Also planned are showings of award-winning films on computers in education, a computer graphics theater, exhibits, and displays.

The CE '83 summer institute, held June 20–July 15, will offer some forty professional development courses for educators and administrators working at the elementary through university levels. Most courses will offer extensive hands-on experience using a microcomputer, so enrollment is limited. The sessions will run an average of three days and may be taken for continuing education or graduate credit. They'll span a broad range of topics, including computer literacy, computer languages

(Basic, Logo, Pilot, Pascal, and machine language), using the computer with special populations (gifted and talented students, and those with special needs), and using computers in the teaching of specific subjects, such as math, the physical sciences, history, the social sciences, and music. Courses on software evaluation, administrative applications of microcomputers, and software development are also scheduled.

Additional information on CE '83 events is available from Dr. Mitchell Batoff, Director, CE '83, Institute for Professional Development, 245 Nassau Street, Suite D, Princeton, NJ 08540; (609) 924-8333.

The International Institute on Microcomputers in Education is slated for June 27—July 29 at Stanford University. Cosponsored by Stanford's School of Education and Interactive Sciences, a nonprofit corporation based in Palo Alto, California, the institute is intended to prepare educators, researchers, and administrators to become computer-resource people for their districts and home schools. During the five-week program, educators will develop key computer-using skills and acquire a background that will help them establish computer centers in their home school systems. Offered concurrently with the institute will be a computer camp for young people ages eleven to seventeen.

Interactive Sciences has been developing and refining effective ways of teaching computer literacy to both children and adults for several years. One particularly significant result of their work is the highly successful Computer Tutor method. At this summer's institute, modeled on an institute and computer camp held by Interactive Sciences and Stanford last year, computer-literate young people will introduce educators to computers, and educators will in turn teach young computer campers how to use computers.

By means of the computer tutor method, participating educators will get lots of hands-on experience with Apples and other microcomputers. They'll learn elementary Basic programming, word processing, database management, and spreadsheet analysis. Through "cross tutoring" (tutoring and learning from other participants) they'll gain and reinforce skills, and learn to teach those same skills to another person. They'll also go on field trips, attend demonstrations of computer equipment, and hear guest speakers and panels of experts discuss computer-related topics.

Interactive Sciences is committed to bringing the power of computer technology to as broad a range of people as possible. In keeping with this philosophy, no prior computer background or mathematical ability is assumed, and a special effort is made to ensure that people who are sometimes excluded from such opportunities (such as girls and women, senior citizens, and the disabled) are included in the program.

To learn more about the International Institute on Computers in Education, contact the School of Education at Stanford University, Stanford, CA 94305; (415) 497-2102. For more information on the computer camp at Stanford or other Interactive Sciences-sponsored computer camps, contact Computer Tutor Camps, 980 Magnolia Avenue, Larkspur, CA 94939; (800) 227-2866; (415) 461-7533 in California.

Troll's Tale. By Sunnyside Soft. In a story as simple as a fairy tale, as serene in its fantasy as a sunlit summer lake, events need not happen in a hard and fast order. As long as all the things happen at one time or another to get the main character to the happy ending, everything's fine.

What an ideal vehicle, then, for an interactive computer story for small children. With their second offering in just this vein, Sunnyside Soft proves it works.

In form, *Troll's Tale* is an introductory adventure, as was its predecessor, *Dragon's Keep.* Instead of typing in commands, the player chooses one of several commands listed on the screen. Press the space bar to move the command you want; press return to tell the Apple you want it.

But, truly, *Troll's Tale* is an interactive story. The player, who is the protagonist, learns right away that the quest is to find and return the lost treasures of King Mark, treasures hidden and guarded by a pesky troll. The player determines the route to travel through the game; it doesn't matter—in the end, players must touch all the rooms and time is not important. Fun is.

All the treasures recovered, the player has a lovely surprise in store. Let's just say it's the perfect fairy-tale ending to this story. Children will be surprised and delighted. And they'll feel they're responsible for it, that they caused this lovely ending to happen.

The graphics aren't the greatest but it doesn't matter; they're sunny and bright and clear. What they lack in polish is more than compensated for by the adventure itself and delightful touches in the text. For example, when you're in a well, something falls in the water and makes a splash; one of the options is to see the splash again. No tricks; it's a neat splash. Another time, you have a chance to check out a bottle in the troll's room. Poison? No way; just something yummy.

Other messages contain little lessons and encourage kids to feel good about themselves and to be curious about the rest of this unique world.

Sunnyside Soft earns its name. *Troll's Tale* is a totally delightful experience for young children, one from which they'll learn many good things about thinking and feeling.

Troll's Tale, by Sunyside Soft, Sierra On-Line (Sierra On-Line Building, Coarsegold, CA 93614; 209-683-6858). $29.95.

Alphabet Beasts & Co. By Classic Family Software. *Alphabet Beasts & Co.* is a well-executed number and letter recognition program for kids ages three and up. It combines sound learning theory, neat, imaginative graphics, discriminating use of color and sound, and touches of humor into a most satisfying and educational whole.

Program operation is very simple, involving single keystrokes and the use of the space bar and return key. It's impossible to do anything "wrong" here; pressing any number or letter key or the space bar has an entertaining result; hitting any other key has no effect.

To see a letter on-screen, the child simply hits that letter key. A picture of a child and an appropriate mythological or fantasy creature then appears, accompanied by a pleasing rhyme for parents or siblings to read aloud that repeatedly uses the sound of the letter chosen. Hitting the same letter key again causes the letter to be printed on-screen by the "magic pencil." This representation of the letter matches the Zaner-Bloser alphabet model—the method most often taught in elementary schools. Hitting some other letter instead produces the picture and rhyme for it.

To see numbers, the child presses the number keys. Doing this activates a nifty animated sequence: A seemingly blank picture frame appears, and then the magic pencil prints out the spelling of the number. When the number key is pressed again, the numeral itself is printed by the pencil. (Hitting a different number or a letter begins the sequence appropriate to that entry.) Next, a little dragon dances onto the screen. When the dragon breathes on the picture frame, a portrait of another dragon gradually comes into view, this one in the shape of the number being displayed. To complete the sequence, more dragons (the correct number) bounce onto the screen, accompanied by a corresponding number of notes of the musical scale.

Alphabet Beasts & Co. contains a fun bonus feature—a Creature Feature, activated by hitting the return key in response to the title page (instead of a number or letter key) or by hitting the return key or the space bar at various other points in the program. This is a mix-and-match sequence in which children can mix up the heads, bodies, and legs of four creatures—a dragon, an alien, a genie, and a boy. The completed creature is then transported to a new setting.

Because the program's creators "know from personal experience that kids and copy protection don't go together," *Alphabet Beasts & Co.* is unprotected; this facilitates the making of a backup. In addition, a sound-off option has been incorporated so that kids can play quietly when necessary.

This program operates effectively on several levels. Its animation, sound effects, and ease of use make it ideal for young children to play with on their own, while its mythological verses and characters encourage family participation and ongoing enjoyment.

Alphabet Beasts & Co., by Classic Family Software, Software Productions (2357 Southway Drive, Box 21341, Columbus, OH 43221; 614-486-3563). $29.95.

Spelling Bee Games. By John Conrad. This program consists of four games intended to reinforce the spelling and reading skills of children ages four through seven. In the process, hand-eye coordination and the ability to memorize are also called into play. The word lists used in the games are the same as those used in *Spelling Bee and Reading Primer*, Edu-Ware's original education system.

In *Squadron*, players are shown words and pictures and must match them up. This is done by guiding a game-paddle-controlled airplane into the correct position and zapping the word that corresponds to the picture by hitting the paddle button. A single player may compete against himself, or two players may compete against each other.

Skyhook may be played by a single player or by as many as four players at a time. This game also involves associating a word with the correct picture. Here, however, a helicopter is part of the action. Players must assemble the word that goes with the displayed picture by choosing the appropriate letters from a truckload of letters provided at the bottom of the screen. Then they must hook each letter individually, maneuvering the helicopter by means of the game paddles and flying the letter to the correct location in another part of the screen and then releasing it. The player gets a point for each letter correctly placed; incorrectly placed letters simply return to the truck to be used later on.

Puzzle is a memorization exercise for one to four players that also involves making word/picture connections. Six squares, each containing a picture, are displayed on-screen for about six seconds. Then they are obscured by a numbered screen or door. Players get points by remembering what and where the objects are when a word corresponding to one of the hidden objects is shown on the screen. When an incorrect guess is made, the object that is actually contained in the box that was guessed is displayed for several seconds. This adds playing value to the game, since it has the effect of giving information to all the players that may cost or benefit them later in the round.

In *Convoy*, a game for from one to four players, each player has a truck with his or her name on it. The object of the game is to type in the letters of the word that corresponds to a picture. Players take turns entering letters, and when a correct letter is entered, the truck of the player who chose it moves part way across the screen. When a player's truck makes it all the way across the screen, a point is earned.

Spelling Bee Games is not a terribly flashy program, but it's well done and offers young learners a nice variety of enjoyable gamelike ways to exercise their spelling and reading skills. It also gives them valuable, unintimidating experience using the computer.

An easy-to-use configuration program contained on disk allows parents and teachers to set up the program for particular players and to test the game paddles to be sure they are functioning properly. From this menu, which is hidden from players' view, adults can also choose which units will be drawn from in a particular game session; there are twenty-two in all, ensuring that players will have plenty of new territory to move on to when they're ready.

Spelling Bee Games, by John Conrad, DragonWare, Edu-Ware Services (Box 22222, Agoura, CA 93101; 213-706-0661). $29.95.

Stickybear Numbers. By Richard Hefter and Janie and Steve Worthington. **Stickybear ABC.** By Richard Hefter and Jack Rice. **Stickybear Bop.** By Richard Hefter, Janie and Steve Worthington, and Spencer Howe. Big, shiny, brightly colored book covers with a whole bunch of goodies inside—what better way to capture the attention of the little ones? How about adding a big, cuddly, funny old bear and giving him a nice wife bear to join in the fun? Now let the bears run rampant through all the good stuff and . . . you've got the kid-pleasing computer packages from Xerox. Xerox? Yup, Xerox. (No, they're not copyable.)

Everything about the Stickybear series and product sticks to the theme. Every detail counts and is executed with quality in mind.

The bunch of goodies consists of bright, uncommercial-looking posters, sheets of sticky (stick-on) Stickybear stickers, a full-size high-quality hard-cover picture book or, in *Bop*, a heavy cardboard punchout and stand-up Stickybear toss game with lots of tossing pieces.

All this is for the preschool-through-kindergarten crowd. Put the child at the keyboard—please with a color monitor or television—boot up a Stickybear disk, and enjoy your favorite novel. If you can quit watching.

Numbers teaches the numbers zero through nine. When the child hits a number, that many somethings come up on the screen in some setting. Let's hit seven; seven plump, spinning spaceships rocket into formation over a moonscape. Hit seven again; seven arrows *thwup* into a target one after the other. Hit it again and seven penguins cavort on an Antarctic iceberg. One more time and seven trains chug into place, smokestacks puffing. There are many more alternatives than there are numbers, and sometimes they come up on different backgrounds as well.

Stickybear is there: Stickybears in the windows of a building; Stickybear downing ice cream sundaes; Stickybear bedazzled by bouncing balls.

In *ABC*, each letter evokes one of two alternative words and pictures illustrating them. It's a lovely touch that not all the words are objects you'd expect. For example, C (for cry) brings a Stickybear with big tears plopping from his eyes; K means kiss, and there's Stickybear and Mrs. Stickybear stealing a kiss.

The graphics are an utter delight. Big, fat, crayon-style pictures done in fine detail in hi-res that's still wondering where it got all those pure, bright colors. Everything has animation—real animation, not Hanna-Barbera style; when the bird's wings flap, his legs and body respond. Some are simpler but fun, chosen for the occasion, not for the ease of execution—for instance, Q, for quilt, evokes a picture of Stickybear in bed, just his nose showing under a mountainous quilt, with a part of the quilt moving up and down as he breathes.

Sound has not been forgotten, and its use is thoughtful. No sound occurs when snow (S) falls in the Arctic; but dancing hats (H) play a little tune, as do happy snowmen. Ducks quack, trains chug, motors putt-putt, rain pings.

Stickybear Bop is a break in the action for action, and it'll appeal to older brothers and sisters, too. (Maybe even moms and pops and teachers. . . .) It's a game using the Stickybears and the same graphics and animation, a shooting arcade that uses pinballs to go after moving objects. The bottom level is unmissable, the top a challenge. Six stages follow completion of the first level; progressing through levels depends on how many objects are hit, not on score. The goals and requirements get harder as the levels progress.

If it continues to publish educational software as carefully made and as delightful as these Stickybear products, Xerox could have on its hands the Golden Books of computerdom.

Stickybear Numbers, *ABC*, and *Bop*, by Optimum Resource/Xerox Education Publications (245 Long Hill Road, Middletown, CT 06457). $39.95 each.

Earl's Word Power. By George Earl. They sound alike, are often spelled alike, but have different meanings. Quick, what are those confusing accidents of language called? That's right, they're homonyms—words such as bark, pool, hear and here, there, their, and they're—and they can cause a person who's just learning them endless frustration. So can other words similar in sound or spelling, such as quit, quiet, and quite; since, sense, and cents.

Subtitled "Horrible Homonyms," *Earl's Word Power* focuses on these problem-causers in an entertaining way. There's no need for documentation explaining how to use the program; simple screen directions (such as "press return") tell users all they need to know. In addition, nice-size hi-res letters make both the instructions and the exercises easier to read.

Homonyms are introduced two or three at a time. Each sound-alike word is defined and used correctly in a sentence. With this information in mind, the learner is ready to begin choosing the correct words to fill in the blanks in the sentences that follow. This is done by pressing a single key rather than spelling out the entire word.

When an incorrect choice is made, learners aren't told that the answer is wrong, only to try again. The incorrect answer then disappears from the screen, narrowing down the choices. When the correct word has been chosen, the differences in meaning and spelling are reiterated and the sentence is presented again.

In *Earl's Word Power*, learning when to use which sound-alike word is seen as preparation, as a "dress rehearsal" of sorts. Three reinforcement exercises—titled *Hamlet*, *Romeo and Juliet*, and *Macbeth* and loosely (very loosely) based on the plots of the Shakespeare plays of the same names—give learners the chance to perform, to use their new knowledge. Their task is to fill in the blanks in a long series of sentences that retell the famous stories. This helps ensure that they know how words are used in context and provides a fun way for kids to consolidate what they've learned.

Earl's Word Power, George Earl (1302 South General McMullen, San Antonio, TX 78237; 512-434-3681). $29.95.

Let's Talk Turtle. The Voice of the Turtle is silent this month. We hope to continue the series in the future. Let us know what you'd like to know about Logo. In the meantime, our thanks to Jim Muller and Donna Bearden of the Young People's Logo Association for their fine contributions.

Schoolhouse Apple reviews were contributed by Margot Comstock Tommervik and Jean Varven.

cdex™
Training Programs Make Using Personal Computers Easy.

Cdex™ Training Programs are available for:

The VisiCalc® Program
The WordStar™ Program
The SuperCalc™ Program
The EasyWriter II™ Program

The IBM® Personal Computer
The Apple® *IIe* Personal Computer

Understanding Personal Computers and Their Applications
Managing Your Business with the 1-2-3™ Program
Managing Your Business with the MULTIPLAN™ Program

Even the best application programs and personal computers still take a considerable amount of time to learn how to use. That is. . .until now!

Cdex Training Programs are computer-assisted training programs that make learning straight-forward and efficient. In an hour, you can master the material on the Cdex disks and begin using the intended product.

It's tough to learn about computers from a book. That's why all Cdex Training Programs are on disk and are:

Highly Interactive...
Creating a dialogue with you and serving as your personal tutor.

Completely Self-paced...
You set your own learning time.

Graphically Oriented...
Remember, a picture is worth a thousand words.

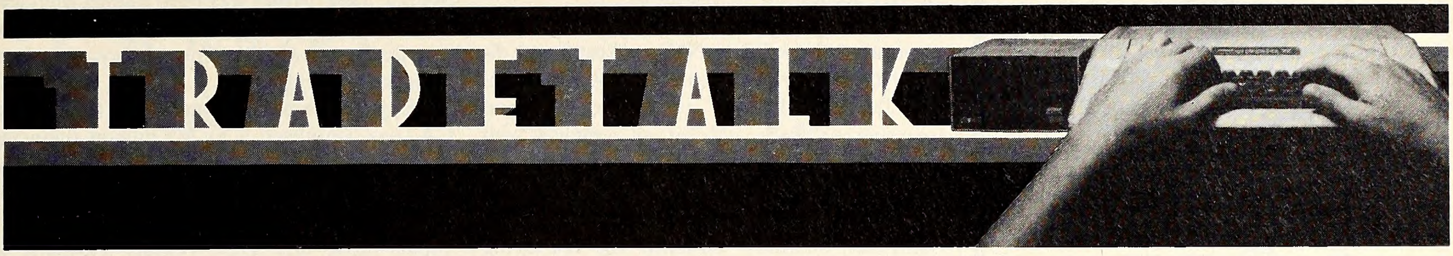
Usable Now and Later...
Clear and concise training and reference programs for the first-time or experienced user.

Cdex Training Programs run on the same Apple® II Plus, Apple® *IIe* or IBM® Personal Computer as your other programs. See how effective a Cdex Training Program can be; ask your dealer for a demonstration.

Cdex™ Training Programs
We make it easy.

cdex™

Cdex Corporation
5050 El Camino Real
Los Altos, CA 94022
415 964 7600



☐ **ComputerLand** (Hayward, CA) will open "satellite" computer stores throughout the Bullock's department store chain in California, Las Vegas, and Phoenix. The satellites will always be located in towns that have ComputerLand service centers for after-sale support. "It's a good match," said ComputerLand vice president of development **Ken Waters**. "For Bullock's it solves the problem of carrying popular but very complicated technology products; to ComputerLand, Bullock's offers a large, high-quality clientele. I think we'll do well together."

☐ **Marcia Klein** has been named president of **The Learning Company** (Portola Valley, CA), producer of elementary-level educational software. She was previously vice president of the consumer clients group at Regis McKenna, a high-technology marketing and public relations firm, where she orchestrated the introduction of the Lisa and the Apple IIe for Apple Computer.

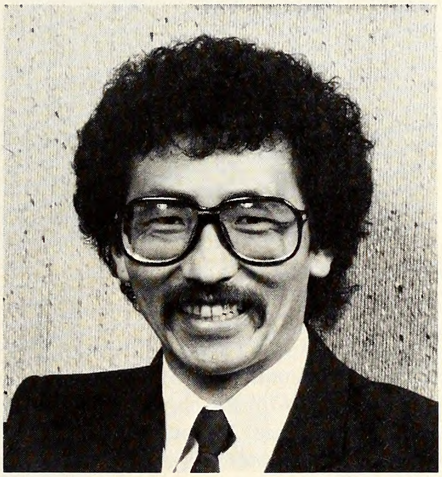
☐ **Saber Software,** maker of *Demon's Forge*, has been purchased by and incorporated into **Boone Corporation**, located at 2900 Bristol, Building A, Suite 102, Costa Mesa, CA 92626.

☐ **Silicon Valley Systems** (Belmont, CA) has signed a contract with **T.C. Data Limited** (Dorval, Quebec), a major Canadian distributor. "We have the utmost confidence that T.C. Data Limited will provide outstanding Canadian service to our Canadian dealers," said Silicon Valley president **Nathan Schulhof**. "They'll be like an extension of our firm." Adding that "we feel strongly that training tomorrow's leaders adds to our corporate strength," Schulhof also announced that his company is donating $30,000 worth of software to the Canadian school system.

☐ **Northwest Instrument Systems** (Beaverton, OR), designer of personal computer peripherals for engineering, service, education, physical science, and manufacturing test applications, has signed a distribution agreement with **Apple Computer** (Cupertino, CA), whereby Apple will distribute Northwest's Apple-compatible instruments in West Germany, France, and the United Kingdom. "The evolution of the personal computer into the technical community has happened faster in Europe than in the United States," says Apple European marketing manager **Phil Roybal**. "This was the impetus for our marketing agreement with Northwest Instruments. According to our research, we currently have the largest installed base of personal computers in technical markets." Dealers are now being selected and trained for the European technical market. The first Northwest product to be distributed by Apple will be the Model 85 aScope Digital Memory Oscilloscope.

☐ **Steve Werschba**, the new controller for **DataMost** (Northridge, CA), will be in charge of the company's accounting department and assist in future planning.

☐ **Infocom** (Cambridge, MA), producer of interactive text adventures, has beefed up its research and development capabilities with the installation of a thirty-six-bit DECSystem 20/60 computer, the largest mainframe manufactured by Digital Equipment Corporation. Of the half-million-dollar machine, president **Joel Berez** says, "When perfecting our skills at the Massachusetts Institute of Technology, we learned that the most cost-effective element in product development is optimum programmer productivity, not hardware; even though this machine is very expensive, in the long run we feel it will be a much more economical method of developing product software than using target machines."



Scott Oki, Microsoft's vice president of international operations.

☐ **Microsoft** (Bellevue, WA), developer of operating systems, languages, hardware, and software tools, has named **Scott Oki** vice president of international operations. Said operations include new subsidiaries in the United Kingdom and Europe, as well as ASCII Microsoft, the largest publisher of computer magazines in Japan. Oki directed the company's penetration of the European market. "We will continue to put heavyweight performers like Scott into key positions as we continue to bolster our management teams worldwide," says president **Jim Towne**.

☐ **Philip C. Davy**, founder of **Infoscribe** (Santa Ana, CA), maker of five models of matrix printers, has been elevated from president to chairman of the board and chief executive officer. Taking over the position of president and chief operating officer is **David M. Connell**, who will also be on the board of directors. The company expects to ship seventy-five hundred printers in 1983 and promoted Davy to prepare for the growth in sales, freeing him to concentrate on overall business and marketing strategies. "Connell's charter," says Davy, "will be to build the solid foundation in business practices necessary to maintain a balance between growth and profitability."

☐ **Dale Caravona** has been appointed national sales manager at **State of the Art** (Costa Mesa, CA), the accounting software house. He joins the company from Siemens Corporation, an electronics firm where he has served as regional marketing and sales manager for the last two years. **Jim Hennings** has been appointed director, professional markets. He will be in charge of establishing educational programs that meet the needs of sales representatives, retailers, and end users, and developing a network of user organizations to support State of the Art accounting applications.

☐ **Joanne Koltnow** has joined **Scholastic Incorporated** (New York, NY) as West Coast director of software development. She will be responsible for finding, commissioning, and acquiring software and for all contacts with West Coast developers and manufacturers in the field of computer education. As product manager for educational software at Apple Computer, Koltnow designed the marketing strategy for Apple Logo and the Sesame Street software. She will bring the viewpoint of a professional educator to the development and uses of software, identifying national market needs in the schools and in home education.

☐ **Jim Hunter** has been appointed product line manager, software, at **Howard W. Sams** (Indianapolis, IN). He will oversee acquisition, development, documentation, and packaging of software, and act as liaison in the areas of marketing and advertising. He was previously Hayden Software's entertainment product line manager.

Barbara Newland, director of product development at Source Telecomputing.

☐ A letter of intent has been signed between **Source Telecomputing** (McLean, VA) and **Control Data** (Minneapolis, MN), making the mini and mainframe computer firm an investor in the Source. *Reader's Digest* will maintain its controlling interest.

A new department has been created to guide the development of the information and communication services the company provides to personal computer users. Headed by **Barbara Newland**, former director of marketing, the new product development department will initially be staffed by former managers from five other departments: **Jeana Hood**, manager of product acquisition, will negotiate the acquisition of database services and handle ongoing relations with information providers for the Source; **Pat Lobenstein** has joined **Julie Peck** and **Beth Silverman**, both from the marketing department, as a new product manager; **Robin Cobbey**, manager of product research, will oversee subscriber surveys and evaluate potential information providers; **Taylor Walsh**, manager of product standards, will create standards for user friendliness and develop a corporate "look" for the Source. **Jane Brown** has been named manager of business analysis and will work in the corporate development department analyzing future business opportunities.

☐ **Epyx/Automated Simulations** (Sunnyvale, CA) has completed a $1.725 million financing package; $1 million in venture capital is going to the computer games company from the Early Stages Partnership, a San Francisco venture capital firm, and U.S. Venture Partners, of Menlo Park, California; an additional $725,000 comes from the Bank of the West in San Jose, California. "This capitalization financially positions the company so that it can take advantage of opportunities in this rapidly expanding market," says vice president **Robert DeDominic.**

Epyx is about to launch a major media effort on behalf of its thirty-odd game programs, and to that end has retained the services of the **Chiat/Day** ad agency, citing the good things the agency has done for Apple Computer. The account will be handled out of the agency's San Francisco office, with **Maurice Goldman** as account manager and **Bill Foote** as account planner.

☐ **Stephen R. Wilson** has been named director of western operations for **Amdek** (Elk Grove Village, IL). His twenty-four years' experience in the computer/electronics business includes various positions with ITT Components, TRW, and Litton Industries. He will be headquartered at the company's new twelve-thousand-square-foot warehouse facility in Costa Mesa, California, opened to provide sales support, service, warehousing, and expediting for the company's entire product line.

☐ **Lifeboat Associates** (New York, NY) has announced the appointment of **Peter Sulick** as vice president and chief financial officer. Formerly company comptroller, he is responsible for the company's financial operations and planning as well as the implementation of corporate policy. Lifeboat software is currently available on almost two hundred eight-bit and sixteen-bit microcomputers.

☐ **Shugart** (Sunnyvale, CA) has opened a purchasing office in Singapore to develop resources in the Far East for its floppy and hard disk drives and controllers. The location was chosen for its proximity to sources of low-cost parts such as die castings, stampings, plastic injection-molded parts, and assorted machine parts. **Wayne Klusmeier** has relocated from the company's Sunnyvale headquarters to act as purchasing manager in Singapore.

☐ The lawsuit brought by **WIDL Video** (Chicago, IL) against **Advanced Software Technology** (Kansas City, MO) claiming copyright infringement of WIDL's *Blue Book for the Apple Computer* by *Vanloves Apple Software Directory* has been settled out of court. "Advanced Software Technology will be licensing the *Blue Book* listings that they are printing in *Vanloves*," says **Barry Fleig** of WIDL. "They will pay a certain dollar amount to WIDL Video for every copy of *Vanloves Directory* sold, and WIDL will be given credit on the spine and copyright page."

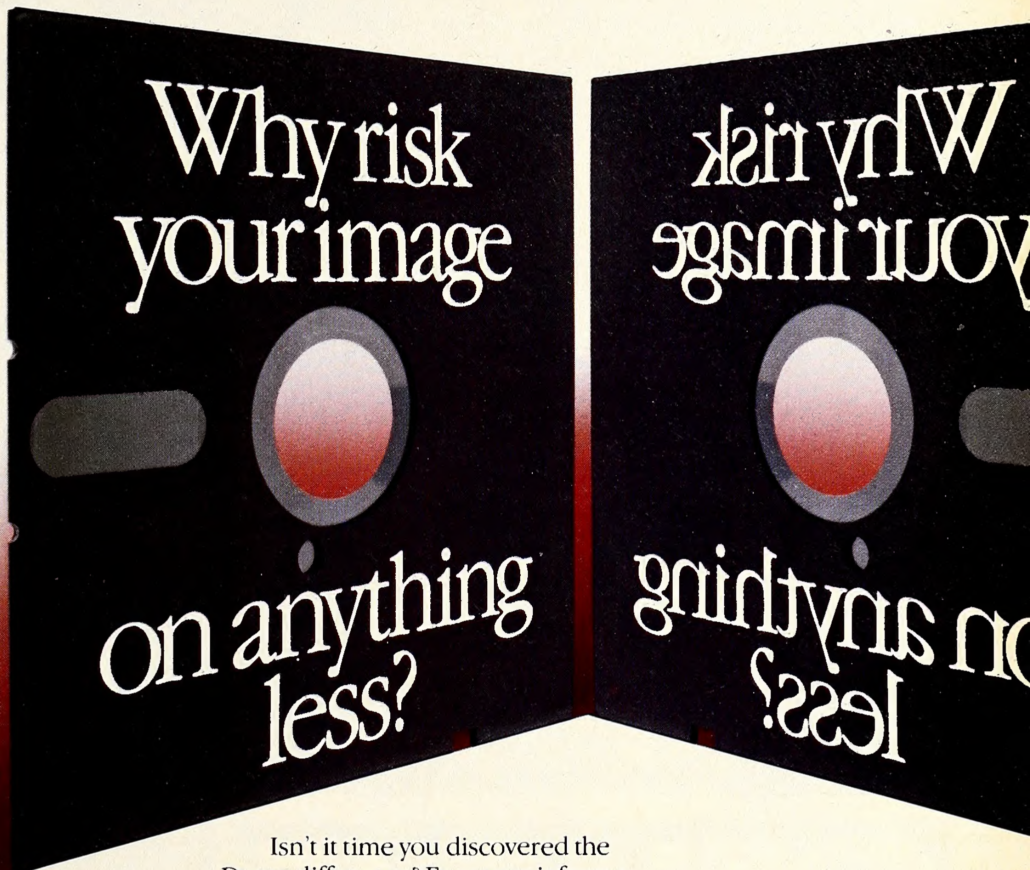
DISCOVER THE DYSAN DIFFERENCE

Dysan Software Duplication:

It's your name on the package label. And your company's reputation on the line. Whether your program retails for $40.00 or $400.00, or is for company internal distribution, the cost of duplicating it on diskettes is just a fraction of the value of your product. Doesn't it make sense to protect the time, money and talent invested in your software with the finest and most complete software duplication services available?

Quality Software Deserves the Quality Media.

Dysan's software duplication services are unsurpassed for fidelity of reproduction. Not only is your program copied unerringly onto the finest media made—the Dysan diskette—but it's also copied on proprietary equipment manufactured by Dysan, exclusively for Dysan. Plus Dysan offers you the widest variety of support services available—from software protection to serialization and packaging.



Isn't it time you discovered the Dysan difference? For more information on Dysan software duplication, fill out and return this coupon today, or call (800) 551-9000.

Dysan CORPORATION

Dysan Software Duplication Division ST-5
5201 Patrick Henry Drive
Santa Clara, CA 95050
(800) 551-9000
(408) 988-3472

Please send me more details on Dysan's Software Duplication Services.

Name: ______________________

Company: ______________________

Address: ______________________

City: __________ State: __________ Zip: __________

Phone: ( ) ______________________

STATE OF THE ART

PROFESSIONAL MUSIC SYSTEMS ON THE APPLE

by Tommy Gear

Art and technology meet in the person of the artist, who feels constantly compelled to find a better or more challenging means of expression. For the artist who has chosen a medium as evanescent as the very air we breathe—the musician—this is especially true. The movement of a column of breath in a metal pipe, the flex of a string—these are simple techniques for producing vibrations in the air, which we perceive as sounds. Such techniques, once developed and refined, make possible the creation of sounds that act as bearers of complex meanings and powerful emotions. The many instruments of the symphony orchestra reflect the variety of methods that have evolved to achieve this. The advent of electronics now offers twentieth-century musicians a new medium with which to produce sound, as well as a new means by which to understand it.

Using electricity, sounds can be produced through circuits called oscillators. The electrical signals oscillators produce become audible only after they are processed by an amplifier and sent to a speaker, which produces the vibrations in the air we call sounds. Before that point is reached, though, signals can be manipulated using electrical voltages and circuits, called control voltages and filters; this process is called subtractive sound synthesis. A speaker's output and a sound's timbre are ultimately determined by manipulations such as these.

The Sound of Electronic Music. A sound can be described by graphing the way it makes the air move. The line on such a graph would be in

the shape of a wave. One simple waveform that's produced by an electronic oscillator is called a sine wave. Though a sine wave can describe any pitch, it is characterized by an almost sterile purity of tone. More complex waveforms can be created by combining sine waves and other simple waves together. This produces more interesting sounds.

Producing sounds by adding waveforms together is called additive sound synthesis. A sound generated in this way would consist of a fundamental wave that determines its audible pitch, and a variable number of barely audible overtone pitches, called harmonics. Each unique sound contains a different number of harmonics in varying strengths. This strength is measured in amplitude, which reflects the change in air pressure caused by the presence of different harmonics.

Another important way to describe a sound indicates how it begins, sustains itself, and ends. The amplitude changes that this process reflects can also be graphed—the result is called a sound's envelope. Along with this, a given sound's unique waveform and harmonic spectrum determine its timbre, or characteristic sound. These are the qualities that distinguish the sound of a horn from that of a piano.

Sound synthesis is the creation of sounds by electronically controlling all the parameters that define them. Using voltage control to vary parameter settings over a range is called analog synthesis, because the voltages represent (or are analogs to) the parameters being controlled. Digital techniques and circuitry can also be used to produce and control sounds, and this is called digital synthesis. To be able to hear sound produced digitally, it's necessary to convert the digital information into analog information, which can then be amplified and played on a speaker. This translation is done with a digital-to-analog converter.

Applying a computer to the creation of sounds and music digitally or as a controller of analog synthesizers coheres the ongoing struggle by musical artists to come to terms with technology and use it as a means of expression. This brief overview gives an idea of the type of knowledge required of musicians who choose to pursue the digital muse.

We'll look at the professional systems currently available to produce music with your Apple. Each works in a distinctive way. Three of the systems use the Apple directly to generate sounds. The others harness various analog synthesizers to the Apple, delegating to it the responsibility of master controller.

Mountain Music Madness. The core of the Mountain Music System consists of two connected hardware boards that must be inserted into consecutive slots of the Apple. Attached to them is an audio output jack through which the system is connected to a mixer or stereo via a standard dual RCA audio cable. A light pen (attached to an impractically short cord) is also provided. The system was designed to accept commands from the Apple keyboard, a set of paddles, and the light pen.

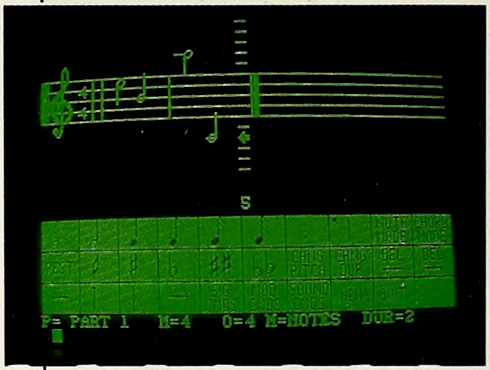
Certain command options are available through only one of these means.

The sounds themselves are generated by sixteen independent user-programmable digital oscillators. A waveform table of 256 bytes is used to store the data for the periodic waveforms generated by an oscillator. When the waveform tables are residing in the Apple's memory, each oscillator needs to read a new entry from its waveform table every thirty-two microseconds.

The digital data is made audible using eight-bit digital-to-analog converters. The oscillators are grouped into two clusters of eight. The even-numbered oscillators are assigned to the left stereo channel, the odd-numbered ones to the right channel. The outputs of each cluster are combined to form a complex waveform. The Passport Soundchaser and the alphaSyntauri keyboards are both designed around the Mountain hardware. Each system provides you with a unique software package, however.

Let's take a look at the menu of the first of the two disks provided, the System 1 software. Its three programs are *Music Player, Music Editor*, and *Music Merger.*

Music Player allows you to play songs that have been stored on the flip side of the System 1 disk as song files. When a song file is loaded, the



With the Mountain Music System software, the *Music Editor* screen displays the notes on a musical staff as you input them with the light pen or paddles.

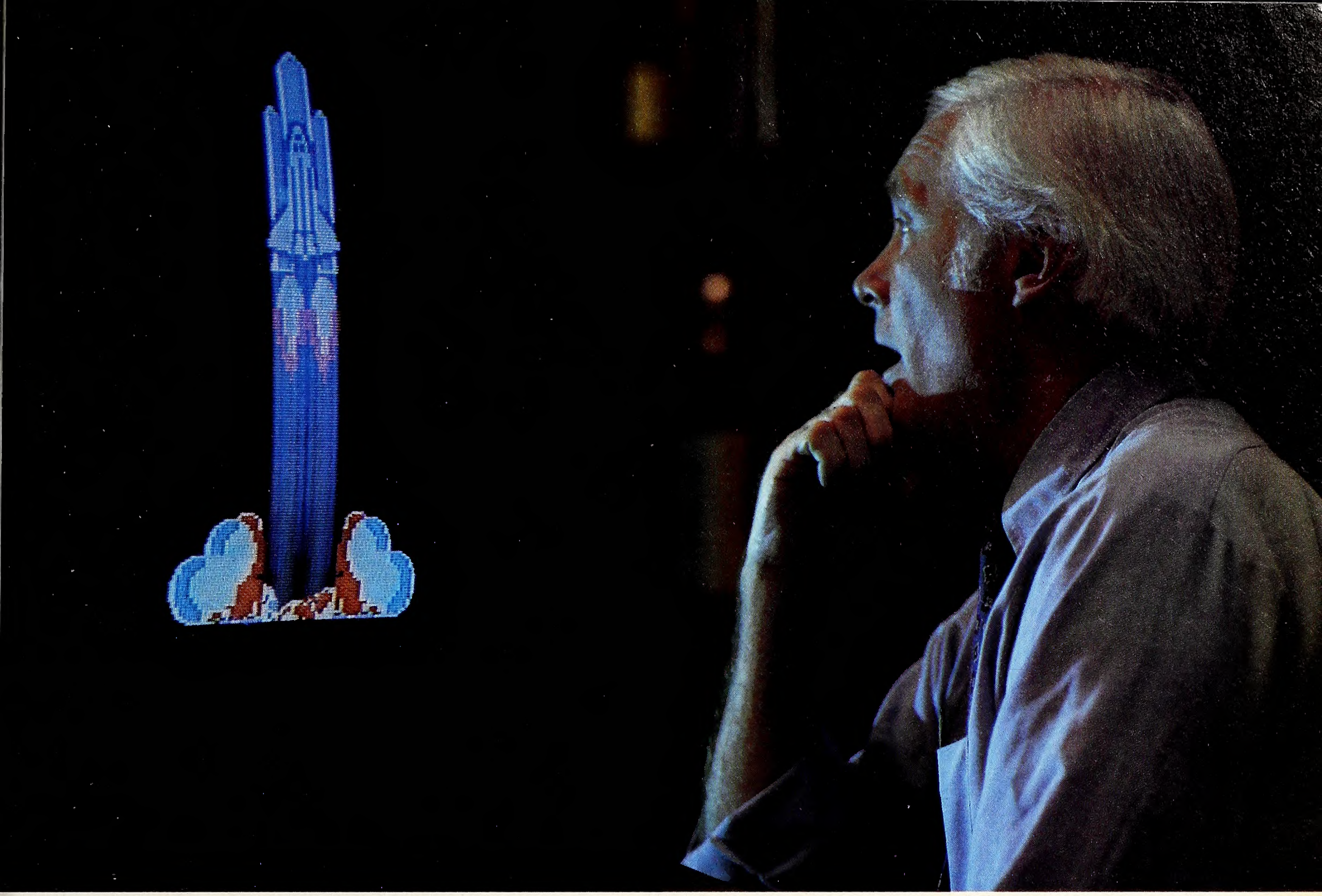
screen displays the song title and a description of some parameters that comprise the song, such as the number of instrumental parts, their names, and how the individual parts have been assigned between the left and right stereo speakers. You can alter the speaker assignment, set the entire output to mono, or change the arrangement of instrumental sounds that are playing the various parts in the composition. The song plays with all the parameters you've chosen. It isn't possible to play a piece of music in real time with the Mountain software. Only music loaded from a song file can be played.

The creation of song files and the modification of existing files is done with the *Music Editor.* The *Music Editor* also enables you to display and print musical scores and to load and save compositions.

In the top half of the *Music Editor*'s hi-res screen is a musical staff on which notes are displayed as they are input. The bottom half of the screen consists of the *Music Editor*'s main command menu and a status line. Using the light pen or one of the paddles, you can make choices about the duration of individual chords, notes, and rests, measure placement, and accidentals. You can also change pitches, delete segments, or scroll to other parts of the piece.

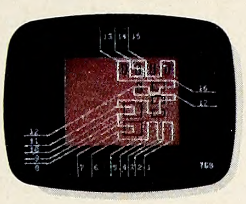
The status line at the bottom of the screen indicates the current condition of some important aspects of a composition. It tells you the part or voice of the composition that you're currently working on (the screen can only display the notation of one voice at a time), the measure number you're on, and the current octave you're working in (there are eight functional octaves; the screen can display only four at a time).

Before entering any notes, you have to make some decisions about what you want to compose. This means specifying the clef, time, and key of the composition from the signature commands menu, accessible from the *Editor*'s main command menu. Once this is done, you can place individual notes or chords on the hi-res staff using a paddle in conjunction



THE GRAPHIC SOLUTION

Solve your toughest communication problems with the Graphic Solution™, a sophisticated, new graphics package from Accent Software.



With precise, multi-speed ANIMATION create captivating sales presentations and product demonstrations that will both intrigue and inform your clients and customers. Watch their reactions; you'll see your messages getting through.

Develop educational materials and training aids that MIX TEXT AND GRAPHICS on the screen, breathing new life into abstract, hard-to-grasp concepts. Mix programs too. Images can be displayed on backgrounds loaded from any of your other programs. Construct custom TYPEFACES AND TYPESIZES to balance the visual elements.

Tired of run-of-the-mill business graphics? Change standard charts and graphs into colorful THREE DIMENSIONAL PERSPECTIVES. Add text and animate the data to show the relative rates of change for your most important information. Like cash flow projections. Or revenue estimates.

Plot flowcharts, time and motion studies, industrial process flows with COLOR-CODED ELEMENTS highlighting critical paths. Animate the sequences to show how flows actually progress.

Work with live action? Prepare film and videotape storyboards using the unique FRAME-BY-FRAME graphic sequencer that lets you create and animate a video story before shooting.

Whatever your graphic communication demands—in the business world, the arts, industry, education—The Graphic Solution™ at $149.95 has the answer. Take a hard look at The Graphic Solution. You'll like what you see.

The Graphic Solution requires a 48K Apple II with ROM Applesoft and DOS 3.3.

See your local dealer or send $10.00 for a demonstration diskette to:

ACCENT SOFTWARE, INC.

3750 Wright Place, Palo Alto, Calif. 94306 Telephone 415-856-6505 with the main commands menu. No music keyboard peripheral is included with the Mountain software package.

Two additional sections give you greater control of refinements. With the sound control menu, you control variations in overall dynamics (graded from pianissimo to fortissimo), tempo (from lentissimo to presto), and spatial location (determining from which stereo speaker the sound output of a given part of the composition will emanate). With the note modifier menu, you can choose note and chord dynamic accents and create ties between notes of different durations.

When you've finished entering the musical information that comprises the current composition, you can hear it by saving it to the song files disk and then reloading it. Some disk swapping is required here if you have one drive. Using the *Music Merger* program, you can combine many composition files together into one composition.

The second disk provided by Mountain is System 2, the *Instrument Definer.* This disk contains the software for creating sounds by means of additive synthesis techniques. On its flip side is a collection of twenty or so predefined instrumental sounds, called presets, that have been synthesized for you. They serve as a good starting place to apply the sound analysis capabilities of the System 2 disk.

In creating or analyzing a sound, the Mountain system allows for the manipulation of a number of discrete parameters. If any of these parameters is altered, the sound they describe will change.

Any given sound, instrumental or otherwise, can be graphically represented as a waveform. The System 2 disk makes it possible to plot waveforms and to combine up to sixteen different ones in varying degrees of intensity in molding a desired sound.

Recall that each waveform consists of a fundamental frequency and a series of harmonics. You are provided with a bar graph on which to plot waveforms to the twenty-third harmonic, each harmonic in amplitude increments of one hundred units. The overall amplitude and frequency modulation of a sound, as well as its envelope, are essential parameters determining its distinctive quality, or timbre. All these aspects of sound synthesis can be controlled and graphically scrutinized using the System 2 disk.

The system is built around sixteen oscillators for generating sounds, but some sounds may use more than one oscillator to achieve their timbre. This implies a lessened note capacity for the piece in which such sounds are used. Therefore, it is incorrect to say that this is a sixteen-voice system. It is left to the user to keep track of the allocation of the oscillators when composing; otherwise problems become apparent only later when you attempt to play the piece.

The Mountain Music System relies exclusively upon standard music notation for generating compositions. When you're trying to harness the computer's power in developing musical ideas, the limitations imposed by standard notation can be stultifying. In addition to this, certain quirks exist even in the way this standard approach is handled. Adjacent flagged notes (such as eighths and sixteenths) won't be connected the way they should be to conform to standard practice, which makes for difficult reading on the staff. Tempo markings are restricted to being the same across the instrumental voices in a given piece; therefore, polyrhythms (overlapping rhythms of different tempos) and other inventive arrangements are ruled out. Also, all the accidentals in a composition will be indicated as either all sharps or all flats, depending on the key signature.

The use of the classic terminology to indicate dynamics is questionable when the computer could control these parameters more specifically if given a range of numeric inputs. It is simply not possible to achieve any subtlety of musical phrasing given the approach taken in the *Music Editor* software. A most annoying aspect of using the system is the long waiting time that often occurs when the program needs to get something from the disk.

The manual, which presents an excellent step-by-step explanation of how to use the system, is appropriate for use by someone new to the concepts of sound synthesis. Reference sections are offered as an aid to those who want to explore these topics more deeply.

In the process of developing a sound, the System 2 software gives you constant audio feedback, so you can hear any changes as you're making them. If you happen to make a mistake as you're learning how to use the system, helpful prompts indicate the error of your ways and try to steer you aright.

Passport to Music. The Passport Soundchaser Computer Music System from Passport Designs uses the Mountain Computer hardware with software specially developed for the Soundchaser. In addition, Passport provides a four-octave, organ-type keyboard connected to an interface card that goes into the Apple's slot 7. Music education software that works with the system is available, as are program packages for transcribing notes played on the keyboard, for alternate keyboard tunings, and for combining pieces in record album form onto a single disk.

The Passport software is fast, versatile, and easy to use. The few global similarities it shares with the Mountain software result from the fact that both systems are constructed around the same hardware base. In addition to the eight-track digital performance software already available for the Soundchaser, Passport has recently released *Turbo-Traks*.

With the Mountain hardware, *Turbo-Traks* provides a sixteen-oscillator, digitally programmable synthesizer and sequencer with real-time access to sixteen preset programmable sounds, individually playable on the Soundchaser polyphonic keyboard. Sounds may be created through additive synthesis, then stored to disk as presets. In addition, 112 presets are provided on disk as part of the system. The *Turbo-Traks* sequencer operates much like a sixteen-track tape recorder. It will store a passage as you play it on the keyboard; then you can listen to it and play along, with the sequencer continually storing each consecutive overdub.

The program to accomplish all this resides on a single unprotected disk. Consequently you can work comfortably with only one disk drive. Both the Apple and paddles are used in interacting with the system. You can use a joystick, but avoid the self-centering kind—otherwise, problems of pitch control will arise.

Upon booting *Turbo-Traks*, you're presented with the cryptic-looking preset screen. This screen contains collective and individual information about the sixteen presets that are automatically loaded when you boot the system. You can now choose any one of these presets and play it

on the Soundchaser keyboard. You can even split the keyboard at any point and assign a different preset to each segment. Enabling the paddle potentiometer for pitch bend is possible from the preset screen, as is setting the control for the overall output volume.

To understand better how *Turbo-Traks* is structured, a little more detail on the way the Mountain hardware boards operate is in order. As you'll recall, this hardware provides a total of sixteen oscillators to produce sounds, divided between the left and right stereo output channels into two groups of eight. These are the physical oscillators, and they limit the number of sounds possible at a given moment to a maximum of sixteen. Although this seems like a reasonable number, recall that if you choose to combine them in order to construct more complex timbres, the number of individual notes you can play is consequently restricted. These sets of instructions in the software that we use to determine how the physical oscillators will sound (their timbres) can be considered logical oscillators. The more logical oscillators we call into play, the fewer the notes we will be able to play on the keyboard at one time.

It would seem reasonable to assume, then, that when using only one logical oscillator it would be possible to play a total of sixteen notes simultaneously on the keyboard. Unfortunately this is not so on the Soundchaser keyboard, and this limitation is due to the structure of the Mountain hardware boards. The logical oscillators of *Turbo-Traks* must follow the same division between channels into groups of eight as the hardware's physical oscillators. If we're utilizing only one logical oscillator, it can only be assigned to either the left or the right channel. The result is that the maximum keyboard polyphony possible with *Turbo-Traks* is sixteen voices, but only if you split the keyboard and assign a single left oscillator to one segment and a single right oscillator to the other. When not using this approach, the maximum polyphony possible is eight voices.

The *Turbo-Traks* preset screen contains a matrix that graphically represents how the sixteen oscillators are distributed among the sixteen presets currently in memory. A simple command changes the sound of any preset by redistributing the oscillators. The change is immediately reflected on the matrix and in the sound produced by the keyboard.

The parameters that determine the sound produced by an oscillator may be varied for each of the sixteen oscillators. Modifying a parameter merely means changing the numeric value associated with it, and this change is audible in real time. The sound parameters controlled in this section of the program are pitch by octave, frequency modulation, and envelope shape. Alternate tunings for the Soundchaser keyboard can be created using a separate utility.

Passport to Pleasure

BY ANDREW CHRISTIE

David Kusek thinks the sound of music is the new frontier in home computers. "Today's music is being generated on inexpensive equipment in people's homes," he says. "There's a whole new market opening up in computer music, and the personal computer is paving the way."

Kusek and John Borowicz have been running Passport Designs out of Half Moon Bay, California, since 1980. They came to the computer music business via employment with Electronic Music Labs, one of the big three of the original synthesizer manufacturers, along with Arp and Moog. Following their stint with EML, they founded Star Instruments in Connecticut, currently one of the largest manufacturers of percussion synthesizers, where they developed the first computer-controlled instrument synthesizers.

"We left Star Instruments to pursue our interests in the personal computer revolution and its effect on the music industry," says Kusek.

Passport's central contribution to that industry is the Soundchaser computer music system. "The home organ of the future" was originally introduced for the Apple in an analog version with *Notewriter*, a program that allowed real-time music composition, and *Musictutor*, a computer-aided instruction package to develop listening skills and general music theory.

The analog system did not do that well in the marketplace. Three years ago there wasn't that much of a marketplace, and the pricing of the system was not competitive with the standalone synthesizers.

Passport introduced the Soundchaser Digital keyboard and performance software in November 1981. The four-octave keyboard comes with software for an eight-voice polyphonic synthesizer, allowing the creation of waveform programs, defining instruments, plus four-track sound-on-sound recording, looping, and real-time control of sound. Orchestral arrangements can be stored and played back with a multitrack sequencer that records and layers the individual parts.

The company's sales increased 258 percent in the six-month period following the introduction of the Soundchaser Digital, a growth aided by its marketing arrangement with Mountain Computer that allowed the packaging of Mountain's Music System synthesizer board with each Soundchaser system.

"With Mountain Computer, we initiated what I believe to be the first co-op marketing campaign between two electronics manufacturers," says Kusek, "and the success has just begun to show. We have developed a product that is higher quality and lower priced than the competition and plan to expand our markets accordingly."

Passport continues to aim at three markets—Apple computer users, performing musicians, and professional educators. It is focusing more on the home consumer market these days, developing more music instruction programs, and looking into other computers. "Our forte is software," says vice president of marketing Chris Albano, "and that's what we're concentrating on." This is in line with David Kusek's policy of continually updating Soundchaser "to offer consumers the most powerful, low-cost computer music system they can buy. The personal computer revolution is letting us put high-level digital synthesis in the hands of people who could neither afford nor understand it a few years ago."

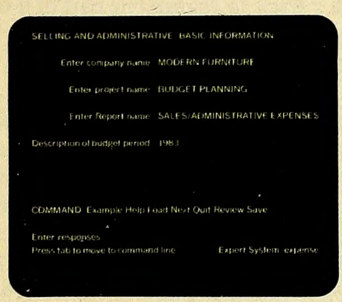
Turbo-Traks was the third software package to join the Soundchaser library, incorporating user ideas and suggestions to create, in effect, a live performance synthesizer and sixteen-track recording studio. It simulates an analog tape deck in software, variable number of oscillators per voice, sync to tape or drum machine, and extended recording time. Soundchaser's latest offering, *Kaleido-Sound*, is a real-time graphics program that synchronizes to any audio output to produce a four-color kaleidoscope on-screen, after the fashion of the color organs that were hot items in the psychedelic sixties. "It's actually quite mesmerizing," Kusek says of his company's first entertainment software product, "not like a game program that you eventually master and get bored with."

Even with the emphasis on home-synthesizing for the average user, Passport has helped out other people, too. The first customer to patronize the company on Half Moon Bay was Alan Greenwood, then keyboard player with Foreigner. Brent Mydland of the Grateful Dead also gets a lot of use out of his Soundchaser, as do Roger Powell of Todd Rundgren's Utopia, Tom Chase, who scores the television series *Fame* and *One Day at a Time*, and Andy Musson, music director for Bette Midler and the Manhattan Transfer.

The intrepid band of engineers and musicians who started the company with Kusek and Borowicz are looking to make a big noise in the business, and the growing reputation of their computer music company is music to their ears. ■

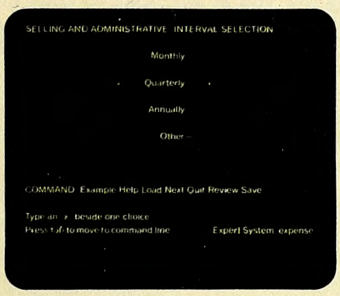
ZERO TO MULTIPLAN™ IN 5.2 MINUTES.

FINANCE OR ACCOUNTING WORKSHEETS FAST.



Gentlemen, start your computers.

Time: 0

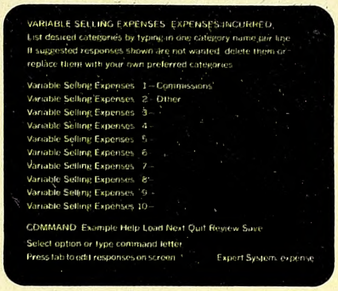


Select budget intervals.

Time: 0.5

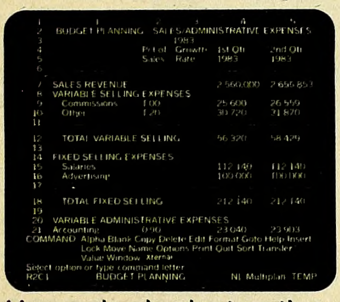
Enter sales revenue.

Time: 1.0



Enter selling expenses.

Time: 1.5



Your sales budget on the Multiplan electronic worksheet - in record time.

Time: 5.2

First, Microsoft created the Multiplan interactive electronic worksheet, to help you analyze your business problems and explore possible solutions. Without asking you to become a computer expert.

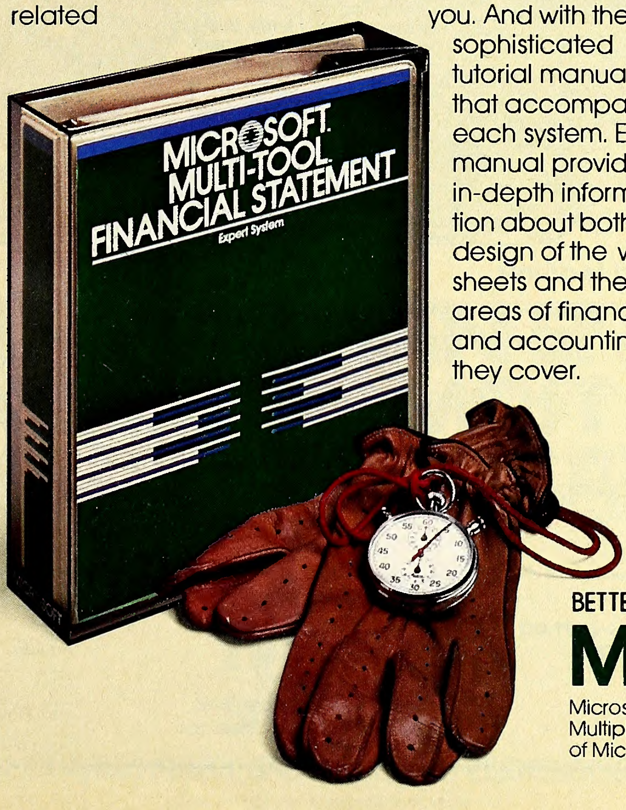
Now we've added the Multi-Tool™ budget and financial expert systems. They can help design and build financial or accounting worksheets tailored to your specific needs. In minutes.

You won't have to worry about developing formulas or formatting screens to build your Multiplan worksheets. Because the expert systems literally do it for you.

For example, the Multi-Tool Budget expert system creates seven interrelated Multiplan worksheets for a total budget planning and control environment.

What's more, each system is developed by experts: business professionals and leading authorities in finance and accounting.

You'll benefit from their knowledge immediately, through the powerful worksheets each Multi-Tool expert system builds for you. And with the sophisticated tutorial manuals that accompany each system. Each manual provides in-depth information about both the design of the worksheets and the areas of finance and accounting they cover.



The result: a tailored electronic worksheet that helps you make high quality decisions.

That's just what you'd expect from Microsoft. The people who let you concentrate on your business rather than on your computer.

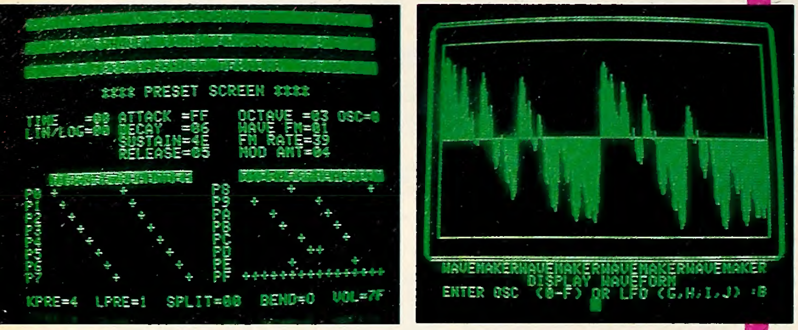
Ask your computer dealer to let you test drive the new Multi-Tool expert systems. Better tools that help you put your business in first place.

THE MULTI-TOOL EXPERT SYSTEMS. A POWERFUL ADDITION TO THE MULTIPLAN ELECTRONIC WORKSHEET.

Available now:
The Multi-Tool Budget expert system.
The Multi-Tool Financial Statement expert system.

BETTER TOOLS FOR MICROCOMPUTERS

MICROSOFT™



Passport's *Turbo-Traks* displays information about the sixteen presets in the Apple's memory (left), and plots the waveforms you've created with the *Wavemaker* (right).

With the *Wavemaker*, you can determine the shapes of the waveforms for the oscillators that go into synthesizing the sixteen presets. You can create waveforms by specifying the relative amplitudes of the fundamental and fifteen harmonics in 255 increments on a bar graph.

The *Wavemaker* also allows you to combine already created complex waves together to produce even more interesting sounds. Four low-frequency oscillators (in addition to the primary sixteen) can be used to control vibrato or phasing effects. Waveforms that you've created, or those that already exist as presets, can be plotted and altered.

You can play musical passages in real time on the Soundchaser keyboard at a chosen tempo, keeping your tempo by following a metronome click track. You can then play along with tracks you've laid down as they're playing back, simultaneously recording the track you're playing. In this way a maximum of sixteen separate, completely polyphonic, and simultaneously playing data tracks can be stored. Although you can record only a single track at a time, each track has individual controls for volume and preset assignment. Preset assignments may be varied track by track after they're recorded, so you can try out alternatives to the arrangement you used when the piece was originally recorded. That's a feature impossible to achieve on even the most advanced sixteen-track tape recorder.

The restrictions imposed by the sixteen-oscillator limit also apply when using the sequencer. The sequencer has an approximate capacity of twenty-eight hundred notes with a 48K Apple, and about double that with 64K. Other features include being able to transpose the entire playback up or down an octave with one keypress, and to loop the entire passage for continuous playback. The *Turbo-Traks* sequencer does not allow you at any point to enter into a track to change one part of it or to fix a mistake that may have occurred when you recorded the track. The only alternative in this case is to re-record the entire track and hope you get it right this time.

By being aware of them, a user can avoid the couple of small pitfalls the program can present. For example, going directly to the *Wavemaker* section of the program automatically erases any information contained in the sequencer's recording buffer. Remembering to merge all recorded tracks before going from the sequencer to other parts of the program circumvents this problem. Another thing to be aware of is how the program reacts if a language card is present. It automatically loads DOS onto the card, thus freeing about 11K of lower RAM for use by the sequencer. If you won't always be using the program on a 64K machine, this could cause problems. Sequencer files created on a 64K machine will load and play on a 48K Apple, but if you go to load or save something else you may obliterate the DOS files. To get around this you must remember to press the space bar as the program boots; this causes the program to ignore the language card and to treat the computer as if it were a 48K machine. Sacrificing the 11K of sequencer capacity is the tradeoff. Even better, of course, is to make sure you always use a 64K Apple.

Although the *Turbo-Traks* manual surpasses its predecessors from Passport in clarity and ease of use, it is not indexed and assumes the user has a prior understanding of sound synthesis. The program doesn't in-

GOTO 221

THE CAVERNS OF FREITAG™

For Apple II+ and Apple II e

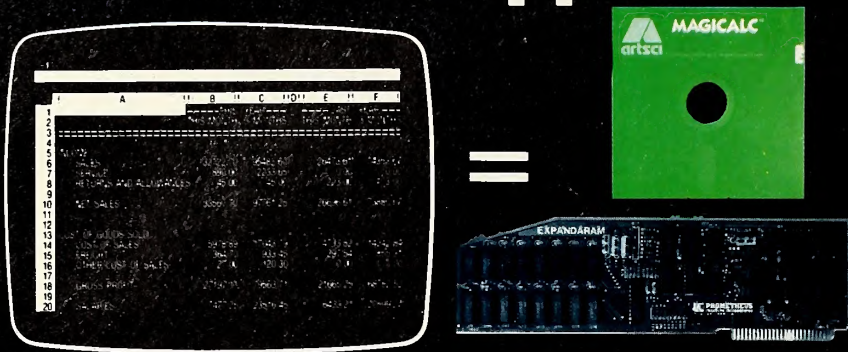
ANYONE CAN ENTER, BUT ONLY THE MOST SKILLED WILL COME OUT ALIVE.

MUSE®
SOFTWARE

347 N. CHARLES STREET, BALTIMORE, MD 21201/(301) 659-7212

Apple is a registered trademark of Apple, Inc.

the 2nd Generation Spreadsheet for Apple II™ and //c™



MAGICALC™ + EXPAND-A-RAM™

Everything VisiCalc can do and much more. Plus 64K or 128K RAM. Plus 80-column display for Apple //c at no extra cost.

MAGICALC is a completely new second generation spreadsheet program for Apple II. This state-of-the-art system includes 70-column upper and lower case video, full 80-column board display, hard disk compatibility, individual column widths, invisible columns for confidential data, and full compatibility with VisiCalc that lets you utilize existing VisiCalc models. Refer to the box below for a comparison of MAGICALC and VISICALC.

EXPAND-A-RAM 64K or 128K of additional RAM memory for expanded MAGICALC worksheets. Includes MEMORY MANAGEMENT SYSTEM — relocates DOS for more free memory in BASIC. You also get RAMDISK emulators for APPLE DOS, APPLE PASCAL,™ and CP/M™ — a software package that emulates a standard Apple floppy disk drive but is 2 to 10 times faster. Plus more.

Registered trade marks: Apple... belongs to Apple Computer, Inc., Magicalc to Artsci, Inc., Expand-A-Ram to Prometheus Products, Inc., VisiCalc to Visicorp, CP/M to Digital Research, Inc.

EXPAND-A-RAM for Apple II plugs into any slot and works with the many software programs that use 16K of RAM in slot 0 — including CP/M, APPLE PASCAL,™ LISA, MERLIN... and many others. EXPAND-A-RAM/80 for Apple IIe, which includes the 80-column display, goes into the auxiliary slot. No modification of your Apple is required.

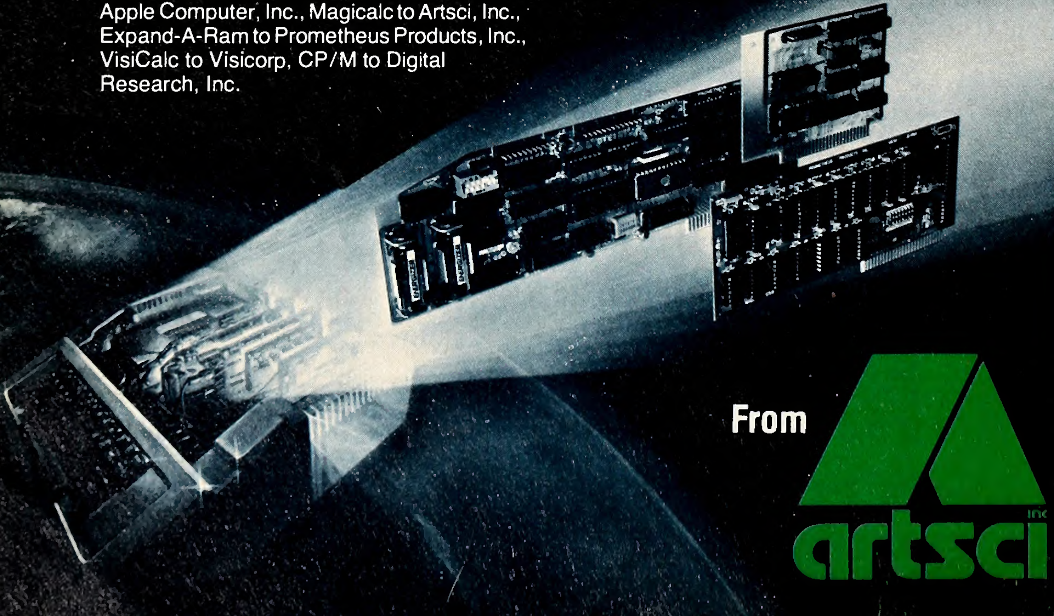
SPECIAL LOW INTRODUCTORY PRICES for the combination MAGICALC plus EXPAND-A-RAM are $399 with 64K RAM and $499 with 128K RAM. Apple IIe users (only), please specify your choice of the standard EXPAND-A-RAM or EXPAND-A-RAM/80 with 80-column Display at no additional charge.

See your local dealer for details:

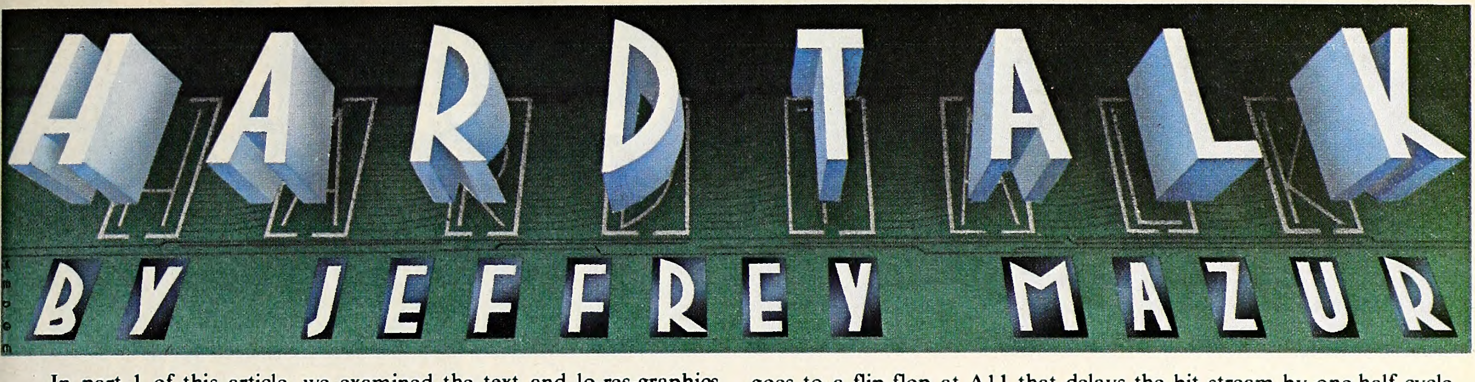
ARTSCI, INC.
5547 Satsuma Ave., North Hollywood, CA 91601, (213) 985-2922

PROMETHEUS PRODUCTS, INC.
45277 Fremont Blvd., Fremont, CA 94538, (415) 490-2370

COMPARE!	MAGI-CALC	VISI-CALC
Preboot required	NO	YES
70 column upper and lower case video display	YES	NO
Full 80 column board display	YES	NO
Individual column widths	YES	NO
Invisible columns for confidential data	YES	NO
Hard disk compatibility	YES	NO
Full compatibility with VisiCalc	YES	YES
Program plus 128K RAM	$499	$745



From artsci inc ... and PROMETHEUS



In part 1 of this article, we examined the text and lo-res-graphics modes of the Apple II. To continue our exploration of the Apple's video capabilities, let's look at the hi-res-graphics mode. In the course of our discussion, we'll discover why statements 2 and 3 from last time are false. Here are those two statements again, just in case you don't recall them right off.

2. True or false? When two adjacent dots on the hi-res screen are "turned on," the complementary colors from each dot blend in the eye to form a white spot.

3. Since the Apple's video signal is described as "NTSC-compatible," it should be possible to use the Apple with other video devices such as video-tape recorders, processing amplifiers, and so on.

As we said last time, the same circuits that are used to create the text and lo-res-graphics modes on the Apple are also used, with a few changes, to generate hi-res-graphics displays (see figure 1). For hi-res graphics, the first change is in the RAM MUX (multiplexor) section, which is where the memory addresses for the visual display are created. When the hi-res mode is in use, this section causes the address range between \$2000 and \$3FFF (or alternately \$4000 through \$5FFF) to be converted to video. This conversion is performed to create a "bit-mapped" image—that is, an image in which each bit corresponds to one picture element (pixel). Unfortunately, the bit-mapping is done in a rather complicated way. The figure on page 21 of the *Apple II Reference Manual* shows how the screen image is mapped into the Apple's memory.

The next change takes place at the shift registers B4 and B9. Due to the connections made through A8, these two four-bit shift registers are cascaded to form one eight-bit device. The shift clock rate is reduced to one-half that used by the lo-res mode, and this sets the dot rate at 7 MHz. The output of this dual-shift register leaves B4-15 (B4, pin 15) and goes to a flip-flop at A11 that delays the bit stream by one-half cycle. This is the "color-shift-bit" delay, which is used to get the alternate colors in the hi-res mode. Both the direct and delayed signals are fed into the video selector switch, A9. One of these signals is then selected as the video source, depending on the high-order color shift bit (follow D7 around into A8-13, A10-4, and finally into A9-11).

RED = HI-RES

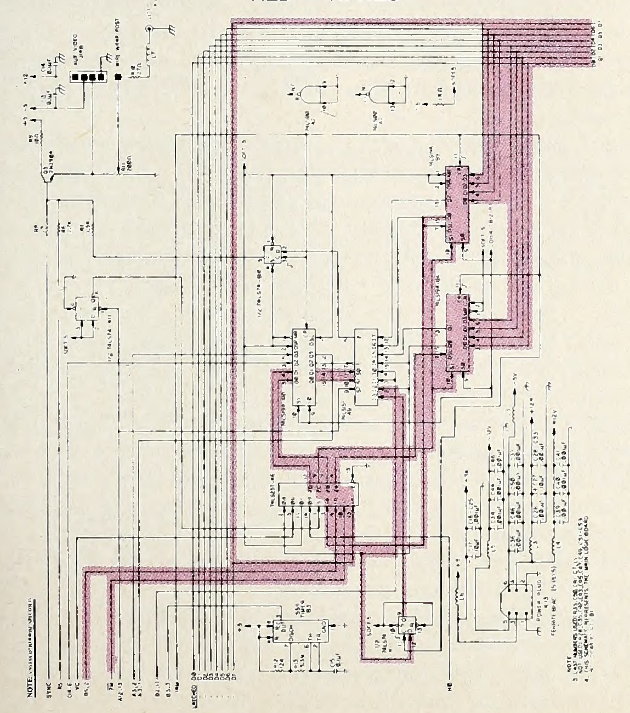


Figure 1. Schematic showing hi-res circuit path.

The Apple's video output, therefore, is nothing more than a continuous stream of bits being read out from memory. Note, however, that for each byte, the bits are shifted out to the left, starting with the least significant bit, D0. That is, D0 gets displayed first with D1, D2, and so on following to the right (remember the CRT scans from left to right). This is the reverse of the way binary data is usually written.

Whenever the video bit stream is high, this appears on the monitor as white. A low video bit stream appears on the monitor as black. When the video signal alternates, however, something special happens. Because the dot rate is exactly twice that of the color subcarrier frequency, any transitions appear to be chroma information. For example, if the entire screen memory is filled with zeros except for a single high bit, a color monitor displays this as a small colored dot. The exact color of the dot depends on two factors: the screen position and the color shift bit (CSB). On a properly adjusted monitor, single dots in the even columns appear as violet (CSB=0) or blue (CSB=1). Those in odd columns are green (CSB=0) or orange (CSB=1). Since there's only one CSB for every seven bits, some color combinations can only alternate every seven dots. Also, because an odd number of dots is created from each byte in memory, a solid-colored object is represented in RAM by alternating byte patterns.

Consider next the case of filling the screen with one solid color. For the entire screen to be one color, the CSB of each display byte must be the same. For a solid *color* (as opposed to black or white), the remaining bits within each display byte alternate between 1 and 0. Again, to keep the output bit stream consistent, every other display byte starts with the opposite polarity (ignoring the CSB of course). Here is how this might look in memory:

01010101 **0**0101010 **0**1010101. . . .

or, if you wish, \$55, \$2A, \$55, and so on.

The boldface numbers represent the color-shift bits and the remaining bits are what get displayed. Thus, the video output stream from this setup would be

1010101 0101010 1010101 . . .

or a continuously alternating pattern.

Let's see what this means from the NTSC composite video signal standpoint. The video signal in this case resembles a 100 percent, or 100-unit, square wave at 3.58 MHz. This is interpreted by a color monitor as a color signal whose luminance level is 50 units (that is, gray). The 3.58 MHz information is regarded as a highly *saturated* (100 percent) chroma signal whose *phase* represents the color violet. If the bit pattern had started with a 0 instead of a 1, then the phase would be shifted 180 degrees to form green (see figure 2). At any point where the color shift bit goes high, the remaining seven dots within that byte get displayed using the alternate hi-res blue and orange. This is because the CSB causes an extra ninety-degree phase shift to be introduced into the video bit stream.

Now to reveal why the second statement from last month is not the correct explanation of why a white spot is produced whenever two adjacent dots are turned on. The true explanation becomes obvious when

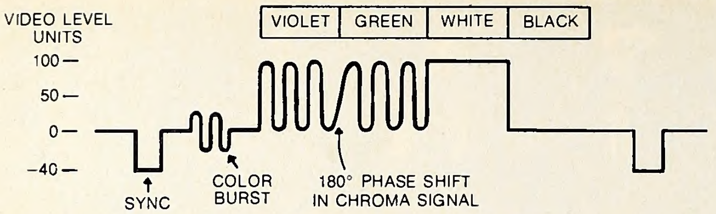


Figure 2. The Apple's composite video signal.

you look at the video signal such a pair of 1s produce. As the shift register clocks out the bits from memory, one high bit followed immediately by another causes the video signal to remain high for two consecutive dot clocks. This means that there is no transition at the 3.58 MHz color rate and therefore no chroma information for this portion of the video signal. Without any chroma information, the monitor simply treats the high video signal as a white area and displays it as such. Therefore, the fact that two adjacent dots produce a white spot has nothing to do with two complementary colored dots blending on-screen to form white (of course, any white spot is technically due to the mixture of light from the red, green, and the blue phosphors, but that's a different subject altogether).

Monitors. There are several ways of displaying the Apple's video signal. One popular approach is to use an RF modulator with a standard black-and-white or color television set. The modulator converts the video signal into a VHF (or sometimes UHF) channel that can be received directly by the television receiver. This offers the advantage of low cost, especially for color, assuming of course that you already have the television set. If the set is also used for normal television viewing, an RF switch can be used to select either the Apple or regular antenna signals. Using a color television set as a monitor presents several disadvantages, however, aside from possible conflicts with broadcast television viewing. The major one is limited resolution or clarity of text display. For this reason, computers (including the Apple) designed to be used with television receivers limit the number of characters per line to forty or less. Most of the "fuzziness" observed on these sets is due to the limited *bandwidth* of the receiver sections.

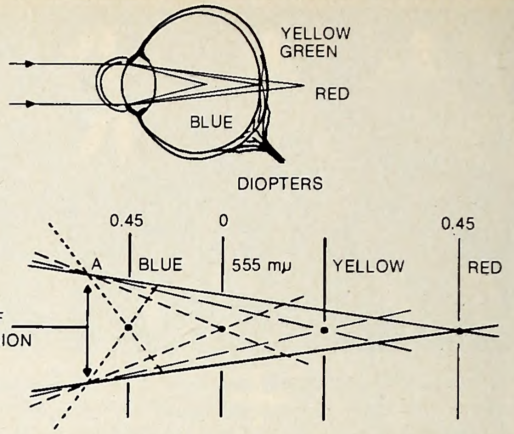


Figure 3. Chromatic aberration of the eye. Shortsighted and long-sighted implications of light/color frequencies and the retina.

Color television sets pose more serious threats to readability—dot size and misconvergence. Color picture tubes must use three different phosphors to produce the primary colors. These phosphors are laid down alternately on the face of the screen in very thin stripes or in triangular patterns called *triads*. The use of this process means that the minimum dot size and the spacing between dots is much larger than on similar monochrome CRTs. *Convergence* refers to how accurately the three separate beams within a color picture tube track each other. Where the convergence is perfect, the red, green, and blue beams land on the same spot, creating a "pure" white spot. Any deviation from this condition causes colored halos or shadows to appear.

A good-quality color monitor can be used to improve the quality of color displays. Eliminating the modulation and demodulation processes can also eliminate much of this degradation. However, the limitations of dot size and convergence still exist. The Amdek Color I is a good example of an NTSC composite color video monitor. It has 260(H)-by-300(V) resolution and also contains a built-in speaker and audio amplifier. For higher-quality graphics, and even eighty-column displays, direct RGB video can be used; more on this later.

For serious computer operation, a good monochrome monitor should be used. Several factors can reduce eye fatigue associated with prolonged computer use. Screen size, phosphor color, and video bandwidth should all be carefully weighed in terms of personal preferences, budget, and specific use.

Most Apple owners choose either a nine-inch or twelve or thirteen-inch monitor to fit on top of the computer. The larger monitor is usually preferred because its larger characters are easier to read. This size monitor is also very good to have when you're going to be using an eighty-column board.

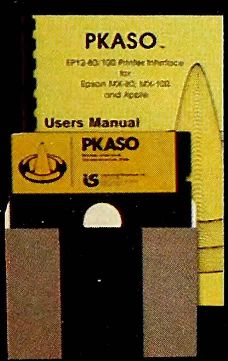
Many people find nine-inch monitors more convenient despite the smaller character size they provide (the characters a nine-inch monitor displays are still larger than the ones you get with a conventional thirteen-inch eighty-column display). Most nine-inch units fit quite comfortably along with two disk drives on top of the computer. They also offer a distinct advantage when portability is important.

One of the biggest debates about monochrome monitors concerns what color is best. For a while, most computer displays used standard black-and-white television picture tubes. These CRTs have phosphors, technically referred to by the designation P4, that emit a white light. White light, of course, is just an equal distribution of all frequencies (or colors) of light. In actuality, however, this light appears more like a combination of blue and yellow light, which combine to form a white image. This would not be of any consequence were it not for some flaws in normal human vision.

The human eye exhibits an optical property known as chromatic aberration. This means that the eye focuses differently on different colors of light (see figure 3). This would seem to imply, however, that everything we see should be surrounded by color fringes. A hypothesis explaining the lack of these fringes was published in 1939 by Gustavus Hartridge, who theorized that while yellow light focuses precisely on the retina, red and green focus slightly on either side. These out-of-focus

PKASO™
Printer Interface Family

Welcomes the New Apple //e... We work together.



PKASO Interfaces come complete with Cable, Instructional Diskette and Comprehensive Manual.

The PKASO family makes you and your Apple Computer a master of text and graphics.

PKASO makes it easy to use the features of your printer—select character sizes, vary line spacing, even print in colors. Simple PKASO commands make these features usable from the keyboard or a program.

PKASO also adds features to your system. Press a few keys and get a snapshot "dump" of the image you see on the screen—text or graphics. Add new characters and symbols that you couldn't print before, using our SuperFont™ system. Add our new PipeLine™ printing buffer and your printer can take its time while you and your Apple move on to the next task. The PipeLine is a modular add-on to the standard PKASO board.

The PKASO interface is designed for Apple II and Apple /// in all the popular configurations. It prints in full color on the IDS Prism Printer, and in striking black on C. Itoh, Centronics, Epson, IDS, NEC, and Okidata matrix printers.

NEW!

The IS Pipeline™ Printing Buffer with Random Access Printing stores paragraphs or pictures for printing in any order—any number of times!

- Universal—works with any parallel (Centronics style) computer/printer combinations.
- 8K to 128K Bytes of memory with data compression for efficient use of memory space.

Interactive Structures Inc.
146 Montgomery Avenue
Bala Cynwyd, PA 19004
Telephone: (215) 667-1713

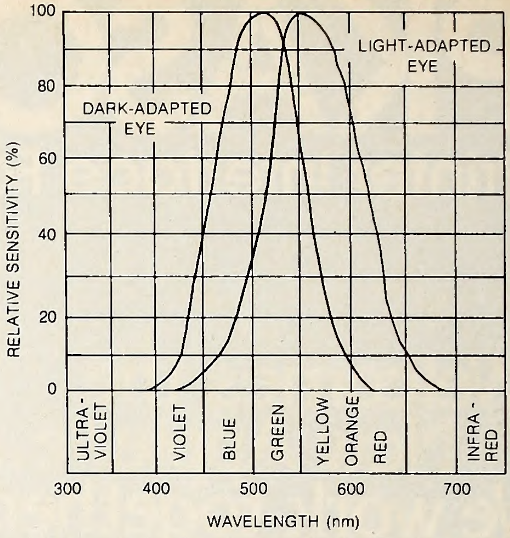


Figure 4. Sensitivity to light of dark-adapted and light-adapted eyes.

images cause sharp points of light to appear as larger "blurs"; the red and green blurs are about the same size, however, and thus tend to form a single, fuzzy, yellow spot. Therefore, while this chromatic aberration creates a decrease in visual acuity, it does not result in a color distortion. By the way, the blue end of the spectrum is not considered because it is so poorly focused. This causes its image to be relatively dispersed and perhaps even below the threshold of vision.

From this discussion, it would appear that a yellow phosphor would make the ideal monitor. This also correlates well with the spectral sensitivity of the eye, as shown in figure 4. The peak of the curve in the yellow region shows that the eye is most sensitive to the color yellow. To verify this conclusion, several studies have been performed. One German group compared the performance of computer users working with green, yellow, amber, and black-and-white screens. The results indicated that a yellow screen with an amber filter resulted in performance figures more than four times higher than those obtained using a black-and-white screen.

One of the few companies offering amber monitors in this country is USI International. Its Pi-3 (twelve inches) and Pi-4 (nine inches) are amber equivalents to its Pi-1 and Pi-2 green-screen monitors. These monitors also feature a video invert switch, which allows for either the traditional amber letters on a black screen or black letters on a solid amber screen. The Pi-4 even has switchable DC restoration (to keep the brightness level constant regardless of what image is being displayed) and a video/data switch that increases the sharpness of characters when computer data is being viewed.

Many companies now offer green-screen monitors that use the popular P31 phosphor. Some monitors employ the P39, also a green phosphor, but with a longer persistence. This reduces any flicker that might otherwise be noticed with our 60 Hz field rate. But flicker is rarely a problem anyway, and the longer persistence can sometimes cause blurring of fast-moving displays (such as program listings).

RGB Color System. The NTSC composite video system was designed for simple transmission through coaxial cable or as a broadcast signal. Therefore all of the information carried by this signal is packed together as efficiently as possible. This is accomplished partly at the expense of bandwidth (resolution) and certainly adds complexity, which inevitably lowers the signal quality.

Since most computer displays are located inside or within the immediate vicinity of the computer, the advantages of this system aren't as important. If the entire encoding/decoding process could be eliminated—so that separate red, green, and blue signals were fed directly into the CRT—a display of much higher quality could be obtained. For one thing, colors would be much more vivid as a result of their increased purity and clarity.

Top-of-the-line RGB monitors such as the Amdek Color IV make possible incredible color displays and crisp eighty-column text displays. This is because of the use of analog video circuitry and a CRT with exceptionally fine dot structure.

HAVEN'T YOU HEARD OF THUNDERCLOCK PLUS™?

If you want to put your Apple® to work—around the clock—Thunderclock Plus is the solution. Just plug it in and your programs can read the month, date, day of week and time—down to the second—in any of Apple's languages. So your Apple can do any number of tasks for you automatically. In the office, the lab or at home.

Most good software packages for business, data base management, communications and time management are made to read Thunderclock Plus. (It's compatible with DB Master,* Micro-Courier** and VisiDex†, to name a few). So no matter how you use your Apple now, Thunderclock Plus can make it a more versatile and efficient tool.

For example, with business or communications software, your Apple can automatically access a data base or send electronic mail when the rates are lowest.

In addition, Thunderclock Plus can organize your disk files. Our optional DOS-DATER™ software upgrades the regular DOS on your disks. So every time a program is saved or a file is modified, the time and date, *to the minute,* are stored in the CATALOG with the file name. Now you can instantly know exactly when your files were last updated.

Thunderclock Plus can even give you a sense of security. Or just make your life a little easier. With our X-10 interface option and a BSR X-10‡ Home Control System, your Apple can turn on your lights, water your lawn...whatever you desire, according to schedules you create. It comes with our menu-driven SCHEDULER software. So it's easy to design and modify schedules that can run in the "background" while you have "hands-on" use of your Apple.

Thunderclock Plus comes with a one-year warranty. Powered by on-board batteries, it runs accurately for up to four years without battery replacement.

So now that you've heard of Thunderclock Plus, isn't it time you put your Apple to work—around the clock? See your dealer for a demonstration or contact us.

THUNDERCLOCK PLUS and BASIC software	$150
DOS-DATER/DEMO disk	$ 29
X-10 Interface option	$ 49
PASCAL software disk	$ 29

®Apple is a registered trademark of Apple Computer, Inc.
*DB Master is a registered trademark of Stoneware, Inc.
**Micro-Courier is a registered trademark of Microcom.
†VisiDex is a registered trademark of VisiCorp.
‡BSR X-10 is a registered trademark of BSR (USA) Ltd.

Thunderware's DOS-DATER time and date stamps your disk files to the minute.

THUNDERWARE, INC.
44 Hermosa Ave., Oakland, CA 94618 (415) 652-1737

Many computers, including the Apple III and the IBM pc, have circuitry to drive an RGB monitor. Since the Apple II has no such provision, a peripheral card is needed to generate the necessary signals. Amdek manufactures such a board; it's called the DVM, or Digital Video Multiplexor. The present version of the board is rather complicated to install and offers only digital control of the three primary colors. These "channels" are controlled by several soft switches that can be turned on and off via simple poke statements. Two other switches select either the standard Apple text video or the output from another video device, such as an eighty-column board. Connection to an eighty-column board is also rather complicated, and detailed instructions are given only for the Videx Videoterm. People who own other eighty-column boards are on their own.

The DVM board works by intercepting the graphics video data as it is read out from RAM. This is accomplished via connection to the data latch ICs B6 and B7, which are placed into a small daughterboard, or adapter socket. Another daughterboard is inserted at the video counters D11 through D14 to pick off the necessary timing signals.

The DVM interprets the RAM data in a different way than do the Apple's graphics generating circuits. Instead of serializing the data to form a composite color signal, the DVM handles the data in four-bit (lo-res) or two-bit plus color shift bit (hi-res) parallel bundles. These groups of bits are then used to turn on the appropriate electron guns of the color CRT. The lo-res mode is limited to eight colors: black, red, green, yellow, magenta, cyan, blue, and white. These are the only possible colors with digital RGB video; they represent all combinations of the three guns being on or off. In the hi-res mode, all of these colors except yellow and cyan are available.

Amdek has announced the release of an improved version of the DVM. Called the DVM-II, this board features much simpler installation, analog intensity control, and better compatibility with eighty-column boards.

For display, Amdek offers a choice of three RGB monitors: the Color II, Color III, and Color IV. The Color III monitor, which is the least expensive, offers 260-by-300 line resolution and should provide satisfactory performance with eighty-column cards that use a 5-by-7 character matrix. Output from cards that use a 7-by-9-or-larger matrix may be subject to smearing or other distortions on the Color III. The Color II offers better resolution (560 by 240) and slightly more accurate convergence. It also has an *intensity* input that increases the number of possible colors. This monitor can produce striking graphics displays and is usable with all eighty-column cards (5-by-7 dot matrix is still preferable). For most users, the DVM-II/Color II combination should provide excellent results.

If higher-resolution RGB graphics boards or a board with analog color control should become available, the Color IV monitor ought to be considered. It has a resolution of 720 by 420 and its analog circuitry can produce an infinite range of colors. Up to ninety-six 5-by-7 characters can be displayed on each line. The Color IV should provide superior performance, even when used with the high-resolution eighty-column cards.

More Eighty-Column Text Cards. While we're on the subject of video and eighty-column text cards, a couple of newcomers deserve attention. The first is the Viewmax-80 from Micromax. This board offers basic eighty-column operation with software video switching, a 7-by-9 character matrix, and support of the game-port shift-key modification. In fact, the firmware automatically attempts to determine whether the shift-key mod is present. Particular care has been taken to make this board software-compatible with the Videx Videoterm. The Viewmax-80's main selling point, however, is its extremely low price. It's also backed with a two-year guarantee.

For a more powerful board with advanced features, there is the SuperVision from Techcom Enterprises. This unit offers superior display characteristics, including inverse, flashing, underline, half intensity, and an alternate character set. The flashing is a true on/off flash, not the alternating normal/inverse flash used by the Apple. All of these attributes are available, in any combination, on a character-by-character basis. This is accomplished by using a twelve-bit word to store each character. Other features include line insert or delete, debug-character-set mode, and GOTOXY cursor positioning. As soon as word processing programs are modified to take advantage of the SuperVision's expanded features, it will finally be possible to see an eighty-column document with underlining, boldface, and so on right on the screen—just as it will appear when printed out. Gone (forever, let's hope) will be the messy control characters or formatting commands that just clutter up a document when it is displayed on-screen.

Although the documentation is a little thin, some information on the firmware's operation and access to the CRTC registers is included. This makes it possible, for example, to change the size of the cursor or its blink rate.

Apple IIe. One of the enhancements the IIe offers is its eighty-column capability. Instead of requiring an entirely separate video board to generate eighty columns, the motherboard contains most of the necessary circuitry. In fact, activating a simple soft switch places the IIe in eighty-column mode. What actually happens is that the dot-clock frequency is doubled to send out the characters faster. This in effect causes the characters to appear half of their normal width. Since the cpu clock is not changed, there's time to put out two characters for each cpu cycle. The video circuits can still only make one memory access during this time, however, so when no circuitry has been added, the Apple just spits out two copies of each character.

Installing an eighty-column text card (either the standard card or the expanded memory one) in the special slot 0 makes a new mode possible. If the memory on the text card is used, two characters can be stored and displayed for each standard location. This added memory can be thought of as residing "on top of" the normal screen memory from $400 through $7FF. Now when the video circuits grab a byte of memory from this location, they actually grab two bytes. One is stored temporarily while the other is being displayed (in one-half the time used for a forty-column character). Then the second byte from the text card RAM is sent to the character generator for display. Thus the characters displayed on the screen alternately come from the motherboard and text card RAM. This technique makes eighty-column operation much simpler on the IIe than it is on the counterpart IIs. It also opens the door to double-resolution graphics, which should be possible with later versions of the IIe.

Video Compatibility. Our final topic this month concerns the compatibility of the Apple II (and IIe) video signal with other NTSC standard equipment, including video-tape recorders (one-half-inch, three-quarter-inch, and one-inch), switchers, processing amplifiers, and other broadcast equipment. Although the Apple video signal is described as NTSC-compatible, monitors are just about the only equipment you can expect this to hold true for. Half-inch (that is, Beta and VHS) and three-quarter-inch (U-matic) VCRs usually record a decent black-and-white signal from the Apple, but the color signal may be distorted, scrambled, or nonexistent. Pre-Revision 7 boards with overly wide sync will probably show no color at all. Production switchers and many proc amps (processing amplifiers, used to clean up distorted video signals) will not accept the Apple's video signal.

The reason for this mess stems from the simplified approach the Apple takes to generating video. The Apple's only rigid adherence to NTSC specifications is with the color subcarrier frequency. From there, however, the Apple deviates just enough to produce compatibility problems.

In the true NTSC standard, the 3.58 MHz signal is divided by 227.5 to derive the horizontal timing. Vertical field rate is produced by dividing the horizontal timing signal by 262.5. As a consequence of these figures, the video signal acquires several characteristics. First there is the odd number of lines per frame—525. This means that each field contains a half line either at the beginning or end of the field. This is done to allow the fields to mesh, or interlace, within each other.

The Apple, on the other hand, generates an even number of lines each field (262), for a total of 524 lines per frame. The Apple video is therefore noninterlaced. Because of this, making reference to both *field rate* and *frame rate* is somewhat redundant. Each field *is* a complete frame, and therefore both consist of 262 lines and get updated approximately sixty times per second. To remain consistent with the standard NTSC jargon, however, two consecutive fields are often referred to as one frame.

If you absolutely must make the Apple fully NTSC-compatible, you have only two possible choices. One is to add a special video board that generates proper video completely from scratch. Several such boards are available at prices in the thousands of dollars.

Video Associates Labs offers the models VB-1 and VB-3 at $1,850 and $2,400 respectively. Both of these boards "genlock" to an existing video source and then superimpose the Apple graphics display on top of it. The VB-3 has a built-in sync generator and can therefore stand alone as a video source for video taping. The VB-3 also has a built-in color background generator with full program control of luminance, hue, and saturation. Even the horizontal and burst phases of the video signal are adjustable from software.

The Adwar ARS-170 from Adwar Video Corporation is another device that can be used to "standardize" the Apple video. Among other things, it converts the 524 lines-per-frame Apple video into the proper 525-line format. The signal from this board may still not be *completely* standard, despite the $2,800 price tag for the ARS-170.

Another company that has shown interest in Apple-NTSC compatibility is Symtec. This company originally made a simple horizontal sync corrector board that improved the sync timing of older Apples. This was primarily done to bring the color burst into its proper time slot so that other equipment could recognize the Apple's video as a color signal. Since the Revision 7 motherboards also corrected this problem, this product became obsolete.

Currently Symtec is manufacturing a $7,000 package called the PGS III, or Professional Graphics System. The PGS III is a complete rack-mounted graphics generator that just happens to connect to an Apple. It uses 32K of its own memory to store images with 512-by-192 resolution in any of more than four thousand colors. Both hue and intensity can be varied, and a special "transparent-pixel" mode allows computer-generated text and graphics to be mixed with scenes from an external video source.

Adwar Video Corporation, 100 Fifth Avenue, New York, NY 10011; (212) 691-0976. Amdek Corporation, 2201 Lively Boulevard, Elk Grove Village, IL 60007; (312) 364-1180. Micromax, 13731 Mercado Drive, Del Mar, CA 92014; (619) 569-2121. Symtec, 15933 West Eight Mile Road, Detroit, MI 48235; (313) 272-2950. Techcom Enterprises, 16819 South Hawthorne Boulevard, Lawndale, CA 90260; (213) 854-0583. USI International, 71 Park Lane, Brisbane, CA 94005; (415) 468-4900.

Save the Dendoron Galaxy from . . .

THE DESECRATION

FIRST IN A SERIES OF ADVENTURECADES

A Mind Games first . . . the **ADVENTURECADE™**, the only computer game that intersperses three arcades within an adventure! As an added bonus, the arcade sections may be played separately from the adventure.

from

progame™

a division of
CALIFORNIA PACIFIC COMPUTER
757 Russell Blvd., Davis, CA 95616
(916) 756-2921 or (916) 756-2923

mind games

Ask your dealer or order directly.
VISA or MASTERCHARGE accepted.

MIND GAMES, Inc.
420 South Beverly Drive., Suite 207
Beverly Hills, CA 90212
(213) 277-8044, Extension A-1

Written by Greg and Gil; Assembly Language (48K); Requires an Apple II or II+ computer with DOS 3.3; Apple II is a trademark of Apple Computer, Inc.

III BITS

JOHN JEPPSON's Guided

ZERO PAGES

USER BANKS

GRAPHICS BUFFERS

TEXT LAYOUT

Here is a ragtag assortment of odds and ends from Apple III, thrown together almost untainted by logical sequence. Some have already been published elsewhere; some are obtainable in the fine print of Apple manuals; and some are the fruit of personal investigation. Accuracy, particularly in the latter category, may not be uniformly high. So be warned.

Let's face it. Extracting information from Apple Computer isn't the easiest thing in the world. In fact, it's usually faster, and more fun, to ask the Apple III itself. There is no obvious reason for Apple's reticence. The folks at Apple intend, they say, to publish the *SOS Reference Manual* and, eventually, the *Driver Writer's Guide*. The reference manual exists already, more or less, as a textbook for the *Apple III Technical Workshop*. If you're keen on writing assembly language for the III, by all means take that course. It tells you lots and lots, although not quite everything you might wish to know. Inquiries, however, seem to drift off into never-never land.

Why so secretive? The effect seems primarily to impede the efforts of would-be Apple III programmers, which you might suppose would not be in Apple's interest. Maybe they are protecting something else? The techniques of RAM-based operating systems may have a more general applicability . . . perhaps to the Lisa?

No doubt a recognized software development firm, prepared to sign certain agreements, can obtain source materials and technical assistance. But many bright ideas must incubate and grow and be played with on the machine before they become sufficiently clear and explicit to warrant a formal approach to Apple. A lot of maybe-we-coulds must simply have vanished because the programmer had insufficient information to permit experimentation.

A Memory Map. At any one time, the 6502 cpu works with 64K addresses arranged as follows:

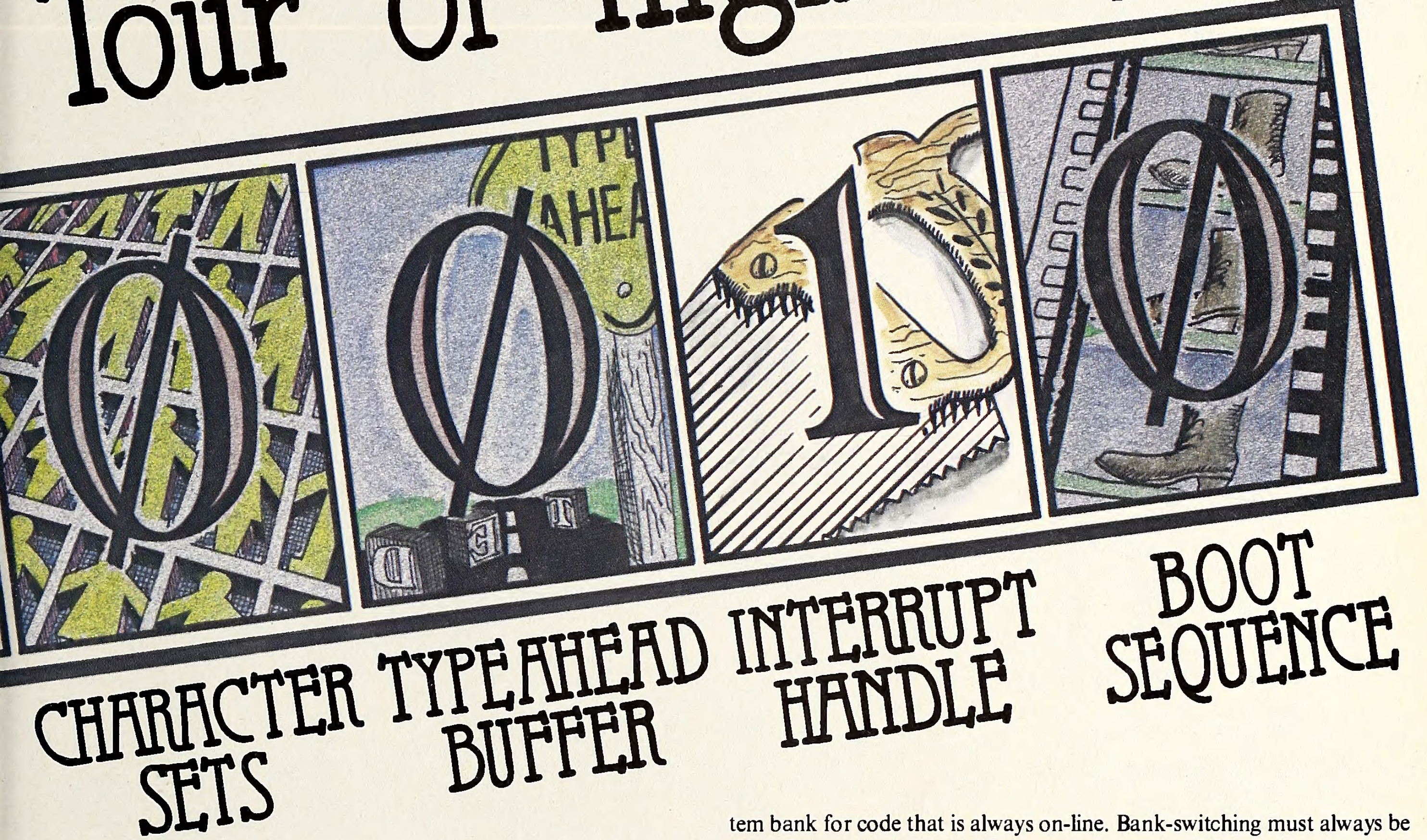
0000..1FFF lower s-bank	2000..9FFF user bank 0-6	A000..FFFF upper s-bank

The system bank is always on-line. It contains SOS.Kernel and other goodies. The user banks are switched in and out. Only one user bank is on-line at any given moment.

Table 1 describes the function of pages in the lower system bank.

Upper System Bank: $A000..FFFF. SOS.Kernel occupies $BC00..FFFF. In the future SOS.Kernel may get longer and extend down as far as $B800. SOS.Interp, which is "absolute" code, is normally loaded below SOS.Kernel. Actually, it is loaded into the highest user bank (bank 6 in a 256K machine) beginning at some predetermined location ($7600 for Pascal). It may then extend upward for any length, up to the lower end of SOS.Kernel (presently $BC00). Thus it usually overlaps from bank 6, a user bank, into system bank $A000..BBFF. This is important because the overlap gives the interpreter a sizable area in sys-

Tour of Highway III



00:	"True" zero page. Used early in boot sequence, and as the zero page for interrupt handlers.
01:	"Normal" 6502 stack. Addressed by PHA, JSR, and so on, whenever bit 2 of environment register ($FFDF) is set. Used as stack page by interrupt handlers, drivers, and by SOS.Kernel itself.
02..03:	I/O buffers for floppy drivers.
04..07:	Text page 1. In eighty-column mode holds screen memory for even-numbered columns 0,2,..78 (decimal).
08..0B:	Text page 2. Memory for odd-numbered columns 1,3,..79 (decimal). Note: Corresponding addresses in TextPage1 and TextPage2 are interchanged by the relation: (high byte) XOR $0C.
0C..0F:	Character set.
10..11:	File names, prefix, ? access routes to files.
12..13:	Used as I/O buffer for reading directories.
14:	Xbyte page when zero page is $18. Used by SOS.Kernel and by drivers.
15:	Typeahead buffer.
16:	Xbyte page when zero page is $1A. Used by interpreter and by assembly modules included in user programs.
17:	Keyboard layout.
18:	System zero page. Used by SOS.Kernel and by drivers.
19:	SOS data and jump tables.
1A:	"User" zero page. Used by interpreter.
1B:	"Alternate" 6502 stack when zero page is $1A (zero page XOR $01). Used by interpreter. Alternate stack is addressed by PHA, JSR, and so on, whenever bit 2 of environment register ($FFDF) is clear.
1C..1D:	Route information for open files.
1E..1F:	Available for use by interpreter.

Table 1. Lower system bank: pages $00..1F.

tem bank for code that is always on-line. Bank-switching must always be done from system bank. If you switch banks while running in a user bank—puuff! Suddenly you aren't.

A very short interpreter might lie only in user bank or only in system bank. The loading site and length are determined by the writer when the interpreter is created. It is absolute code ".org'd" on the intended loading site.

Usually the upper system bank is all RAM, except for $FFD0..FFDF and $FFE0..FFEF, which are the onboard D and E VIAs (versatile interface adapters). In particular, if bit 6 of the environment register is clear, then $C000..CFFF is RAM. If that bit is set, then this area is I/O. There are also $20 bytes of RAM "under" the VIAs at $FFD0..FFEF. Normally they are off-line. These RAM bytes can be accessed only by "8F" extended addressing. This small area of RAM is unique in that it is not disturbed by a control-reset reboot, so that's where they keep the last valid clock value—less useful, since you obtained your functioning clock chip.

The RAM of SOS.Kernel area $C000..FFFF can be write-protected by setting bit 3 of the environment register. Normally this RAM is protected while the interpreter is running. This is the user environment, and Apple doesn't trust you. It is unprotected in the driver, SOS.Kernel, and interrupt handler environments. Write-protection doesn't affect I/O $C000..CFFF when that is enabled, nor the VIA registers $FFD0..FFEF.

Highest User Bank: Bank 6 in a 256K Machine. At boot time SOS.Interp is loaded here, at whatever site the writer has designated, assuming, as is usually the case, that the interpreter is not confined entirely to $A000..BBFF in system bank. Next the drivers are loaded below SOS.Interp, one after another, in whatever order they are encountered in SOS.Driver, which is just the reverse of the order listed by the *System Configuration Program.* For this purpose a modular driver is just one

TODAY'S PRODUCTIVITY IS TOMORROW'S PROFIT.

That's why Quark designs office automation software to increase your productivity. With programs that will dramatically enhance your Apple*III or IIe. So you'll do more work—more efficiently—in less time—for less money.

And all Quark programs are extremely friendly. So you can devote your efforts to using them, not learning them. Yet you'll discover many sophisticated features no other software can offer.

Find out for yourself. See your dealer today for a complete demonstration of our Word Juggler™ word processing system. And innovative utilities programs, such as our Catalyst™ hard disk boot.

Because tomorrow belongs to the productive. And who wants to be left behind in yesterday?

Quark Incorporated,
2525 West Evans, Suite 220,
Denver, Colorado 80219,
(303) 934-2211

Quark™
INCORPORATED
Office Automation
Software

driver, no matter how many modular units it may contain. The drivers extend, if necessary, down to the bottom of bank 6. If more space is needed, they continue from the top of the next lower user bank ($9FFF with bank 5 switched in). Each driver, however, must be completely contained within its bank, so if there is insufficient room for the complete driver in bank 6 it will be placed entirely in bank 5. Any space left over below the drivers is free and available for requisition by the interpreter.

Interpreter Strategy. Interpreters should never assume they are resident in the highest bank, even though that is where they are normally loaded. Interpreters will run perfectly well in other banks. All you need to do is copy the *user bank portion* of the interpreter into the corresponding bytes in another (free and available) bank. Then switch in that bank (being careful not to self-destruct) and perform a jump to the first byte of interpreter code. The interpreter still overlaps into system bank ($A000..BBFF) just the way it always did. Bank-switching affects only the user bank. Interpreters, therefore, should always find out where they are by checking the bank register ($FFEF), never by making assumptions or by using location $1901, which does contain the highest bank number.

This relocation trick can be used for interpreter-switching schemes. A small switching interpreter is loaded from disk as the original SOS.Interp. It is placed entirely in the upper part of the highest user bank, with the drivers immediately below. The switching interpreter, in turn, loads in another interpreter (perhaps Pascal) but places the user-bank portion in a lower bank, where it runs very happily. The switching interpreter remains in the highest bank, taking up very little room, just waiting for you to call it back by pressing a special key combination (for which you will need a small modification of the console driver). Then the switching interpreter can load in, and run, some other interpreter, such as Basic.

User Bank 0: "8F" Addressing. The "zero-page anomaly" in Apple III means that every time the 6502 executes a zero-page instruction it actually operates on the designated zero page, found as the value of the zero-page register ($FFD0). The same thing happens when the (sixteen-bit) operand of an instruction has $00 in the high byte, since that also refers to zero page. Similarly, in extended addressing, if the place you are headed has $00 in the high byte of its address, then that is also interpreted as a zero-page location and you are given whatever is the current zero page. But that may not be what you want.

Extended addressing looks at a 64K stretch of memory comprising two consecutive user banks. In extended addressing, $0000 is the bottom of one user bank and $FFFF is the top of the next higher user bank. Whichever pair of user banks is active depends on the Xbyte of the extended address. This works fine except for the lowest page of the lower of the two user banks in the pair. That would have to be addressed as $00*xx*/8b. But the high byte of that address is $00, so you are given zero page instead.

Normally you get around the zero-page anomaly by decrementing the Xbyte. Then you are looking at a different pair of user banks, with your target bank as the higher member of the pair. The lowest page of that bank can then be addressed as $80*xx*/8b−1. But what about the lowest page of bank 0? There is no lower user bank to put underneath it in a pair. Hence 8F addressing.

8F is an Xbyte which, when present, causes the extended addressing mechanism to look at 64K of memory constructed as follows:

0000..1FFF	2000..9FFF	A000..FFFF
lower s-bank	bank 0	upper s-bank

It is almost exactly like system addressing (Xbyte $00), with bank 0 switched in. (There is one other interesting feature. It is all RAM, including the RAM beneath the VIA registers $FFD0..FFEF.)

Thus if you are doing a lot of talking to user bank 0, you should use 8F as the Xbyte and address the bank as $2000..9FFF, corresponding to $0000..7FFF in the bank. Then you can get to the lowest page without worrying about the zero-page anomaly.

User Bank 0: Graphics. Which, of course, is why you'll want to be talking to bank 0. That's where graphics are, when graphics are allocated. Pascal and Basic each provide for allocations of $00, $40, or $80 pages of graphics, depending on the graphics mode. Pascal also allows $20 pages, which is enough for one lo-res black-and-white buffer. Hi-res mode appears to interleave two lo-res modes in alternate columns or groups of pixels, much as eighty-column text interleaves two forty-column screens.

The number of pages allocated for graphics is stored in location $1907 in the SOS data area. Presumably this byte is used by the video generator apparatus, as are surrounding bytes in that area.

In black-and-white lo-res mode (BW280), buffer 1 runs from $2000/208F..3FFF/3F8F (which is $0000..1FFF in bank 0). Buffer 2 is found in $4000/408F..5FFF/5F8F. In buffer 1 the lowest byte ($2000/208F) represents the upper-left corner of the screen. Each bit represents one pixel. Successive bytes (and their contained bits) are in order from the left edge of the screen. One accesses individual bytes by indirect Y-indexed addressing (extended addressing) off the base address, which is the leftmost byte in that horizontal row. The following algorithm (for BW280, buffer 1) relates corresponding bytes in successive rows. It was discovered empirically and is doubtless pathetically slow:

```
next line up:    subtract $0400
                 if < $2000 then add $2000 and subtract $80
                    if < $3C00 then add $0400 and subtract $28
next line down:  add $0400
                 if >= $4000 then subtract $2000 and add $80
                    if >= $2400 then subtract $0400 and add $28
```

In hi-res mode (BW560) alternate bytes (groups of eight pixels) come from corresponding bytes of the two lo-res buffers just discussed. Thus the sequence is $2000, $4000, $2001, $4001. . . . Hi-res buffer 2 is the corresponding structure beginning at $6000/608F. In color mode (CP280) the "upper lo-res buffer" presumably contains color information. We are not sure about COL140 mode. And we are not sure if the base address algorithm holds for these modes.

When you are ready for the video generator to display your graphics, it is necessary to fiddle with the soft switches (see table 8). Graphics information is always taken from bank 0, regardless of which user bank is switched in. Presumably this is hard-wired, although it is just conceivable that the source bank is software-selected. If so, we don't know how.

The Text Pages: $0400..07FF and $0800..0BFF. Apple III text memory is very similar to Apple II; possibly identical for forty-column mode. In eighty-column mode the two "text pages" are interleaved: even columns from $0400..07FF, odd columns from $0800..0BFF. The reason for this peculiar arrangement is found in the direct memory access (DMA) apparatus of the video generator. When Apple III was designed for an eighty-column display, the video generator had to call up twice the amount of information as it did for the forty-column display of Apple II. But it did not have twice the time in which to do it. So the memory-fetch path in Apple III was made sixteen bits wide. Every data fetch actually gets two bytes. The video generator uses both. The 6502 chip uses one and ignores the other (except in the case of the Xbyte, which is that extra byte used in extended addressing). The memory fetch does *not* get two adjacent bytes. It gets the byte at *address* and the byte at *address: high byte XOR $0C*. Thus a fetch to $0400 also gets the byte from $0800, which the video generator puts in the odd column. And this is also the reason why the Xbyte page is related to its zero page by the same relation, high byte XOR $0C.

The Apple II "screen holes" are there, but you aren't supposed to use them for peripheral card scratchpad space. In the Apple III these locations are used as transfer ports when downloading character sets to the video generator. But downloading occurs only at boot time or when programs deliberately change character sets. It is relatively rare. The rest of the time these locations seem to be idle. It may be that a peripheral card could use them for a while. But it's illegal according to the definition of Apple III.

It is possible to write directly to the screen from assembly, bypassing the console driver. Just put ASCII codes in the appropriate memory locations. The high bit should be clear for inverse and set for normal, assuming you are using a standard (not inverted) character set.

The bytes in each horizontal line are accessed by X-indexed addressing off the base address, which is the leftmost byte of that line (see table 2).

If column is odd, add $0400 to the address.
Use X index := column DIV 2;

00	0400	08	0428	10	0450
01	0480	09	04A8	11	04D0
02	0500	0A	0528	12	0550
03	0580	0B	05A8	13	05D0
04	0600	0C	0628	14	0650
05	0680	0D	06A8	15	06D0
06	0700	0E	0728	16	0750
07	0780	0F	07A8	17	07D0

Table 2. Text screen line numbers versus base addresses.

In eighty-column mode use X := column DIV 2. If the column number is odd, you must also add $0400 to the base address given in the table. Alternatively, the base address and index may be computed with a modification of the Apple II subroutine Bascalc (table 3).

```
         Entry:  Line, column
         Exit :  Base address (addrL,H),  X-index
bascalc  lda     line
         pha
         lsr     A
         and     #03
         ora     #04
         sta     addrH
         pla
         and     #18
         bcc     $1
         adc     #7F
$1       sta     addrL
         asl     A
         asl     A
         ora     addrL
         sta     addrL
which    lda     column
         lsr     A
         tax
         bcc     $2
         lda     addrH
         eor     #0C
         sta     addrH
$2       rts
```

Table 3. Subroutine Bascalc.

The Character Set: $0C00..0FFF. At boot time the system character set is loaded from SOS.Driver and stored in these pages. Similarly, if you download another character set from a program by issuing a DControl call #16 or #17 to the console driver, the new set is also placed here. But these pages are not the active character set in current use by the video generator. This is merely a staging area. From these four pages the character set is further transferred to the video generator's storage area, wherever that is. It is *not* in addressable memory. Presumably the machine contains a 1K RAM chip dedicated for this purpose, analogous to the ROM chip beneath the Apple II keyboard that contains the character set for that machine. In any event, you can change the copy in $0C00..0FFF all you wish, but nothing happens.

The console driver uses a complex mechanism to transfer the character set into the video generator. It sets up an interrupt-driven background program (spooler) by allocating system internal resources (SIR) numbers $05, $06, and $10. The video-generator mechanism then interacts with the SIR#06 interrupt handler (embedded in the console driver) to transfer the character definitions at its leisure. The computer's attention is returned to the user's programs, and the video generator interrupts when it feels ready for another swallow. There may be simpler ways if you are willing to let the main program wait. For an entire character set the download procedure takes about a second to complete.

The actual transfer involves the E-VIA's peripheral control register ($FFEC), interrupt enable register ($FFEE), a couple of sites in $C000 I/O space ($C0DA and $C0DB) that are probably soft switches, and the notorious screen holes. Apparently the interrupt handler moves the character descriptions piecemeal from the $0C00 area to the screen holes and then alerts the video generator to move them on from there into its own dedicated RAM.

This transfer could probably proceed just as easily from any memory buffer to the screen holes; the $0C00 staging area is merely a convenience. But if you are operating from a background program, and if that program is the interrupt handler itself, then the buffer must be in system bank. If the buffer were in a user bank it would surely go off-line due to bank-switching. Extended addressing is not available for interrupt handlers; it doesn't work on the true zero page. Hence the $0C00 buffer.

Typehead Buffer: $1500..15FF. Page $15 is set aside for use by the console driver as a typeahead buffer. It is nothing more than a first-in-first-out queue. Actually two queues. The first queue ($1500..157F) contains KBD values, which are the ASCII codes generated by the keyboard. For each KBD there is also a KBDFLG byte, the second keyboard byte, which flags the various modifier keys. KBDFLG is stored in the corresponding byte in the second queue ($1580..15FF). The console driver maintains a count of the current number of characters in the queue and keeps index pointers to the current front and rear of the queue.

When a key is pressed, KBD appears at $C000 just as it does in Apple II. At this time KBDFLG also becomes available at $C008. The keyboard interrupt is cleared with the keyboard strobe, $C010, just as it is in Apple II.

KBD and KBDFLG are picked up by the keyboard (SIR#02) interrupt handler, which is embedded in the console driver. If they represent one of the five console control keys, that function is executed immediately. Otherwise, if a standard key was pressed, KBD would be used as an index into the keyboard layout look-up table (page $17). KBDFLG and the modified value of KBD are then stored in the typeahead buffer. The console driver will retrieve them when it feels so inclined.

Before exiting, the keyboard interrupt handler also checks to see if the "any-key" event is armed or if this is the "attention" event character. If so, the handler queues up the appropriate "event." Later, before returning to the user program, SOS checks the event queue and transfers control to the event handler as a subroutine.

SOS Data and Jump Tables: $1900..19FF. The first few bytes on page $19 contain important status information (table 4). During ordinary business, some (or all) of these bytes control the video generator and/or similar accessory apparatus. But when the monitor is running, they have no perceptible effect. So there must be more than one way to control the video generator.

1900:	10	??
1901:	06	Highest user memory bank.
1902:	00	Console control #7 and #9. Setting bit 7 suspends screen output; bit 6 will "flush" screen output. Low nibble: ??
1903:	00	High bit set indicates NMI pending.
1904:	8F	??
1905:	19	??
1906:	82	Console control #5. Clearing bit 7 turns off video. Bit 6 may be involved in graphics. Low nibble contains text mode [0..2].
1907:	00	Number of pages allocated for graphics.

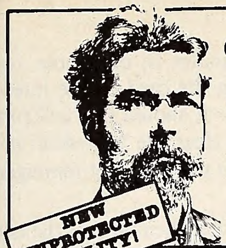
Table 4. SOS status info. Some bytes control video generator.

Page $19 also contains a jump table beginning at $1910. The jumps take the form "1913: 4C CA E2 JMP E2CA". The table provides fixed entry addresses for certain subroutines that apparently will be supported in future versions of SOS. The list is in table 5. Those marked with an asterisk are documented by Apple and are legal to use. The others . . . well, they do appear in the jump table.

Access	SOS address	* = Legal to use
1910	198F	Probably debug.
1913	E2CA	* AllocSIR: Allocate internal resource.
1916	E352	* DealcSIR: Deallocate internal resource.
1919	E3C2	Disable reset key (unless NMI pending).
191C	E3F3	Enable reset key (just sets FFDF bit 4).
191F	E41D	* Queevent: Queue an event.
1922	E3A9	* SelC800: Grab $C800 expansion space.
1925	EE2A	Writes "system failure," the value of A, and hangs.
1928	EE17	* SysErr: reports errors from drivers to caller.
192B	F5C5	? error number look-up for internal buffer allocation.
192E	F686	? error number look-up for internal buffer allocation.
1931	F710	? error number look-up for internal buffer allocation.
1934	19D3	Probably debug (AND #20, STA 19D2, RTS)
1985	1910	Probably debug.

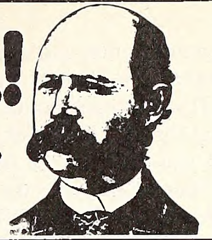
Table 5. Supported SOS subroutines.

Page $19 also contains a copyright notice at $1990, a few other data bytes of mysterious function, and a lot of zeros. The subroutine SysErr stores the error number at $1980 and the return address at $19FD and $19FE. The "system-failure" routine uses the end of this page to store the



®APPLE REVERSE-GEAR!

Search through Listings and Catalogs in Both Directions!



NEW UNPROTECTED UTILITY!

Double-Take

2-Way-Scroll/Multiple-Utility Disk

by Mark Simonsen

A hundred times a day, you type **L-I-S-T** and your Applesoft Listings dutifully appear on your monitor... then promptly scroll off of the screen into Hyper-Space. If the program line you are looking for goes by, you must LIST AGAIN to read it. There's a better way...

2-WAY SCROLLING: Now you can list your programs (all or part) with the added ability to CHANGE LIST-DIRECTION using the Apple Arrow Keys. The monitor becomes a "Search Window" to be moved UP AND DOWN through a listing at will.

IMPROVED LIST-FORMAT:* (optional) Each program statement is listed on a new line for easy tracing of program flow, and FAST de-bugging. Commands are properly-spaced (one space between words, not two) and much easier to follow.

*Similar to XLISTER on Utility City, BUT operates in both directions at Machine-Language speed, directly from the LIST command. For-Next Loops and "If-Then's" are not called out, as they are in XLISTER.

High-speed PRINTER LISTINGS in Improved Format are a snap too, in any column-width.

MONITOR-LISTINGS feature 2-Way-Scroll too. Normal Disassemblies AND Hex Dumps are scanned quickly in BOTH DIRECTIONS. Not to mention informative 2-Way Hex/Ascii dumps—

MACHINE-CODE Scrolls Both Ways

```
6ØØØ-  53 41 4D 5Ø 4C 45 2Ø 54   SAMPLE T
6ØØ8-  45 58 54 2Ø 46 49 4C 45   EXT FILE
6Ø1Ø-  2Ø 4C 49 53 54 45 44 2Ø    LISTED
6Ø18-  57 49 54 48 2Ø 44 4F 55   WITH DOU
6Ø2Ø-  42 4C 45 2D 54 41 4B 45   BLE-TAKE
6Ø28-  27 53 2Ø 48 45 58 2F 41   'S HEX/A
6Ø3Ø-  53 43 49 49 2Ø 44 55 4D   SCII DUM
6Ø4Ø-  5Ø 2Ø 46 45 41 54 55 52   P FEATUR
6Ø5Ø-  45 2E 2E 2E 2E 2E 2E 2E   E.......
```

Bonus Utilities: Any or all of the following enhancements may be hidden in memory, "unseen" until accessed—

CROSS REFERENCE: Displays existing variables, strings & line nos. on which each occurs—

```
A$:  1ØØ  2ØØ  25Ø  3ØØ
X:   1Ø   2Ø   3ØØØ  3Ø1Ø  3Ø2Ø
Y:   5Ø   3ØØØ  4ØØØ  52ØØ
```

VARIABLE DISPLAY: Displays all program variables & strings with each one's current value—

```
A$  =  "NOW IS THE TIME"
X   =  255
Y   =  3.14159
```

Better RENUMBER and APPEND: Append program lines ANYWHERE into other programs (not just at the end) without renumbering.

PLUS: Free-Space-On-Disk, Enter Machine Language from Basic, Fast Program-Stats, Built-in Hex-Dec Converter, Ctrl-Char.Display, Cursor-Omit or Replace, Applesoft Auto-Line-Numbering . . .

Double-Take $34.95

UNPROTECTED (copyable) and compatible with **Apple II and IIe**, DOS 3.3®, Beagle Bros' PRONTO-DOS™ and Synergistic's G.P.L.E.®

Normal Applesoft Listing:

2-WAY VIDEO SCROLLING

(Hi-Res Picture Mover)

```
90   HOME : HGR2 : POKE 768,160: POKE
      769,Ø: POKE 770,76: POKE 771
      ,44: POKE 772,254
100   FOR PG = 2 TO 3: POKE 23Ø,PG
      * 32: CALL 62454: HCOLOR= 3
      + (PG = 3): FOR X = Ø TO 15
      9 STEP 5: HPLOT Ø,X TO X * 1
      .8,159: NEXT X,PG: HGR
110  ST = 16384: GOSUB 2010:ST = 2
      4576: GOSUB 2010: GOTO 110
2000   POKE L,N -  INT (N / 256) *
      256: POKE L + 1, INT (N / 25
      6): RETURN
2010 N = ST:L = 60: GOSUB 2000:N =
      ST + 8191:L = 62: GOSUB 2000
      :N = 8192:L = 66: GOSUB 2000
      : CALL 768
2030   POKE 50,63 + 192 * (ST = 16
      384): VTAB 21: PRINT  SPC( 1
      20): VTAB 22: PRINT " P=PICT
      URE #";: PRINT 1 + (ST = 163
      84)":";: GET A$: PRINT A$: IF
      A$ = "P" THEN  RETURN
```

Optional Improved Format:

2-WAY VIDEO SCROLLING

```
90     HOME
     : HGR2
     : POKE 768,160
     : POKE 769,Ø
     : POKE 770,76
     : POKE 771,44
     : POKE 772,254
100    FOR PG = 2 TO 3
     : POKE 230,PG * 32
     : CALL 62454
     : HCOLOR= 3 + (PG = 3)
     : FOR X = Ø TO 159 STEP 5
     : HPLOT Ø,X TO X * 1.8,159
     : NEXT X,PG
     : HGR
110    ST = 16384
     : GOSUB 2010
     : ST = 24576
     : GOSUB 2010
     : GOTO 110
2000   POKE L,N - INT (N / 256) * 256
     : POKE L + 1, INT (N / 256)
     : RETURN
2010   N = ST
     : L = 60
     : GOSUB 2000
     : N = ST + 8191
     : L = 62
     : GOSUB 2000
     : N = 8192
     : L = 66
     : GOSUB 2000
     : CALL 768
2030   POKE 50,63 + 192 * (ST = 16384)
     : VTAB 21
     : PRINT SPC( 120)
     : VTAB 22
     : PRINT " P=PICTURE #";
     : PRINT 1 + (ST = 16384)":";
     : GET A$
     : PRINT A$
     : IF A$ = "P" THEN RETURN
```

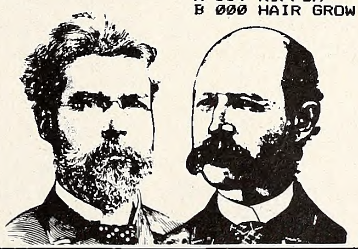
Each statement lists on a new line. Compare to the normal listing above.

Listings scroll Up AND Down at high speed, for easy location of program code.

LONG CATALOGS feature fast 2-Way-Scroll too, speeding up location of file names. Hit the Left-Arrow Key to SCROLL BACKWARD, any-key to continue forward, or ctrl-C for a clean break.

2-WAY VIDEO SCROLLING

```
]CATALOG
DISK VOLUME 123
*A 009 HELLO
 A 010 MULTI-CAT
 A 052 SORTFILE
*A 010 KEY-CAT
*A 052 BYTE ZAP
 A 024 HI-WRITER
 B 034 BBROS LOGO
 A 050 WOWZO
 T 002 XLISTER
 A 062 DOS BOSS
 T 010 LETTER
 T 020 STATEMENT
 T 003 NAIL FILE
*A 052 ALPHA PLOT
*A 008 TEXTTRAIN
*A 009 BEAGLE MENU
*A 021 HANG PERSON
 B 034 GOOD PIC
 B 034 UGLY PIC
 A 020 BUZZWORD
*B 020 FID
 A 020 TEXT DUMP
 I 010 CITY DUMP
 A 009 NIFFUM
 B 000 HAIR GROW
```



Beagle Bros™ Micro Software Inc.

4315 Sierra Vista/San Diego, Ca 92103
Telephone 619-296-6400

GOTO Your Software Store.

If they don't have the Beagle Bros disks you want, tell them to GET ON THE STICK by phoning Beagle Bros, 619-296-6400, or any Apple Software DISTRIBUTOR.

BEAGLE BROS DISKS:

All of our disks are unprotected (copyable), and come with a free Apple Peeks & Pokes Chart.

☐ **ALPHA PLOT** $39.50
Keyboard-draw hi-res pictures & charts for your programs. 2-Page drawing, relocate any section, compress hi-res images to 1/3 Disk Space.

☐ **APPLE MECHANIC** $29.50
Create custom type and Shapes for Animation. Six proportionally-spaced fonts (see "Typefaces" below). LIST-able hi-res demos. BYTE-ZAP inspects & re-writes any sector for making "illegal" changes.

☐ **BEAGLE BAG** $29.50
12-Games-Plus. The best Apple game bargain around. See review: January 1983 Softalk, pg. 148.

☐ **DOS BOSS** $24.00
Reword DOS—"Catalog" to "Cat", "Syntax Error" to "Oops!" etc. Protect programs—unauthorized Save-attempt can produce a "Not Copyable" message.

☐ **DOUBLE-TAKE** $34.95
2-Way-Scroll Multiple-Utility (read this ad).

☐ **FLEX TEXT** $29.50
70-columns/no-hardware! Basic commands produce expanded, normal & condensed hi-res type. Use Flex fonts, DOS Tool Kit®, or create your own.

☐ **FRAME-UP** $29.50
Create key-controlled or unattended high-speed shows of your hi-res, lo-res and text images.

☐ **PRONTO-DOS** $29.50
Bload hi-res & load programs at TRIPLE SPEED. Supports all commands including INIT.

☐ **TIP DISK#1** $20.00
100 Beagle Tip Book programs on disk. With TWO CHARTS— Apple Commands and Peeks/Pokes.

☐ **TYPEFACES** $20.00
26 additional type fonts for Apple Mechanic.

☐ **UTILITY CITY** $29.50
21 listable utilities— List-Formatter, Multi-Column-Catalogs, Trick Filenames... NUMBER ONE on Softalk Hobby-10 Best-Seller List, Feb. 1983.

AT YOUR APPLE DEALER NOW!
Or order directly from Beagle Bros—

VISA

master charge THE INTERBANK CARD

Visa/MasterCard/COD, call **TOLL-FREE**
Nationwide: 1-800-854-2003 ext. 827
California: 1-800-522-1500 ext. 827
Alaska/Hawaii: 1-800-854-2622 ext. 827

OR mail U.S.Check, Money-Order or Visa/MC#
to **BEAGLE BROS**, 16th-Floor
4315 SIERRA VISTA, SAN DIEGO, CA 92103

Add $1.50 First Class Shipping, any-size order.
Overseas add $4.00. COD add $3.00. California add 6%.
ALL ORDERS SHIPPED IMMEDIATELY

program counter and all register values for use in debugging. Other SOS routines store various temporaries on this page.

The copyright notice at $1990 is a good spot if you want to store things that can be found from the Monitor. If you want to store a lot of stuff you can also use the character set area.

Those Registers: the Onboard 6522 VIAs. The two VIAs are referred to as D-VIA ($FFD0..FFDF) and E-VIA ($FFE0..FFEF) respectively. They are fully occupied with Apple III hardware manipulations. You cannot, for example, use the VIA timers for your own purposes. The VIAs manage bank-switching, zero-page selection, and much of the other machinery that permits Apple III to accommodate the 64K address space of the 6502 cpu chip.

$FFEF: Bank register (E-VIA IORA). The low nibble selects the currently switched-in bank. The high nibble is generally $F. Attempts to change the high nibble have no effect. Those four bits are flags for interrupt requests from the slots.

$FFDF: Environment register (D-VIA IORA). Table 6 lists the significance of its bits. Apple would be happier if you confined your attention to bit 7 and didn't mess with the others. Table 7 contains a variety of information about the various standard environments.

$FFD0: Zero-page register. Selects the current zero page, which can be assigned to any page in memory. If alternate stack is enabled (bit 2 of $FFDF is clear) then all stack-using opcodes use the current zero page XOR $01. Extended addressing, on the other hand, functions only for zero pages in the range $18..1F. The user zero page is $1A. That's you and/or the interpreter. Drivers and SOS.Kernel use $18. Interrupt handlers use $00. SOS is supposed to decide these things; you are not. The SOS call handling routine even checks to see if the caller's zero page is $1A. If not, it crashes the system. Somewhere in darkest Cupertino, Apple maintains a coven of witch doctors who will cheerfully do unspeakable things to your image, should you violate this trust.

Value	Bit	Function	Bit = 0	Bit = 1
01	0	F000..FFFF	RAM	ROM
02	1	ROM#	ROM#2	ROM#1
04	2	stack	alternate	normal (true 0100)
08	3	C000..FFFF	read/write	read only
10	4	reset key	disabled	enabled
20	5	video	disabled	enabled
40	6	C000..CFFF	RAM	I/O
80	7	clock speed	2MHz	1 MHz

Note: ROM#2 doesn't exist.

Table 6. Environment register ($FFDF).

(Data mostly from unpublished *SOS Reference Manual*)

	User	Kernel	Driver	IRQ	Monitor
Environment register	$38	$34	$74	$74	$77
Clock speed	2 MHz	2 MHz	2 MHz	2 MHz	2 MHz
I/O space	disabled	disabled	enabled	enabled	enabled
Screen	on	unchanged	unchanged	unchanged	on
Reset key	unlocked	unchanged	unchanged	unchanged	unlocked
Write protect	read only	r/w	r/w	r/w	r/w
Stack	alternate	normal	normal	normal	normal
ROM	disabled	disabled	disabled	disabled	enabled
Zero page	$1A	$18	$18	$00	$03
Xbyte page	$16	$14	$14	none	none
Bank register	unchanged	unchanged	unchanged	handler's	$F0
6502 interrupts	enabled	enabled	enabled	disabled	??
Functions allowed:					
Issue SOS call	yes	no	no	no	no
Be interrupted	yes	yes	with care	with care	n/a
Handle interrupt	no	no	no	yes	n/a
Queue event	yes	no	yes	yes	n/a
Handle event	yes	no	no	no	n/a
Allocate SIR	yes	yes	yes	yes	??
Call SelC800	see text	yes	yes	yes	n/a
Call SysErr	no	yes	yes	no	n/a

Note: Upon entry to an interrupt handler X points to a $20 byte scratchpad area on zero page. These bytes should be addressed $00,X and so on. If the interrupt source is the onboard ACIA then Y contains the ACIA status register.

Table 7. The standard environments.

$FFDD: "Any-slot" interrupt flag. When a peripheral card in one of the slots pulls down the interrupt line, the interrupt handler is entered with 6502 interrupts disabled. The interrupt handler is, of course, responsible for clearing the interrupt condition on the card. If the interrupt handler wishes to enable 6502 interrupts (as it should if it will run longer than 500 microseconds) then it must also clear the "any-slot" interrupt flag by storing $02 in $FFDD. Otherwise the interrupt manager will do it for you when the handler exits.

I/O Space: $C000..CFFF. I/O space is on-line when bit 6 of the environment register ($FFDF) is set. It is actually $C000..C4FF and $C800..CFFF. The intervening bytes $C500..C7FF are always RAM. Table 8 lists those registers of which we have some clue. There are many mysterious others. When in doubt, there is a good chance that a register's function is similar or identical to its role in Apple II.

```
C000:  KBD. ASCII value of the most recent keypress.
C008:  KBDFLG. Bits are flags for modifier keys.
C010:  Clear keyboard strobe.
C020:  Deselect all peripheral slots (CFFF more commonly used).
C030:  Clicks speaker (Apple II type).
C040:  Beeps speaker (Apple III type).

  Soft switches
C050:  Black and white on.
C051:  Color on.
C052:  Forty-column mode, low-res mode.
C053:  Eighty-column mode, hi-res mode.
C054:  Display buffer 1.
C055:  Display buffer 2.
C056:  Text on.
C057:  Graphics on.

  Peripheral card I/O (each slot has $10 bytes)
C090:  Slot 1.
C0A0:  Slot 2.     Normally addressed as C080,X
C0B0:  Slot 3.        where X = s0 (slot number in high nibble).
C0C0:  Slot 4.     Note: there is no slot 0.

  Onboard 6551 ACIA
C0F0:  ACIADR      Data register.
C0F1:  ACIASR      Status register.
C0F2:  ACIAMR      Command mode register.
C0F3:  ACIACR      Control register.

  Peripheral card PROM space (one page for each slot)
C100:  Slot 1.
C200:  Slot 2.
C300:  Slot 3.
C400:  Slot 4.
```

Table 8. I/O space: $C000..CFFF.

Notice that the $10 bytes beginning $C080, $C0D0, $C0E0, and $C0F0 are not used for slots in the III. In Apple II they would be slots 0, 5, 6, and 7 respectively. There may or may not be a clue to their function in the assignment of various connecting plugs to imaginary slots in emulation mode. For example, $C0F0+ is the ACIA (asynchronous communication interface adapter), as indicated in table 9. The ACIA runs the serial port, which in emulation mode is assigned to an ethereal slot 7. Similarly, emulation mode assigns the floppies to slot 6, and the floppy drivers (buried in SOS.Kernel at $E899) probably access bytes in $C0E0..C0EF, and in $C0D0..C0DF as well. But this may only be speculation.

```
                   Entry: A = slot number ($00 deselects all slots)
                   Entry point: (via JMP table) at $1922
E3A9:  C9 05      * CMP  #05          ; range check
E3AB:  B0 14      * BCS  ->E3C1       ; error returns carry set
E3AD:  08         * PHP
E3AE:  78         * SEI               ; disable 6502 interrupts
E3AF:  8D C0 DF   * STA  DFC0         ; save slot number
E3B2:  09 C0      * ORA  #C0
E3B4:  8D BF E3   * STA  E3BF         ; build instruction at E3BD
E3B7:  2C 20 C0   * BIT  C020         ; deselect strobe
E3BA:  2C FF CF   * BIT  CFFF         ;    same
E3BD:  2C FF C0   * BIT  C0FF    <--- ; becomes CsFF
E3C0:  28         * PLP               ; restore 6502 interrupts
E3C1:  60         * RTS
```

Table 9. SelC800 disassembler listing.

GREAT PLAINS ACCOUNTING □ MANAGEMENT SOFTWARE

Our books simplify yours!

Great Plains Software shows you how to appraise your business from every angle. Our books handle your accounting in expert fashion. They analyze your management policies and evaluate the profitability of your decisions. And our books give you the reports you need – so you can see where your business is making profits – and losing them, whether it's by salesman, product, territory or profit center.

Our new version of the Hardisk Accounting Series includes General Ledger, Accounts Receivable, Accounts Payable and two new modules – Payroll and Inventory with Point of Sale. Our programs are fast, powerful and efficient. Our books make it easy to operate and understand.

You can find Great Plains Software at your local computer store – or call us for more details. We'll be glad to show you how the Hardisk Accounting Series can simplify your books.

SPECIFICATIONS

OVERALL
Password Privacy System
Written in UCSD Pascal**
Hard Disk Oriented
Operates on Apple III, IBM PC

GENERAL LEDGER
All entries on line entire fiscal year
Flexibly formatted financial statements
Comparative income statements and balance sheets

ACCOUNTS RECEIVABLE
Up to 32,767 customers*
Profit by customer, customer type, salesman and state
Open item or balance forward
Automatically posts to G/L

ACCOUNTS PAYABLE
Up to 32,767 vendors*
Accommodates manual or generated checks
Automatically posts to G/L

PAYROLL
Up to 32,767 employees*
Up to 20 deductions per employee
Withholding computed
Prints W2, 941 and checks

INVENTORY
FIFO, LIFO, standard cost, weighted moving average and serial number valuation
5 price levels per part
Concise report including profit by part and line
Point of Sale for cash and credit sales
Part numbers up to 15 characters
Automatically posts to A/P and A/R

**depending upon disk storage space*
*** TM UC Regents*

GREAT PLAINS™ SOFTWARE
123 15th Street North, Fargo, ND 58102-4292 • (701) 293-8483

$C800..CFFF is a 2K peripheral card expansion space used "in common" by all the slots. As in Apple II it can be selected by referencing one of the peripheral card I/O locations assigned to that slot. $CFFF does deselect all slots, but $C020 (formerly the cassette output toggle) is the preferred Apple III deselection strobe.

There are some new rules for using $C800 space that are intended to mesh with Apple III's interrupt-driven operating system. You are supposed to allocate the space prior to use by calling the SOS subroutine SelC800. The slot number is passed in the [A] register on a JSR to the entry point at $1922. (See the subroutine listing in table 9.) A value of $00 deselects all slots. Note that SelC800 saves the slot number in $DFC0; this allows the interrupt manager to restore the proper card allocation should an interrupt occur. The interrupt manager routinely deselects all slots on entry and reselects the proper slot on the way out.

The documentation states that SelC800 may be called from any environment including interpreters (except an NMI handler). This turns out not to be entirely true. The subroutine builds an instruction on-the-fly by storing the slot number ORA #C0 as high byte of the operand in bit $C0FF. The bit instruction then physically enables $C800 space for that slot. But this area of SOS is write-protected while running in the user environment, so the STA instruction doesn't work and the subroutine fails without notifying you. If you want to call SelC800 from the user environment you must enable write by clearing bit 3 of the environment register ($FFDF).

There must be another soft switch somewhere. When you enter the Monitor (with control-open-apple-reset), it comes up in forty-column mode. You can change to eighty-column mode with escape-8, and back again with escape-4. From eighty-column mode you might suppose you could also change back to forty columns by fiddling with the soft switches, perhaps by reading $C052 and maybe $C054. Things change, and you can tell it's really trying hard. But no combination quite makes it. We don't know why.

System Internal Resources: SIRs. When an interrupt occurs, the interrupt manager must know which interrupt handler goes with which interrupting device and where the handler address is located in memory. The SIR allocation scheme provides a look-up table. It also establishes "ownership" of a resource in order to prevent squabbles. Resources should therefore be allocated whether there will be interrupts or not. Somewhere in your code, place the following data table:

```
SIRADDR    .equ   SIRTABLE
SIRTABLE   .byte  00        ; SIR#
           .byte  00        ; ID code (will be assigned by SOS)
           .word  handler   ; interrupt handler address (or
                              $0000)
           .byte  bank      ; interrupt handler bank
```

Allocation is performed by JSR AllocSIR ($1913) and deallocation by JSR DealcSIR ($1916). The 6502 registers must contain: X = SIRADDR; Y = SIRADDR+1; A = total number of bytes in SIRTABLE. This will be $05, or some multiple of $05 in the event that several resources are allocated at the same time. AllocSIR returns with carry clear if the resource is successfully allocated.

Table 10 lists the numbers assigned to various resources. Examination of AllocSIR suggests that the range is $00..17. There are a lot of question marks. One wonders about the digital/analog audio converter, the paddle ports, and other mysteries, such as whether the interrupt line of the MM58167A clock chip is wired up.

SIR#	Resource
00	(?)
01	ACIA
02	keyboard
03	(?) clock chip
04	(?)
05	used by console screen code 22. "SYNC"
06	character set downloader interrupts
07..0F	(?)
10	(?) character set downloader
11	slot 1
12	slot 2
13	slot 3
14	slot 4
15..17	(?) pseudo slots 5-7

Table 10. Internal system resource numbers (SIRs).

Boot Sequence: On power-up, or after control-reset, the boot process begins in ROM#1 (ROM#2 doesn't yet exist). Low-level diagnostics are performed. Then block 0 is read from the disk in the built-in drive. This is the SOS boot code and is present on every disk that has been formatted by the *System Utilities* program. It must be present for a successful boot. It consists of one block of "absolute" code and is loaded into the computer at $A000, where it begins to run.

The boot code begins by locating and switching in the highest bank of RAM. Then it goes back to the disk and loads in five more blocks (blocks 1..5). These are placed in $A200..ABFF. Block 1 currently contains all zeros; blocks 2..5 are the disk directory. The boot code then scans the directory and locates SOS.Kernel, which it loads into memory at $1E00..73FF.

When SOS.Kernel begins running, it promptly relocates bytes $3000..73FF into the area $BC00..FFFF. This is the functional SOS.Kernel. The loader portion is eventually overwritten and discarded. First, however, it locates and loads SOS.Interp and loads the drivers from SOS.Driver. It then initializes SOS.Kernel and each of the drivers. Finally, control is transferred to the first instruction in the interpreter and you are in business.

Data Disks: Your Own Boot Code. If you want to end up in Apple III native mode, the boot process had better find the SOS boot code in block 0 on the disk in the built-in drive. Any disks you ever expect to use as SOS boot disks must have that code. On the other hand, you may wish to create data disks that have the SOS directory structure but cannot be booted. Or you may want the disk to boot, but to end up with some entirely different operating system in the machine, such as an emulator, for example. For either of these alternatives you will want to put your own code in block 0 on your disk.

You start with a single block: the 512 bytes contained in block 0. It will be loaded and begin to run at $A000. You may then use the ROM

Don't Pick Just Any Card

Pick the APPLI-CARD™ and APPLI-DISC™ combination and be a Sure Winner!

If you have an Apple® computer, we can deal you a once-in-a-lifetime hand that just can't be beat.

Now you can plug our APPLI-CARD into your Apple and have total CP/M capability. Adding the APPLI-DISC you'll have the potential to speed overall program execution by greater than 10 times while increasing your data storage by more than 30%.

We give you a choice of either a 64K or 128K ram extender. Add this to the APPLI-CARD's existing 64K and you have a 128K or 192K APPLI-DISC. Eliminate the need for an expensive add-on disk drive and end up with a faster system with more storage and more memory for a lot less money!

We've dealt you an unbeatable combination. Now it's up to you to call and cash in on a great deal. Ask about our 8088 Card, another innovative product from PCPI.

4 MHz 64K APPLI-CARD w/APPLI-DISC Software	$295
4 MHz 64K APPLI-CARD w/ 64K Ram Extender / APPLI-DISC Software	$395
6 MHz 64K APPLI-CARD w/APPLI-DISC Software	$375
6 MHz 64K APPLI-CARD w/ 64K Ram Extender / APPLI-DISC Software	$495

PCPI

Personal Computer Products, Inc.

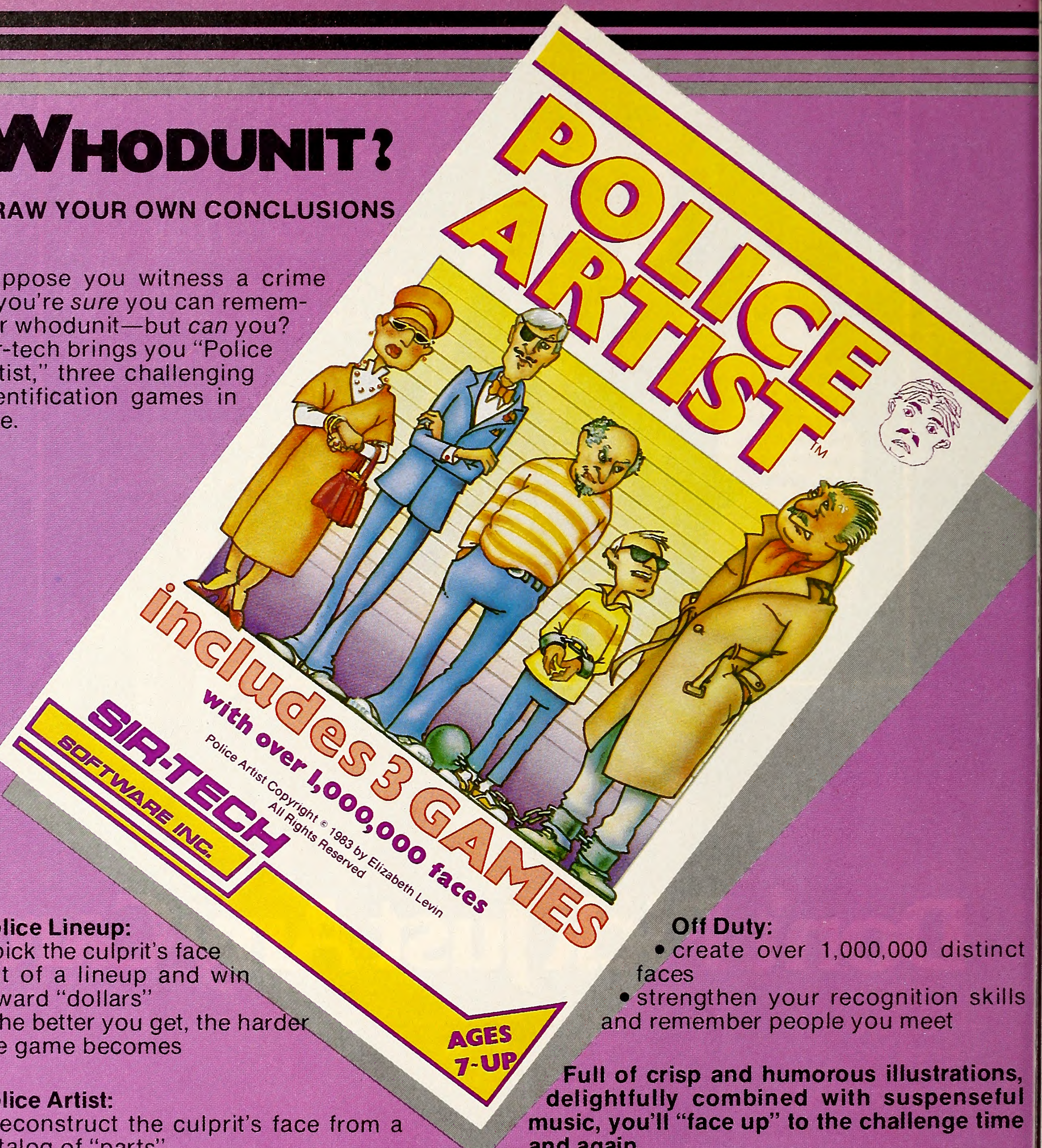
THE COPROCESSING CO.
16776 Bernardo Center Drive
San Diego, CA 92128
(619) 485-8411

Prices subject to change without notice.

WHODUNIT?

DRAW YOUR OWN CONCLUSIONS

Suppose you witness a crime—you're *sure* you can remember whodunit—but *can* you? Sir-tech brings you "Police Artist," three challenging identification games in one.



Police Lineup:
- pick the culprit's face out of a lineup and win reward "dollars"
- the better you get, the harder the game becomes

Police Artist:
- reconstruct the culprit's face from a catalog of "parts"
- peek at the culprit as much as you like, but the less you peek, the better your score
- develop the memory of a supersleuth

Off Duty:
- create over 1,000,000 distinct faces
- strengthen your recognition skills and remember people you meet

Full of crisp and humorous illustrations, delightfully combined with suspenseful music, you'll "face up" to the challenge time and again.
Three ways to be sure you'll "never forget a face," from the people who bring you "Wizardry" and its companion scenarios.

AVAILABLE FOR THE APPLE AT YOUR FAVORITE RETAILER.

SOFTWARE THAT COMPLIMENTS YOUR INTELLIGENCE

SIR-TECH SOFTWARE INC.
6 MAIN STREET
OGDENSBURG, N.Y. 13669
(315) 393-6633

Graphics created with the aid of Graphics Magician

Apple is a registered trademark of Apple Computer, Inc.
Police Artist is a trademark of Elizabeth Levin

subroutines to load in more blocks, so you can actually requisition as much space as you require. At the time your code begins to run, you will be in the Monitor or something very similar. The environment register reads $77, zero page is $03, and bank 0 is switched in. You have available all the hardware, including the VIA registers, extended addressing (with proper zero page), and all the internal resources. You do not, of course, have any of the SOS subroutines and facilities.

Your code should be assembled as "absolute" code by the Pascal 6502 assembler and should be ".org'd" on $A000. For data disks it should end in an infinite loop. *Word Juggler*, for example, creates data disks that, when you try to boot them, print "Can't boot *Word Juggler* data disk" in the middle of the screen and politely hang the computer.

Formatting Data Disks. After you've assembled your own boot code, it is a relatively simple matter to format data disks from application programs. It can be done entirely from Pascal and almost entirely from Basic. From Basic you will also need an invokable module that will write to a floppy disk by block number, just as Unitwrite will do in Pascal. The assembly source text for such a module is appended at the end of this article.

The floppy format driver (FMTDX) is activated by issuing a DControl call, code number 254 ($FE), to the appropriate driver (.FMTD1 for the built-in drive). In Pascal this is done with Unitstatus procedure. In Basic you use the Request.Inv invokable module that comes on the Apple III Basic boot disk. If you're working in assembly you just issue SOS call $83. When the DControl call is issued, the format process begins *immediately*. All error checking and confirmation requests must be done by your program before you issue the call.

The DControl call must specify a control list buffer. FMTDX.Driver expects a one-page (256-byte) buffer that will be reproduced on *each page* of the new disk. Normally this buffer should contain all zeros. The formatter places address code on each track and sector and fills the data fields with zeros, or whatever you put in your buffer. Then it quits.

You now have a formatted disk. It is not yet a SOS disk. It contains neither a directory nor the block 0 boot code. You must store those yourself from program buffers using Unitwrite (if you are working in Pascal). If you ever want to use the disk as a SOS boot disk, just copy block 0 from some other boot disk. Otherwise transfer your own code. Remember to chop off the header block, which the assembler will have placed in front of your code. Start the transfer at block 1 of the codefile.

Next you must install a directory. The minimum requirements for a usable SOS directory are listed in table 11. You must store the indicated byte values on the disk. Just put them in the proper place in the 512-byte buffer and write the whole block onto the disk.

```
The boot code—blocks 0..1 (bytes 0000..03FF on the disk)
   Your code, or the SOS code from another boot disk
The directory—blocks 2..5 (bytes 0400..0BFF)
   0400: 00 00 03 00
   0404: Fx—where x is the length of the desired volume name in the
         low nibble. The high nibble should contain F, for root directory
   0405: The volume name in ASCII capitals (do not prefix with "/")
   0414: 75
   0422: C3 27 0C 00 00 06 00 18 01
         (These last two words are 0600 = the block number of
                                          the bit map
                                    0118 = 280 dec. = blocks
                                          on volume )
   0600: 02 00 04 00
   0800: 03 00 05 00
   0A00: 04 00 00 00

The bit map—block 6 (bytes 0C00..0CFF)
   0C00: 01 FF FF FF FF...through byte 0C22
```

Table 11. Minimum requirements for a SOS disk.

Word Juggler manages to write all this information onto the disk by a short segment of elegant and compact code. The utilities program uses the brute-force approach. It simply includes fourteen pages of a standard SOS structure (mostly zeros) and transfers the whole thing to disk in one piece. No wonder the utilities program is 123 blocks long.

Unitread and Unitwrite for Basic: an Invokable Module. The

Device.IO.Inv invokable module contains two procedures. In Basic they are external procedures and require the perform statement (see page 162, *Apple Business Basic Manual*).

The procedures are

```
unitread (% devnum%, @ buf% (0), % length%, % block% )
unitwrite (% devnum%, @ buf% (0), % length%, % block% )
```

These procedures read from or write to a specified block number (block%) on the disk in a specified device number (devnum%). They transfer (length%) bytes to or from the buffer in memory. The buffer must contain enough bytes or Unitread will spill data over onto surrounding memory with disastrous results. Normally the buffer should be dimensioned as an integer array; for example, DIM buf%(512). This buffer will contain more than enough room for two blocks (1,024 bytes).

Unitwrite is a dangerous procedure. There is absolutely no protection from errors. It is easy to write all over a disk directory, destroying it and rendering the entire disk unusable.

The device numbers of the floppy disks are .D1 = 1, .D2 = 2, and so on.

After typing in the text, save it to any pathname, perhaps Devio.Text. Then assemble it to the corresponding codefile, Devio.Code. Finally, change the name to Device.IO.Inv. If the assembler is not allowed to append the suffix .code, the file-type designation will get all screwed up and it won't invoke.

```
        .macro  pop
        pla
        sta     %1
        pla
        sta     %1+1
        .endm
;
        .macro  push
        lda     %1+1
        pha
        lda     %1
        pha
        .endm
;
        .macro  SOS
        brk
        .byte   %1
        .if     "%2"<>""
        .word   %2
        .else
        .word   param0
        .endc
        .endm
;
DRead   .equ    80
DWrite  .equ    81
;
buffer  .equ    0E8
;
        .proc   unitread,4
        .def    return,devnum,param0,param1
        .def    param2,length,block
;
        jmp     start
;
return  .word   0000
devnum  .word   0000
;
param0  .byte   00          ; number of parameters
param1  .byte   00          ; device number
param2  .word   buffer      ; pointer to buffer
length  .word   0000        ; bytes to read/write
block   .word   0000        ; block number to begin
                              read/write
param8  .word   0000        ; bytes read - result
;
start   .equ    *
        pop     return
        pop     block       ; pop procedure parameters
        pop     length
        pop     buffer
        pop     devnum
        lda     #05         ; number of parameters for
                              DRead
        sta     param0
        lda     devnum      ; transfer one byte
        sta     param1
        SOS     DRead       ; issue DRead SOS call
        push    return
        rts
;
        .proc   unitwrite,4
        .ref    return,devnum,param0,param1
        .ref    param2,length,block
;
        pop     return
        pop     block       ; pop procedure parameters
        pop     length
        pop     buffer
        pop     devnum
        lda     #04         ; number of parameters for
                              DWrite
        sta     param0
        lda     devnum      ; transfer one byte
        sta     param1
        SOS     DWrite      ; issue DWrite SOS call
        push    return
        rts
;
        .end
```

XFR.Block is a short Basic program intended to illustrate use of these procedures. It transfers whole blocks (512 bytes each) between specified block numbers on (separate) floppy disk drives.

```
10   INVOKE "device.io.inv"
20   DIM buf% (512)
30   HOME: PRINT "Transfer disk blocks utility"
40   PRINT
50   INPUT "Source device number: "; source%
60   INPUT "Destination device number: "; dest%
70   INPUT "Number of blocks to transfer (0..2): "; blks
80   length% = CONV% (blks * 512)
90   INPUT "Block number to begin reading: "; readblk%
100  INPUT "Block number to begin writing: "; writeblk%
110  PRINT: PRINT "Press any key to begin transfer": GET g$
120  PERFORM unitread (% source%, @ buf% (0), % length%,
     % readblk%)
130  PERFORM unitwrite (% dest%, @ buf% (0), % length%,
     % writeblk% )
140  PRINT: PRINT "DONE"
```

The new Sensible Speller IV™ corrects spelling mistakes immediately ...and knows over 80,000 words

Sensible Speller,™ the first complete spelling verification program for the Apple computer, now offers immediate correction of spelling mistakes. Find out for yourself why three out of four Apple owners buying a spelling checker chose the Sensible Speller[1] and why Sensible Speller is the only spelling program to ever appear on Softalk's Top 30 list. It's so popular and well rated not only because it offers the authoritative 80,000-word Random House Dictionary on diskette, but also leaves plenty of room for a virtually unlimited number of your own specialized or technical words. And Sensible Speller is fast: a ten-page document can be checked in a minute or two. It's menu-driven to make it friendly and easy to use. And each misspelled word is shown to you in the middle of a small excerpt from the document to quickly remind you of the context.

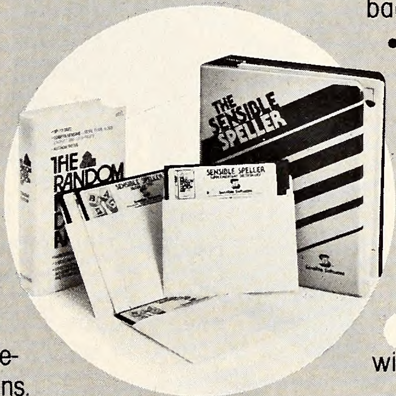
Get more features with Sensible Speller IV:

- Now you can immediately correct misspelled words by replacing them with the proper spelling.
- Now the Sensible Speller even suggests the correct spelling for your mistakes.
- Now there are more options for searching the dictionary, including multiple-character and single-character substitutions.

More dictionaries:
The Official Black's Legal Dictionary is now available for $99.95. Other specialty dictionaries, including Steadman's Medical Dictionary, will be available soon.

More word processor compatibility:
Each Sensible Speller package contains multiple versions for compatibility with almost all Apple word processors including (partial list): DOS 3.2, DOS 3.3 (Apple Writer, Magic Window, PIE, Screen Writer, etc.), SuperText, Word Handler, CP/M (Wordstar, etc.), and Pascal word processors. The Sensible Speller works on all Apple //e, II + and Apple-compatible computers with one or two disk drives.*

*New features not available in Apple CP/M, Pascal and Word Handler versions.



A special second chance $50-OFF trade-in offer:
If you or someone you know bought the wrong spelling verification program (Sensible Speller is the right one), that's still another mistake Sensible Speller can quickly correct. How? With this **$50-Off "Switch to Sensible" Offer:** Just send in your old "dictionary" program (manual and disks) with a check for only **$75** and we'll rush you a brand new Sensible Speller. **This offer expires July 31, 1983.** (Of course, current owners of the Sensible Speller can upgrade at any time for only $15).

Package includes:

- an easy to read instruction manual.
- the Sensible Speller program and one backup.
- a main dictionary diskette with the 43,000 most commonly used words (and room for 10,000 of yours).
- a supplementary dictionary diskette containing the remaining 37,000 words in the Concise Edition of the Random House Dictionary.
- a FREE Concise Edition hard cover Random House Dictionary, so you can check pronunciation and meaning with an authority, too!

Just $125.00 at your local dealer. If you order direct, please add $1.25 for shipping. COD, check, Visa or Mastercard welcome.

Sensible Software, Inc.
6619 Perham Drive
West Bloomfield, MI 48033
(313) 399-8877

[1]Softalk Magazine, August 1982

Yes, we know it's springtime and the sun is shining, and we'd all rather be outside playing, but we only meet once a month, so let's get through with this; then we can go out and get a tan.

The System Master and Sample Programs disks that come with the Apple have some useful and interesting things on them, but sooner or later you're going to have to use disks other than those two, unless you're the type of person who buys an expensive tape deck and listens to only the freebie stereo demonstration tape and no others.

Even if you plan on forever using prewritten programs and never writing a single program of your own, you'll probably need blank disks for storing data generated by programs.

One of the nice things about using the Apple is that it takes care of storing things for you. Thanks to the computer's memory and those wonderful things called disks, you don't have to worry about where stuff goes, how much room it takes up, or where to look for it.

The computer's memory is already set up for you. When you load programs from disk, the computer already knows where to put them. Disks, however, present a different problem. When Apple puts together its computers, it knows that whatever language or operating system you'll be using, it will be stored inside just fine. But disk manufacturers make disks for all different computers, and each computer is capable of using different operating systems.

So, to avoid discriminating against users of different systems (and to save themselves extra work), the disk manufacturers crank out disks in such a way that they can be used by anyone, regardless of the computer or operating system. It's the same reason the supermarket sells eggs in the shell. If they sold them already scrambled, people who like boiled eggs wouldn't be able to use them, bakers wouldn't be able to separate the whites from the yolks, and sunnyside up would be out of the question.

That's just fine for them, but what about us? Before you make an omelet, you have to scramble the eggs; before you use your new disk, you have to put it in a form that's understandable to DOS. This process is also called initializing the disk. The natural reaction to this bit of news is usually, "That's crazy. I never have to format my cassette tapes to record on them; why do I have to format a disk? Grrr!" Perfectly understandable, but it's not the same thing.

Another way to look at disk formatting is to think of two different reading systems. The first one is the way we're accustomed to reading—each line from left to right, a whole line at a time. The second reading system is for people who have a rare eye disorder that prevents them from moving their eyes back and forth. To save them from moving their heads from left to right all the time, the second reading system has them reading the first three words of each line, going from top to bottom. When they reach the bottom of the page, they go back to the top and read the next three words of each line, and so on.

It's easy to see that if you don't know who will be reading a book, you won't know how to write on the page; all sorts of confusion could arise with the good-eyes people reading the books written for the bad-eyes people. Operating systems work the same way. Disks intended for use with DOS, for instance, can't be read or written to by computers using CP/M, and vice versa. And when you consider all the other types of computers with all their operating systems, you can see why disk makers threw up their hands and said, "Fine. We'll make the things; you people format them however you want."

Now that we're convinced that disk formatting is a privilege and not a chore, let's break out a new disk. Most of the time, whether you're using a database, spreadsheet program, word processor, or adventure game, any information you store will have to go on a disk that's been initialized. That's because, when you're trying to save information, the computer looks at the way your disk is set up, and if there are no "lines" or boundaries on it defining how things should be written, the computer panics and doesn't know where it's safe to write.

Formatting a disk is just like setting up a chart. You don't have to determine where the lines are drawn, however. DOS 3.3 sets up the disk with thirty-five tracks (concentric circles like lanes on a track field) and sixteen sectors (like pie slices). DOS knows how to read things written in this form and none other. Any variations on this setup—thirty-four tracks or fifteen sectors, for instance—will just confuse and frustrate it.

Boot any disk that already has DOS on it. Either the System Master or Sample Programs disk will do. When the disk stops spinning, DOS will have been loaded, as well as the disk's greeting program (the program that loads first when the disk is booted, remember?). All you have to do now is copy the DOS that's now in memory to your new blank disk. In doing so, you also format the new disk with tracks and sectors just like those on the System Master.

The command to begin the whole process is *init*, which is short for "initialize." Just like load and save, init requires that you use it with a program name tacked onto it. You can do this right now, by typing *init Hello.*

At this point, the disk drive will grunt a few times, spin the disk around for about thirty seconds, and then stop. Here's what's going on. When the Apple sees you type *init Hello*, it understands you as saying, "I want you to do three things. First, look at the DOS that's in memory now, and divide the disk into tracks and sectors just like those on the disk that this DOS came from. Second, copy DOS from memory to the disk. Third, take the Basic program that's in memory, put it on the disk as the greeting program, and save it under the name Hello."

When the initializing has been completed, catalog the disk and you'll see that the greeting program is the only program on it. The rest of the disk is almost all empty.

It's important to note here that as time goes by you'll hear people refer to the greeting program as the *hello* program. Truth is, you can call the greeting program anything you want (Bonjour, Ace Face, Omelet), but Hello is the conventional name. Whatever name you give it, the Apple will look for that program whenever you boot that disk.

You can also have the greeting program do anything you want. While the one on the System Master happens to load Integer Basic into memory, you can write your own that will catalog the disk, load another program, or do nothing at all. The choice is yours.

If you're using the greeting program on the System Master as the greeting program on your new disk, you might be concerned when you boot it up and it says, "File not found." That's because the program tries to load Integer Basic from disk, but Integer Basic isn't there. You can get around this stumper every time you boot the disk by writing your own greeting program before initializing the disk.

We said that except for the greeting program the rest of the disk is "almost all empty." What that means is the disk has no other programs or files on it. DOS, however, takes up three tracks, and the disk's catalog takes up another. The rest of the disk is blank. Just to confuse things some more, "blank" is one of those words computer people use loosely. When someone gives you a disk and tells you it's blank, that means that either it's uninitialized or it's initialized with nothing stored on it. Use judgment.

Finally, we have ourselves a fresh disk. So what? If you're not using it to store word processing, database, adventure game, or data files from other programs, you can use it to store programs that you or someone else has on another disk. It's really easy. You just load the program from the original disk, put your disk in the drive, and save the program under any name you want. What could be simpler?

Now let's try it with binary files. Bload a file and then try to bsave it. There's that darned "?syntax error" message again! But wait, we know bsave is the command to save binary files; is the computer growing senile? No. The computer's fine; it's those snobbish binary programs that are causing such a fuss—they demand special treatment if you want them to behave.

Each time you load a Basic program, DOS puts it in memory so that its beginning is at a specific memory location. No matter how big or small the program is, it must always begin at the same place as other Basic programs. It's the same as when you go to the theater to see a play; the people who bought cheaper tickets (Basic program) begin filling in the balcony from front to rear (these are organized people).

For the next performance, even though there are only five people who bought cheap tickets (smaller Basic program), the first audience clears out entirely. Then the second audience begins filling in the balcony, also from front to rear. Basic programs are nice and organized.

Binary programs, however, are like the people who bought the expensive tickets and sit on the orchestra level. Not only are they separated from the section of memory where Basic programs sit, but they also get reserved seating. When binary programs fill the theater, they don't fill the rows from front to back because they have reserved places. Because of reserved seating, binary programs can stay in their seats even when other programs enter the theater. New arrivals have their own seats.

What all that means is you can have several binary programs in memory at the same time, whereas you can have only one Basic program in memory at a time. This is where bsave gets messy. With all those different binary programs in there, how does the computer know which one to save, and how does it know where one program ends and another begins? It doesn't.

When you use the bsave command, you have to tell the computer at which memory address to start reading, and where to stop. So, saving a binary program might look something like *bsave Binary Junk, A$4000, L$2000*. The specific numbers aren't important right now; it's the A$ and L$ that we're interested in here.

The A$ tells the Apple that the following number is the address in memory where it's supposed to start getting information, while the number following L$ tells it the length of the program, or how far it should go before it stops reading information. (We should note here that the address 4000 doesn't equal four thousand, nor is the length 2000 equal to two thousand. In the land of computerdom, numbers with dollar signs in front of them are hexadecimal numbers, another concept we'll get into some other time when the wind is right.)

When you type *bsave Binary Junk, A$4000, L$2000*, the Apple sees bsave and knows the program is in binary. Next, it goes to memory address $4000 and reads all the information in the next $2000 addresses. After it's done with that, it saves everything to disk under the name of Binary Junk.

It's not hard to see from all this talk about starting addresses and program lengths that transferring binary programs from one disk to another can be pretty messy. And if you think that's bad, try loading and saving text files. Whereas loading binary programs is easy and saving them is hard, transferring text files is next to impossible for us beginners.

Lucky for us, there's a program on the System Master disk called *Fid* that takes care of figuring out program addresses and lengths, and other hard stuff. And yes, it does transfer text files. *Fid* helps cut down the chance of errors in your typing by letting you perform DOS commands with single keystrokes. You can rename, delete, lock, unlock, verify, and copy files just by finding the command you want and typing in the corresponding number.

Fid also saves you a lot of time when working with files that have similar names. Suppose you have ten text files named Document.1, Document.2, and so on through Document.10. If you wanted to delete them all without *Fid*, you'd have to type

```
delete Document.1< return >
delete Document.2< return >
.
.
.
delete Document.10 < return >
```

Geez, that's a lot of typing. With *Fid*, when it asks you for the name of the file you wish to delete, you can just enter Doc=, which tells it to delete all files that begin with Doc. Likewise, =tion means "delete all files ending with tion," and =c= would mean "delete all files containing the letter c."

Before using *Fid* to copy things, you have to have an initialized disk to store things on. If you're too lazy to go through the initialization process yourself, there's another program on the System Master called *CopyA* (the Applesoft version; *Copy* is the Integer Basic version) that not only copies an entire disk for you, but also initializes your new disk with the greeting program of the original. It does all the stuff you already know how to do, but it eliminates all the init, load, and save commands you would normally have to type, and it does it faster.

Storing programs on disk can be a pretty risky process if you're not careful. True to computer form, the Apple doesn't do things the way we do. Let's say you write a program that generates square roots of numbers and you want to save it as *Roots*. When you save the program, the Apple writes the program's name in the catalog and then finds an empty spot in storage space to start writing the program. So far, so good.

Now let's say you write a program a few days later that traces your family's genealogy. Being an absent-minded human, you forgot about the previous program, and you save this one also as *Roots*. Catastrophy strikes. The Apple is so anxious to save time that when it looks in the disk's catalog and sees you already have a program called *Roots*, it thinks, "Great, *Roots* is already there, so I don't have to find space to put this; I'll just write it in the same place." The answer to the question you're about to ask is yes; the square-root program is gone.

What makes all this even more confusing is that even if the first *Roots* was twenty sectors long and the second *Roots* was just five sectors long, the catalog still says *Roots* takes twenty sectors. The reason for this is that when you write something on disk for the first time, the catalog measures it and knows how much space is allocated for it. The next time you save something under that name, it remembers how much space is reserved for the program, and tells the program to go to that location. If you want to store the genealogy program as *Roots* and have it show up on the catalog as taking only five sectors, you have to delete the original *Roots* first, then save the new *Roots*.

In this way, the catalog is sort of like a desk clerk at a hotel. When the Roots Club reserves a conference room for two hundred people, the desk clerk will direct anyone who comes to the desk looking for the Roots conference to that room. Suppose the Roots Club finishes its meeting and then leaves without telling the desk clerk. When Mr. Rex Roots comes to the desk and says, "I'm Roots . . ." the desk clerk will direct him to the conference room, even though he only wanted a single for the night.

Ideally, the clerk should have asked if he was with the conference or if he wanted a place to sleep. Ideally, the catalog should have asked whether or not you want the second program to go where the first one is. Ideally, you should have checked the catalog to make sure *Roots* wasn't taken already, thus avoiding the whole mess. Let's go back to the second ideal.

Remember those asterisks in front of file names in the catalog? Whenever you see one, that tells you the file is locked. Locked doesn't mean the file is inaccessible; it means you can't save anything under that name. You probably guessed that the DOS commands you use to enact this safety feature are *lock* and *unlock* plus the file name.

If you load a locked program, change it a little, and then try to save it under its original name, the Apple sees in the catalog that the file is locked, and responds, "File locked." If you've written a new program and try to save it under a name that already exists as a locked file, the "File locked" message lets you know that the name is spoken for. So, the locking feature is nothing more than a way to keep you from accidentally erasing valuable files.

The whole point of all this is that when in doubt, you should check the catalog before you begin writing to or changing stuff on your disk. It takes a few extra seconds, but it's worth it.

There's one more way to prevent yourself from accidentally writing over important information on your disk, and that's to make it impossible to write to the disk at all. You could keep the disk in its paper envelope and never remove it or you could cover up the area on the disk where the disk drive reads and writes, but that's neither practical nor realistic.

Fortunately, the disk drive gods put a handy device inside the drive called the write-protect switch. As you look at your disk, you see a small notch cut out. That little cutout is what lets the drive know whether it's okay or not to write to the disk. When you insert your disk into the drive, just before it's all the way in, you can hear and feel a small click as the disk locks into place. That's the write-protect switch popping into the notch on the disk.

When the switch is up, the drive knows it's legal to write stuff on the disk; likewise, when it's down, the gremlins who do all the writing go on coffee break and don't come back until they see the switch in the up position again. You can protect any disks you load information from and never need to write to by sticking on write-protect tabs that cover up the disk's notch, thus holding the write-protect switch in the down position while that disk is in the drive. Now you know what those little funny-looking stickers that came with your box of disks are for.

That's the basics of disk storage. Experiment with *Fid* and *CopyA* until you're confident with how each one works. If you're ever not sure whether or not something will work, the best way to find out is to try it and see what happens. But make sure you're not messing around with any valuable data. If you don't have any expendable disks, get someone else's and try it on those. Whether it works or not, you will have learned something. And if you completely destroy data files holding your checking account or yearly budget, you will have learned something valuable.

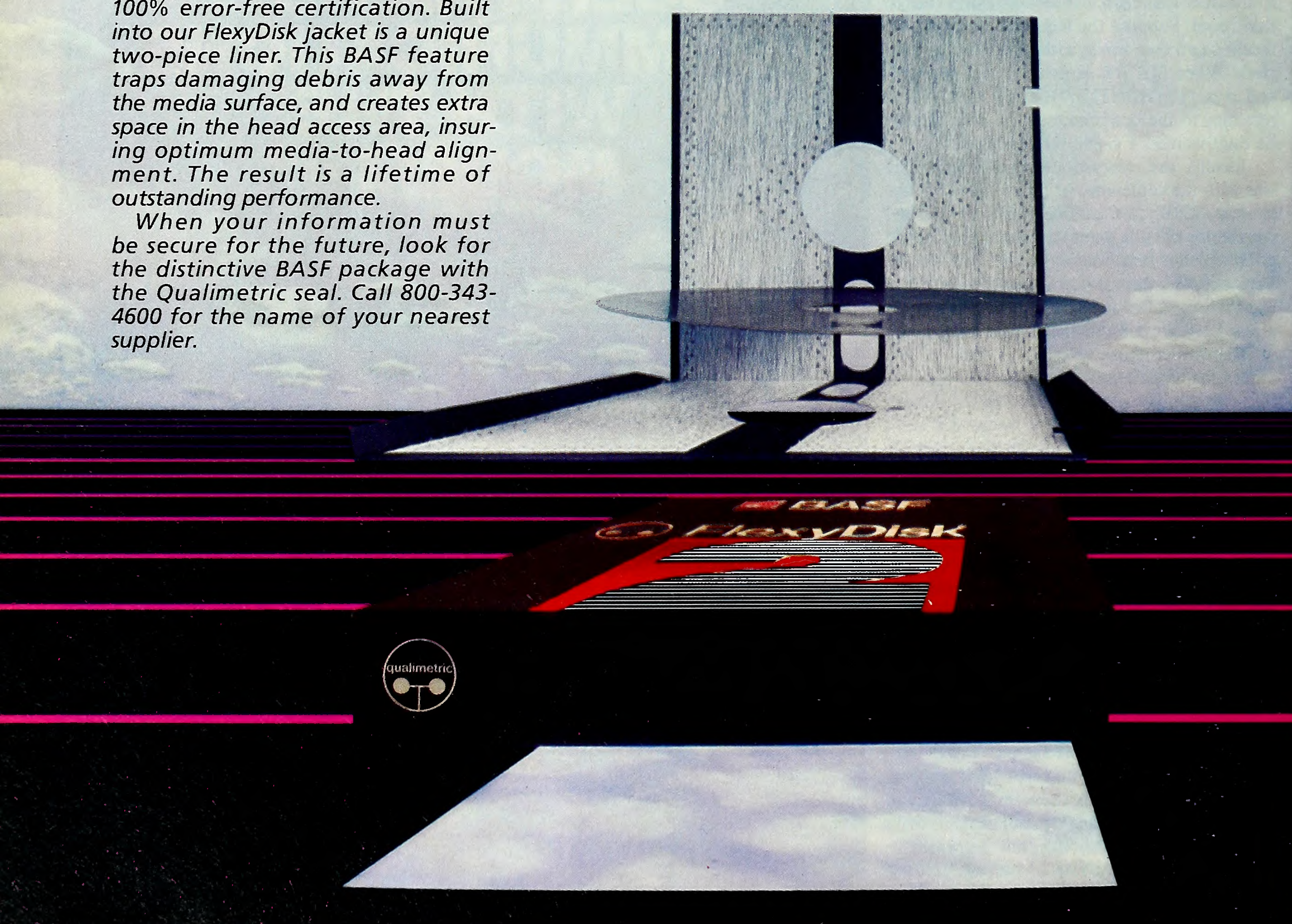
But learning is what we're here for.

BASF QUALIMETRIC™ FLEXYDISKS®
BUILT FOR ETERNITY - WARRANTED FOR A LIFETIME.

BASF Qualimetric FlexyDisks® offer you more...an extraordinary new lifetime warranty. The BASF Qualimetric standard is a dramatic new international standard of quality in magnetic media...insurance that your most vital information will be secure for tomorrow when you enter it on BASF FlexyDisks today.*

We can offer this warranty with complete confidence because the Qualimetric standard reflects a continuing BASF commitment to perfection...a process which begins with materials selection and inspection, and continues through coating, polishing, lubricating, testing, and 100% error-free certification. Built into our FlexyDisk jacket is a unique two-piece liner. This BASF feature traps damaging debris away from the media surface, and creates extra space in the head access area, insuring optimum media-to-head alignment. The result is a lifetime of outstanding performance.

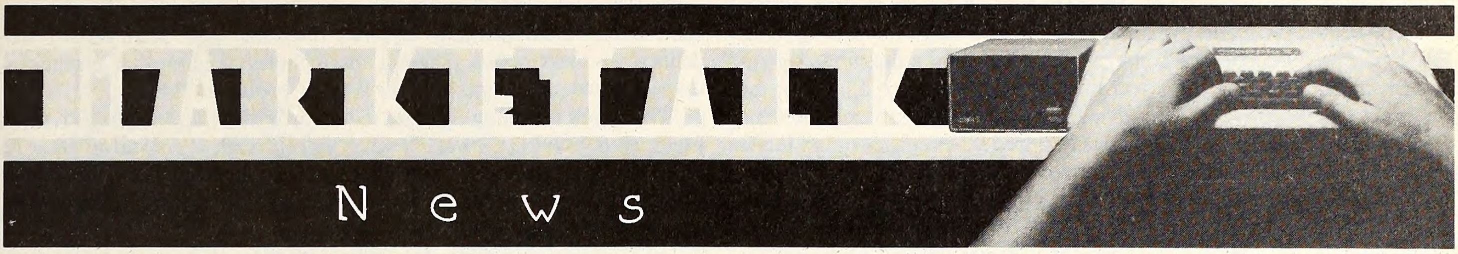
When your information must be secure for the future, look for the distinctive BASF package with the Qualimetric seal. Call 800-343-4600 for the name of your nearest supplier.



ENTER TOMORROW ON BASF TODAY

BASF

*Contact BASF for warranty details.



Unless otherwise noted, all products can be assumed to run on either Apple II, with 48K, ROM Applesoft, and one disk drive. The requirement for ROM Applesoft can be met by RAM Applesoft in a language card. Many Apple II programs will run on the Apple III in the emulator mode.

☐ **Kengore Corporation** (3001 Route 27, Franklin Park, NJ 08823; 201-297-2526) has announced the New Jersey-New York-Connecticut Microcomputer Show and Fleamarket, which will be held June 11–12 at the Meadowlands Hilton Hotel in the New Jersey Sports Complex. There will be more than seventy-five commercial exhibitors and two hundred fleamarket sellers displaying hardware, software, books, magazines, and accessories. Show hours are 10:00 a.m. to 5:00 p.m. Registration, $5.

☐ San Antonio is the site for the first annual Heart of Texas Computer Show. The show is scheduled for September 16–18 at the Convention Center in downtown San Antonio. The show's focus will be on microcomputer small-business systems, with a special emphasis on finance and inventory, agri-business, education, medical, and personal use. For information, contact Robin Mann at **Heart of Texas** (Box 12094, San Antonio, TX 78212; 512-226-4636).

☐ **SuperSoft** (1713 South Neil Street, Box 1628, Champaign, IL 61820; 217-359-2112) is entering the field of computer-aided instruction with the announcement of *Basic Tutor*, a self-instructional course that teaches you to program in Basic. Understandable to persons with no computer background, *Basic Tutor* contains graphic illustrations, frequent summaries, and a supplemental manual. $99.

☐ You can connect as many as four disk drives to your Apple with the Disk-Master board from **Computers Unlimited Enterprises** (5696 Highway 431 South, Brownsboro, AL 35741; 205-883-2933). The board supports single and double-sided eight and five-inch drives and DOS, Pascal, and CP/M operating systems. It also has internal ROM and RAM that can be used to store software drivers, keeping the main memory of the Apple free for programs. $285.

☐ *The Specialist* is a medical billing and accounts receivable package from **Digital Marketing** (2670 Cherry Lane, Walnut Creek, CA 94596; 415-938-2880) and is available for five specialty practices: anesthesiology, family practice, internal medicine, radiology, and surgery. Each version can be used for practices of one to nine physicians, and each is menu-driven, with three levels of password control. Billing and service dates may be kept on file for specialists who bill weeks after service, and more than twenty-five reports and mailing labels may be generated for billing and recalls. Runs under CP/M and requires either CBasic2 or CB-80. $995 to $1,495.

☐ **Southwestern Data Systems** (Box 582, Santee, CA 92071; 619-562-3221) is now including a subscription to the Source with every purchase of *ASCII Express the Professional*, *P-Term the Professional*, or *Z-Term the Professional.*

☐ You can plug up to four cards into Switch-a-Slot from **Southern California Research Group** (Box 2231-S, Goleta, CA 93118; 805-685-1931). Choose the card you want by using a switch on the front of Switch-a-Slot; only the selected card draws power. $179.50.

☐ If you make a lot of phone calls, especially with discount services such as Sprint, MCI, and ITT, then the *Address/Dialer* from **Christopher Systems** (2775 Glendower Avenue, Los Angeles, CA 90027; 213-664-4880) may be a help. It's a phone utility program to be used with the Novation Apple-Cat II. The *Address/Dialer* gives quick access to all names, addresses, phone numbers, appointments, birthdays, and memos on business associates. It automatically dials anyone you select and redials busy numbers continuously. $79.

☐ The **Forth Interest Group** (Box 1105, San Carlos, CA 94070; 415-962-8653) has announced its 1983 schedule of events and publications. *Forth Dimensions* includes articles on music, graphics, voice synthesis, project management, Forth in the laboratory, the history of Forth, and more. A subscription is included in the group membership.

☐ The newest product from **Ampower Electronic Instrument Company** (26 Just Road, Fairfield, NJ 07006; 201-227-7720) is the Track Pen for Apple's Lisa and VisiCorp's *VisiOn* software. The pen's function is the same as Lisa's mouse, but offers more speed, precision, and dexterity. It fits in your hand and rolls across the desktop to choose from menu selections on the screen. $295.

☐ From **RCM Software** (815 Friendship Drive, New Concord, OH 43762) comes *Pascal/CPM/DOS File Transfer Utilities.* Now you can transfer files in any of the three operating systems to either or both of the other two systems. Files are reformatted according to the requirements of the host operating system. All six transfer programs provide a common interface for easy and consistent operation. If you want to transfer files into Pascal, you'll need 64K. $45.

☐ **Desktop Computer Software** (303 Potrero Street, 29/303, Santa Cruz, CA 95060; 408-458-9095) has released its Apple III version of *Graph 'n' Calc*, a menu-driven graphics and statistical analysis program. The program lets you quickly prepare a variety of stacked and side-by-side bar, line, pie, and combined bar and line charts. Any number of charts can be stored on disk and later displayed in any sequence. *Graph 'n' Calc* can extract a row or column of data from any *VisiCalc* or *Desktop Plan* data disk. Supports Apple's ProFile hard disk. Requires 256K. $249.

☐ The **Educational Computer Consortium of Ohio** (4777 Farnhurst Road, Cleveland, OH 44124; 216-291-5225) is now accepting proposals for presentation at its Third Annual Educational Computer Fair, which will be held October 21–22 at Cleveland State University. Classroom teachers and those with practical computer education experience are encouraged to submit proposals. ECCO is searching for proposals in all content areas and grade levels.

☐ *Quickscore*, a program for scoring and evaluating objective tests with an optical mark card reader, is now available for schools. Available from **Pica Foundation** (Box 35487, Charlotte, NC 28235; 803-656-3455).

☐ **Universal Data Research** (2457 Wehrle Drive, Buffalo, NY 14221; 716-631-3011) has released a line of modems and an I/O card that automatically selects the appropriate baud rate. The 300-baud modems provide half and full duplex operation in originate, answer, and auto-answer modes. The 1200-baud modems provide full duplex operation with switch-selectable local echo. 300 baud, $149 to $219; 1200 baud, $449 and $499; 212A, $549 and $599; I/O speed select board, $119.

☐ **Intercalc** (Box 254, Scarsdale, NY 10583; 914-472-0038) is the international spreadsheet group, and it's announced the availability of *Spreadsheet*, the group's monthly newsletter, in which members provide tips, applications, templates, and share reviews with each other. $52 per year; $27 for one-half year.

☐ **Word-Power** (Box 736, El Toro, CA 92630; 714-859-7145) has developed Magic Typer, an interface card that connects the Apple to a Royal or Adler 5010 electric typewriter, making the typewriter function as a letter-quality printer. The product comes with the interface card, a six-foot cable, and the typewriter adapter card. $199.95.

☐ *Grids* is a software enhancement tool from **Associated Technology** (Route 2, Box 448, Estill Springs, TN 37330; 205-837-4718) that simplifies the process of including graphic data in a word processor text file. The package includes more than fifty of the most commonly used graph paper formats with instructions on how to create hundreds more. $23.

☐ **Lifeboat Associates** (1651 Third Avenue, New York, NY 10028; 212-860-0300) has made available *HomeTax*, a software package that helps you prepare federal income tax returns. Using a question-and-answer format, *HomeTax* leads you through the process of filling out the required forms and schedules, giving positive feedback along the way. The package includes more than two hundred screen pages of tax tables and information contained in the program. Requires CP/M. $95.

☐ Good documentation is what **Technology Training Systems** (1078 Ravine Ridge Drive, Worthington, OH 43085; 614-431-2174) is about. The company has developed twenty-five research-based guidelines for improving user manuals, and has packed them into *25 Ways To Improve Your Software User Manuals.* $5.

☐ **Prentice-Hall** (Englewood Cliffs, NJ 07632; 201-592-2347) is helping users learn *WordStar* and its associated programs with its release of the 272-page *Illustrated CP/M WordStar Dictionary: with MailMerge and SpellStar Operations.* It can be a tutorial for the new user or a reference tool for the experienced operator. $14.95.

☐ **Edu-Ware Services** (Box 22222, 28035 Dorothy Drive, Agoura Hills, CA 91301; 213-706-0661) has introduced *Hands-On Basic Programming*, a combination of workbook and instructional software. With the help of a tracing function, you can watch the computer execute your program. Interactive commands and error-trapping tools let you examine programs, locate errors, and debug. $79.

☐ **Rainbow Computing** (19517 Business Center Drive, Northridge, CA 91324; 213-349-0300) has released *Bat-Stat*, a program that keeps track of cumulative batting statistics for each player on a baseball team. The program is menu-driven and designed for a team of up to twenty players. Statistics are given for each game and for the season, and they include at-bats, runs, hits, doubles, triples, home runs, batting averages, and more. The program features easy data entry, error handling, and game and season report printing. $49.95.

☐ The latest educational software product from **Davidson & Associates** (6069 Groveoak Place, Suite 12, Rancho Palos Verdes, CA 90274; 213-378-7826) is *Math Blaster*, an instructional tool and math arcade game. The game includes more than six hundred problems in addition, subtraction, multiplication, division, fractions, and decimals for students six to twelve years of age. It also includes an editor that lets the student, teacher, or parent enter new math problems. $49.95.

☐ **FlipTrack Learning Systems** (526 North Main Street, Box 711, Glen Ellyn, IL 60137; 312-790-1117) walks you through *WordStar* in *How To Use WordStar*, the latest in its line of cassette tutorials. Three two-hour cassettes teach listeners to create, edit, reorganize, merge, format, save, and print documents. $49.95.

☐ *PFS:Sampler* is a thirty-four-page collection of forms and related reports created by a cross section of *PFS:File* and *PFS:Report* owners. The forms in the *Sampler* are organized by industry and profession and are cross-referenced by applications type. The *PFS:Sampler* is being sent to all registered *File* and *Report* owners and to dealers who carry PFS: software. From **Software Publishing** (1901 Landings Drive, Mountain View, CA 94043; 415-962-8910).

☐ **Sybex** (2344 Sixth Street, Berkeley, CA 94710; 415-848-8233) has just released a book for business application programming with Pascal. *Doing Business with Pascal* is written for anyone interested in a modular, integrated programming approach to business systems. The 416-page book explains the necessary Pascal language extensions, program development, and system integration for special applications. Includes extensive examples of each application, including listings and sample runs. $15.95.

☐ As its premier product, **Multisoft** (120 East Ninetieth Street, New York, NY 10028; 212-534-0602) has released *Multi-Trieve*, a database management system. The program displays several records at a time in the form of a table, with field justification, column headings, decimal point alignment, and running totals for numeric fields. It features horizontal scrolling and supports the eighty-column card on the IIe. *Multi-Trieve* is menu-driven, eliminating complex commands, and it has fast search and edit routines. $199.

☐ The PrintMate 99 from **Micro Peripherals** (4426 South Century Drive, Salt Lake City, UT 84107; 800-821-8848) has a new price tag. Formerly $695, the printer now sells for $599.

☐ From the halls of **Spinnaker Software** (215 First Street, Cambridge, MA 02142; 617-868-4700) comes *Hey Diddle Diddle*, a collection of thirty classic nursery rhymes featuring color graphics and lively music. The product is designed to provide children between the ages of three and ten with hours of fun and learning. *Hey Diddle Diddle* helps children understand how words and rhymes create poetry and lets children take broken thoughts and rearrange them to make coherent verse. $29.95.

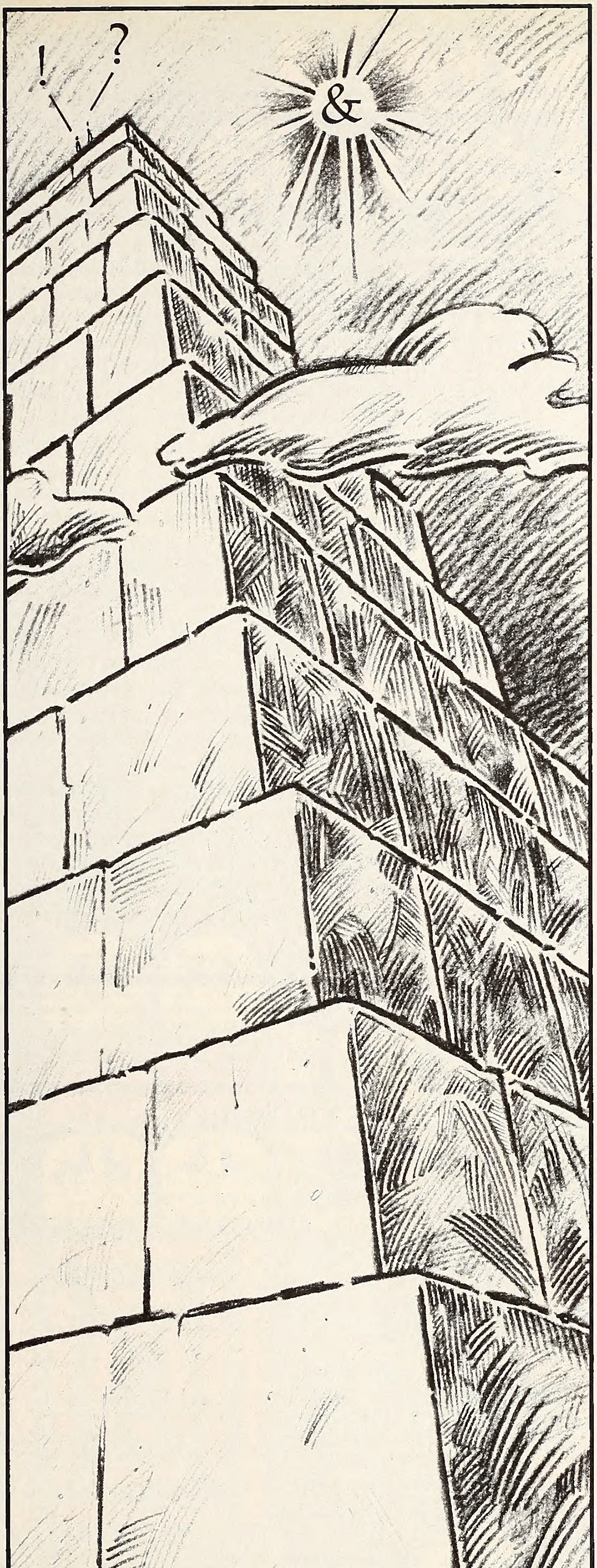
☐ Adding to its game controllers, **Wico** (6400 West Gross Point Road, Niles, IL 60648; 312-647-7500) has introduced the Boss joystick, the first in its line of low-priced game controls for the home market. The Boss is an Atari-type stick that's designed to fit the hand comfortably. It has a thumb-fire button, four rubber feet, and a five-foot cord. $19.95.

☐ A detachable keyboard for the II and II Plus is available from **Amkey** (220 Ballardvale Street, Wilmington, MA 01887; 617-658-7800). The Pro-100 offers keys supporting all existing Apple functions plus horizontal and vertical cursor movement, a separate number pad with enter key, auto-repeat, a relocated reset key, caps lock, and upper and lower-case keys. Included are twenty-two *VisiCalc* and twenty-five Applesoft keys, as well as eighteen programmable keys. The package includes the keyboard, lower-case chip, boot disk, and a six-foot interface cord. $265.

☐ **Compco Industries** (159 West Walnut Street, Painesville, OH 44077; 216-354-4186) offers the Compu-Table to keep your computer system organized. It features a slot for printer paper and a built-in cord and ribbon slot to keep wires and cables from tangling. Available in three sizes. Prices start at $53.95.

☐ The *Master Utility Disk* (MUD) from **WM Enterprises** (9348 Santa Monica Boulevard, Suite 101, Beverly Hills, CA 90210; 213-273-3412) helps you out when you blow it. More than twenty specialty routines let you alphabetize your catalog, determine free space on the disk and in the VTOC, find file addresses, have twenty-seven more sectors on your disk without removing DOS, undelete files, load DOS without losing your program in memory, track files, and more. $69.95.

☐ The *1983 Skarbek Software Directory* is now available with program descriptions for the Apple II and III. More than one thousand programs



We've moved ahead a byte in time since the Tower of Babel. And at least three things have improved.

Here they are:

Thing Number One: Our Communication's Improved.

With AMPER-MAGIC™ there's no need to know machine language to benefit from its power.

To your Applesoft™ programs you can attach slick, finished machine language routines (referenced by name, not by address, no less) in seconds. No separate BLOADing. And then, with AMPER-MAGIC commands, you can pass variables back and forth directly, just as you would with built-in Applesoft commands.

So, name your routine, perform the append procedure (only once please), and AMPER-MAGIC takes over — generating machine language speed where it counts: in *your* programs. Even your commercial programs. And, if for some reason you want to remove the routine, AMPER-MAGIC can do this too.

Thing Number Two: Our Communication's Improved.

AMPER-MAGIC also gives you access to hundreds of relocatable machine language routines, including those from magazines and other libraries, and - most notably - those from AMPER-MAGIC itself, which has over fifty routines of its own in two volumes.

Some of AMPER-MAGIC's routines are:

Find substring	Swap variables	GOSUB, GOTO a variable
Speed up Applesoft	Input anything	Search string array

And, of course, there are others too, for business, educational, recreational and word-process uses.

With AMPER-MAGIC you attach and access. It's really that simple. You can attach an unlimited number of commands to any program and you can move any set of commands as a unit to any other program.

A marvelous simplicity, wouldn't you say?

Thing Number Three: Our Communication's Improved.

Compatible with AMPER-MAGIC you can get the AMPER-MAGIC COMMAND LIBRARY. Already available is Volume 2, which contains twenty-seven machine language routines devoted to information displays and output capabilities.

With Volume 2 you get: a WAIT command, which specifies time in ordinary tenths of seconds either as a variable or as an expression which can be controlled by the program; the most powerful PRINT USING command yet devised; and, of course, twenty-five other commands. Future volumes include INPUT USING, fancy memory management and various sorting routines.

Imagine!
Only $75.00 for the original AMPER-MAGIC by Bob Nacon
Only $35.00 for AMPER-MAGIC COMMAND LIBRARY, Volume 2 **Plus shipping. MC and Visa accepted.**

Amper-Magic™

&

See your dealer or contact us.
Anthro-Digital. 103 Bartlett Avenue, Pittsfield, MA 01201
Telephone (413) 448-8278, Telex 467622

"The Company That Interfaces People and Computers"

For those of you who would like AMPER-Magic's routines in your own programs, just ask for Anthro-Digital's no cost licensing policy.

Also available from Anthro-Digital: Versacalc (several versions), Visicalc Formatting Aids, The Executive Secretary, The Personal Secretary, and Executive Speller, Hebrew II and Hebrew II Plus, S-C Macro Assembler, QuickTrace, Amper-Magic, The Rental Manager, F.A.R.M. accounting packages, The Performance Manager, Omniscan VideoDisc interface, and Flipper. DEALERS INVITED are listed, alphabetized by category: business, database, education, entertainment, graphics, home/personal, programming aids and utilities, special interest, and word processing. Included for each listing are program title, publisher, hardware requirements, description, and price. Also included is a dictionary of computer terms and a vendors list with addresses and phone number. From **Skarbek** (1531 Sugargrove Court, Saint Louis, MO 63141; 314-567-7180). $14.95.

☐ **Data Courier** (620 South Fifth Street, Louisville, KY 40202; 502-582-4111) publishes a comprehensive directory of serial publications for computer and software industries. *Select: A Guide to Computer and Software Publications* is designed for information specialists and others interested in industry publications. Each of the directory's more than five hundred listings includes name, content description, cost, publisher and address, average number of pages, advertising, and other information. $50.

☐ **Ring King Visibles** (215 West Second Street, Muscatine, IA 52761; 319-263-8144) has come out with the Diskette File Tray Caddy. The chrome and putty-colored rack holds either six full-size or eight mini-disk-storage trays. It rolls on barrel-type casters and protects disks from environmental problems. $129.95.

☐ Now you can get twelve DOS utilities on one unprotected disk at a low price. Each utility is written in listable Applesoft and fast machine language. It's called *DOSpac*, and its included utilities will alphabetize your catalog, find unreadable disk sectors, read or rewrite bytes, rename DOS commands, undelete deleted names, remove DOS to give you more room, make control characters readable, and more. From **Marshall Associates** (Box 12042, Huntsville, AL 35803; 205-881-7578). $24.95. Also from Marshall is *Mastersort 2.0*, a sort/merge parameter editor and machine language sort/merge program. *Mastersort* sorts random or sequential text files, merges two to five presorted files into one file, supports the Corvus hard disk system, and lets you chain files to Applesoft programs. $49.95.

☐ **CW Communications** (375 Cochituate Road, Box 880, Framingham, MA 01701; 617-879-0700) is alerting educators that the Executive Microcomputer Conference and Exposition held June 23–25 at the New York City Sheraton Centre has scheduled a special seminar for educators and administrators on Saturday, June 25. "Microcomputers in Education" will include a series of sessions highlighting important issues in educational computing. The seminar begins at 8:30 a.m. For preregistration, contact Louise Myerow at (800) 225-4698; (617) 879-0700 inside Massachusetts. $95.

☐ **C P & You Computer Learning Center** (5403 Elmer Drive, Toledo, OH 43615; 419-535-0130) is offering two summer programs of computer activities from June through September for children and adults. The first is a weekly program for children as young as four years of age that will include two hours of daily computer instruction. $55 per week. The second program offers one-and-a-half-hour classes, one day a week for four weeks. $40 per month. Courses for beginner, intermediate, and advanced students will be available; Basic, Logo, machine language, and advanced graphics will also be taught.

☐ *Legal Care for Your Software* shows how to take advantage of software protection laws and offers advice for writers and publishers of computer software. The 250-page publication includes tear-out contracts and forms ready to be used or modified. From **Velocity Engineering** (1720 Bancroft Way, Berkeley, CA 94703; 800-824-7888; 800-852-7777 in California). $19.95.

☐ Computer forms are available in small quantities from **Micro Format** (1271 West Dundee, Buffalo Grove, IL 60090; 312-537-2426). Available is a variety of stock forms, including continuous letterhead forms, report papers, continuous envelopes, labels, and continuous checks. The starter kit includes five hundred blank letterheads, five hundred labels, and five hundred continuous index cards. $24.

☐ Two games from **Sirius** (10364 Rockingham Drive, Sacramento, CA 95827; 916-366-1195) have made their way into the marketplace. *Fowl Play* puts you on the defensive as you protect your men from meal-seeking vultures. And watch out for that darned penguin (frnk, frnk)! $39.95. *Critical Mass* is Sirius's latest illustrated adventure. You're on the trail of a sicko who plans to blow up the world's five largest cities with thermal

Build up to the ultimate challenge.

If you thought you'd conquered every maze on earth, guess again.

Now there's the Maze Craze Construction Set™. The newest and greatest challenge of them all. You can actually build your own mazes from the ground up—and make them as mind-boggling as you dare.

That's only part of the fun. You can also create players and menacing monsters from your own imagination. Or nightmares. And choose speed, intelligence and sensitivity too.

To get the hang of it, play some of our most challenging combinations. Then, construct your own and save your favorites for future skirmishes.

Keep your wits about you, though. You are not alone. Maze Craze can be played with two players in the same maze. Try to beat your own game—or indulge in a little friendly vengeance with an adversary.

Meet the ultimate challenge: Maze Craze. With over a million possibilities, it's not likely you'll get bored in *this* century.

Look for the MAZE CRAZE Construction Set at your nearest dealer or call DTI Data Trek toll-free at **(800) 654-SOFT.** In California, call toll-free **(800) 652-DATA.**

Suggested Retail: $39.95

dti data trek

121 West E Street
Encinitas, California 92024
(619) 436-5055

Available for the Apple II/II+

Apple is a trademark of Apple Computer, Inc.

nuclear weapons. Think like a sicko, stop the sicko, save millions of lives. $39.95.

☐ **Strategic Simulations** (883 Stierlin Road, Building A-200, Mountain View, CA 94043; 415-964-1353) has released several games in its spring line-up. *Fighter Command* lets you rewrite or relive the strategies of the Battle of Britain. You can play either a German or a British commander trying to outwit the other in this thirty-four-turn campaign. Solitary play is possible with the computer as an opponent. $59.95. Next is *Knights of the Desert*, a re-creation of Rommel's campaign to push his Panzer divisions to Alexandria. As the German commander, your movement is hampered by supply shortages, while the British have unlimited supplies. As the British commander, you struggle westward to push Rommel's fast-moving troops back to El Aghelia and win the game. $39.95. Soviet forces have seized an oil-rich area in Saudi Arabia. The United States responds by sending its Rapid Deployment Force. *RDF 1985* is the second game in SSI's "When Superpowers Collide" series. After taking airfields and bringing in paratroopers, Marines, and infantry, your goal is to take control of the local towns, cities, and oilfields. $34.95. It's 1986 and Russia has won the great war in Europe and threatens to extend its airpower into the North Atlantic. This land-sea-air simulation of a hypothetical Soviet-NATO conflict is *North Atlantic '86*. You're the commander of NATO forces, and your duty is to hold Iceland against the Soviet assault. At the same time, your convoys to Britain must get through with supplies. $59.95. In the RapidFire line, SSI unveils *Cosmic Balance II* as its sequel to the successful *Cosmic Balance*. As an aspiring galactic emperor, you discover and colonize planets, establish commerce networks, organize production of supplies, starships, and research, and send starships out on missions of conquest. Five scenarios are included, and you can create your own, too. $39.95. The second, improved edition of *Computer Ambush* is here. In addition to the hand-to-hand combat found in the first edition, this one lets you create your own soldiers, rearrange their characteristics, and distribute weapons as you like. It's written in assembly language for faster execution of the game. $59.95; current owners can update for $20. Strategic also has fast-running updates for the following games: *Computer Bismarck*, *Guadalcanal Campaign*, *Cartels & Cutthroats*, *Computer Air Combat*, *Operation Apocalypse*, *Computer Baseball*, and *Napoleon's Campaigns: 1813 & 1815*. $10 each. *Computer Quarterback*, $15.

☐ *Functions* is a courseware module that covers the properties of functions and their graphical representations. Aimed at eleventh and twelfth-grade math students, the program is from **Avant-Garde Creations** (Box 30160, Eugene, OR 97403; 503-345-3043). $29.95. *Hi-res Computer Golf II: Pro Courses Series* has graphics and sound that improve on those of its predecessor. The series consists of a master disk with professional courses disks being available separately. $29.95. Finally, *Paint Master Scene Utility* teaches, guides, and provides you with routines to create colorful scenes. No royalties are required for use in your own programs. $34.95.

☐ It's not a golf game, even though it's from **GolfSoft** (10333 Balsam Lane, Eden Prairie, MN 55344; 612-941-2172). *Statistician* is a self-help program that examines impact feel, initial direction, resulting direction, trajectory, and relative distance of each shot. With the collection of these statistics, the program aims to smooth out and lower the score of the average golfer's game. $34.95.

☐ Software that supports file transfers between Apple IIs and IBM Personal Computers has been announced by **Trax** (8948 West Twenty-fourth Street, Los Angeles, CA 90034; 213-670-9699). *Direct Connect* requires no communications adapter or serial interface. The program uses the IBM's built-in cassette port and the I/O port of the Apple. Transfer rate is ten thousand bits per second, and the software checks for errors during transfer. $170.

☐ A new offering from **Microsoft** (10700 Northup Way, Bellevue, WA 98004; 206-828-8080) combines CP/M, 64K, eighty-character display, and Basic on one card. The Premium SoftCard IIe fits into the video slot of the IIe. After installing this Z-80-based card, you specify the operating system you want by booting the appropriate Z-80 or 6502A disk. With the additional RAM, you can now have 128K total memory. The Premium SoftCard IIe is compatible with Microsoft's Fortran and Cobol compilers. $495.

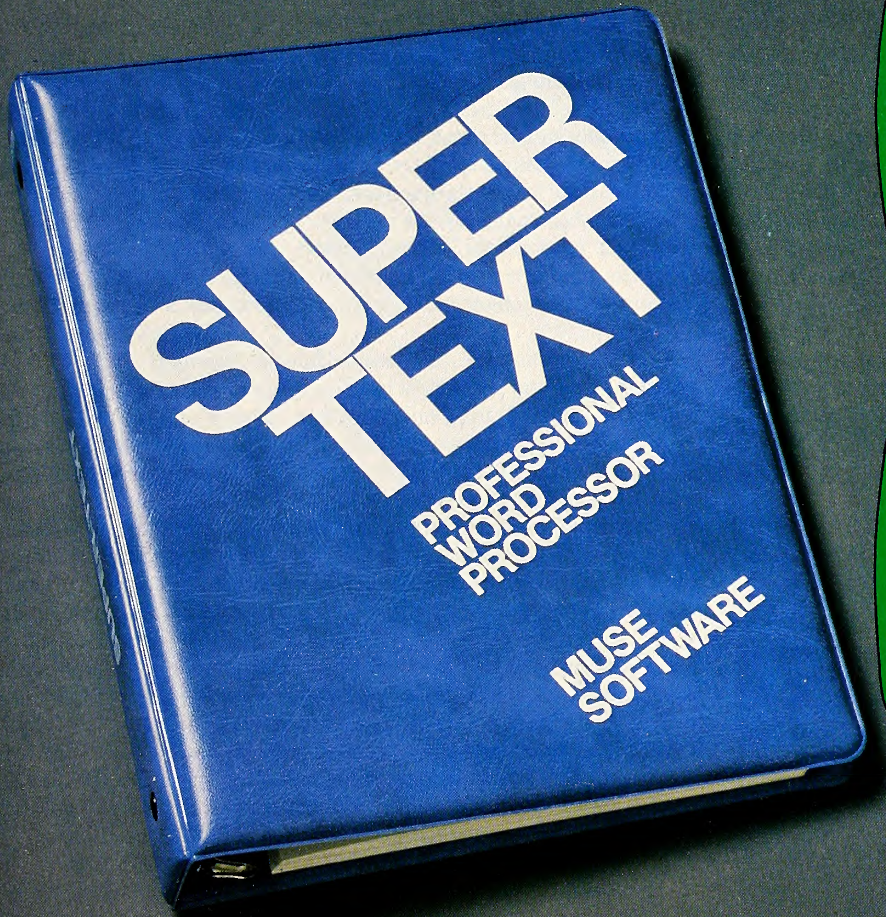
☐ "What-if" department: What if you could have a personal tutor come to your office and teach you about *VisiCalc*? Well, **Personal Tutor Associates** (Box 246, Clinton, MD 20735; 301-856-2280) offers the *VisiCalc Audio Course*, a set of three tapes that walk you through *VisiCalc* in a conversational manner. Tape one introduces *VisiCalc* commands; tape two shows you how to use its functions; tape three teaches you how to design and produce templates and overlays. $49.95.

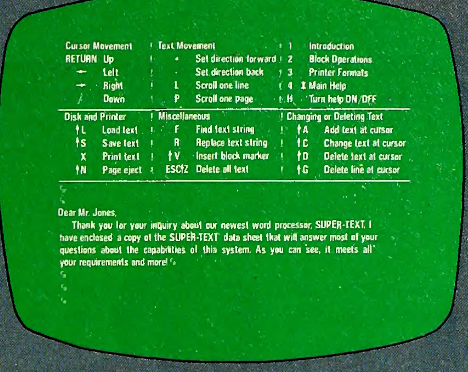
☐ **TMQ Software** (82 Fox Hill Drive, Buffalo Grove, IL 60090; 312-520-4440) has released *File-Fax Version 2.0*, an update of its database system. Written in machine language, the program is completely memory based—no more disk swapping. It has numerous text-editing features and "help" screens that review command functions. *File-Fax* has an eight-level sort and wide-range search capability. Ranges such as greater-than or less-than can also be used. $149.

☐ Engineers, scientists, mathematicians, economists, linguists, and other professionals can now have Greek and other special mathematic characters in *WordStar*. **Techware** (2510 Cresta de Ruta, Eugene, OR 97403; 503-343-0566) has released *Chartech*, an add-on to *WordStar*, that allows the word processor to print ninety-four special characters on a dot-matrix printer. It can also drive daisy and thimble printers. Special utilities make it easy to design new characters and add them to the system. $65.

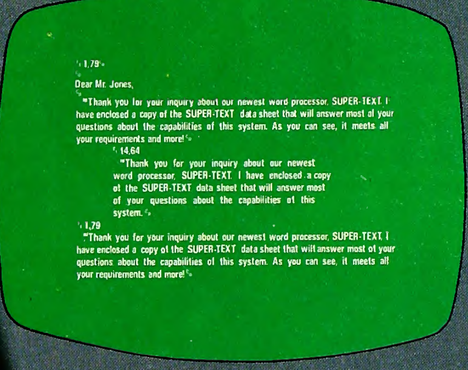
☐ **Telos Computing** (Santa Monica, CA) has developed *TeloFacts*, a survey and data analysis program for anyone who must administer and analyze data from surveys, questionnaires, tests, or polls. **dilithium Press** (11000 S.W. Eleventh Street, Suite E, Beaverton, OR 97005; 503-646-2713) publishes *How To Use TeloFacts*, a guide to using the program. The 120-page book shows you how to develop and design the questionnaire, how to enter answers, and how to analyze and tabulate data. $9.95. To help you overcome the fear of computers, dilithium publishes *Bits, Bytes and Buzzwords*, a 110-page book that started out as a pamphlet and will eventually be broadcast as a five-part series on public television. The first four sections explain a computer system, each summarized with highlights of the chapter. The book ends with a glossary of

SUPER-TEXT Professional does everything the competition does, except one thing.





ON-LINE "HELP"



ON-SCREEN FORMATTING

It doesn't cost nearly as much.

SUPER-TEXT PROFESSIONAL (40/80) gives you valuable **ON-SCREEN FORMATTING** and **"HELP" GUIDES***, unique **AUTOLINK, SPLIT SCREEN** and **MATH MODE.** Plus all the functions you'd expect from an expensive word processor, like an 80 column screen display.* Automatic page headers and footers. Automatic page and chapter numbering. Preview Mode to check page endings. Easy text editing. Special block operations for text copy, save and delete. Automatic tabbing and formatting. Multi-file search and replace. Superscripting and subscripting. All for only: **$175.00**

Compare Super-Text Professional with the other professionals. You'll find there's no comparison. Available now for Apple II+ and Apple IIe.

*These features available with the use of an 80 column board.

SUPER-TEXT HOME/OFFICE (40/56/70) is the best choice for introducing word processing efficiency into your home. It offers you the basic text editing features of Super-Text Professional, and includes special features tailored for home use. It provides 40, 56 or 70 column screen display without any additional hardware. The Character Design Mode lets you create display characters in foreign languages, with special symbols, for any personal application. All at a price you can live with: **$125.00**

Available now for Apple II+ and Apple IIe.

INTRODUCTORY OFFER!
YOUR CHOICE SUPER-TEXT PROFESSIONAL OR HOME/OFFICE
$99.

MUSE®
SOFTWARE

347 North Charles Street, Baltimore, Maryland 21201 301/659-7212

computer terms. $7.95. A collection of programs designed to expand the horizons of *VisiCalc* users is presented in *32 Different Worksheets for the VisiCalc User.* Many of the worksheet examples can be extended or modified; 150 pages and 140 illustrations. $19.95. Combined with disk, $29.95.

□ The Guild Computer Rack fits over the Apple, holds two disk drives and a monitor, and accommodates a cooling fan if you have one. Constructed of solid mahogany or ash, no assembly is required of the rack. From **Guild Computer Rack** (225 West Grand Street, Elizabeth, NJ 07202; 201-351-3002). Mahogany, $69.95; ash, $54.95.

□ New science and English education programs are available from **BrainBank** (220 Fifth Avenue, New York, NY 10001; 212-686-6565). *The Skeletal Systems* contains five programs, each covering different parts of the system. Included are line drawings and a teacher's guide. Disk or cassette, $70; $32 for backup copies. *Word Functions* contains nine separate programs in two parts. The first part covers homonyms and synonyms; the second part covers antonyms and homonyms that cause a lot of trouble for students of English. A teacher's guide is included, along with a review test. Two disks or cassettes, $99; $37 for backups. *Classes of Nouns* helps teach recognition of common and proper nouns, as well as special classes of nouns; included is a teacher's guide. Disk or cassette, $60; $22 for backups.

□ **ATI** (3770 Highland Avenue, Suite 202, Manhattan Beach, CA 90266; 213-546-4725) is offering a line of disk-based interactive training products for the IIe with an eighty-column card. Training programs cover *MultiPlan*, *VisiCalc*, *BPI General Accounting*, and MBasic. $75.

□ **MinuteWare** (Box 2392, Columbia, MD 21045; 301-995-1166) is marketing *Minute Manual for Apple Writer II.* The manual gets you started with simple instructions for six basic word processing procedures as well as more advanced ones. Included are instructions for producing all of the Epson printer print styles. The book also has chapters on word processing concepts, word processing hardware, and software enhancements for the II Plus. $7.95.

□ A compact programmable switching unit that expands one serial port to three ports through software control is being introduced by **Digital Laboratories** (600 Pleasant Street, Watertown, MA 02172; 617-924-1680). ESP-1 uses DIP switch-control codes that recognize the user's symbol selecting each port. Permits computer output to letter-quality printers, dot-matrix printers, plotters, or any RS-232 peripheral. $395.

□ *Quick Check* is a bookkeeping system that lets you print checks, pay bills, keep checking and payable accounts, and get income and expense reports. It features clear displays, few keystrokes per command, error warning, and more. Available from **Chuck Atkinson Programs** (Route 5, Box 277-C, Benbrook, TX 76126; 817-249-0166). Requires 56K and CP/M. $250.

□ **Arrow Instructional Systems** (Box 543, Newport, RI 02840; 401-847-1955) has released its *SAT Preparation Program* to be used by students getting ready for the college entrance examination. The program introduces students to the test's parts and challenges them with a combination of verbal and mathematical questions. It has a built-in timing and scoring function that converts to the student's equivalent SAT score. There's also a tutor mode that gives an analysis of how a correct answer is arrived at and why other answers are incorrect. $199.95.

□ **Nexa** (Box 26468, San Francisco, CA 94126; 415-387-5800) has debuted with two games. *Cybernation* is a strategic war game that lets you travel to the year 3922 and battle the cyborg Entontions. The disk is so packed that you need 64K to play it. $39.95. *Delta Squadron* is a strategic space war simulation. As the Legion Alliance squadron commander, you must direct up to thirty-four small fighters down a long trench and destroy the enemy's main power induction inverter. You must assign ships to fly cover while others are set for on-course targeting. This one requires 64K also. $39.95.

□ Two computer-assisted instruction programs are available from **Cygnus Software** (8002 East Culver, Mesa, AZ 85207). *Metric System Tutor* covers the development of the linear, volumetric, and mass units, and the conversion of metric units. $66. *Characteristics of a Scientist* shows the student how curiosity, observation, skepticism, and open-mindedness enable the scientist to gather information and arrive at conclusions. The student is given the opportunity to put the information to practice through a series of puzzles, quizzes, and tricks played by the computer. $39.

□ **Technical Educational Consultants** (11 Barby Lane, Plainview, NY 11803; 516-681-1773) is sponsoring the Computer Assisted Instruction: Design and Development Workshop, which will be held June 6–7. The seminar will show how to design and write effective CAI. Topics include instructional design, writing languages, computer concepts, lesson flows, and communication techniques. There will also be hands-on experience designing a CAI lesson. For information, contact Dr. Jerrold Kleinstein at TEC.

□ **Personal Computer Products** (16776 Bernardo Center Drive, San Diego, CA 92128; 619-485-8411) offers Appli-Card as a one-board answer to several problems. The card has 64K, giving you 63K for program execution, and it supports all ninety-six ASCII characters for input and output. The card operates simultaneously with the Apple's 6502 processor. If you have the company's 128K RAM extender, you can use DOS and CP/M at the same time. Appli-Card comes in two versions. 4MHz Z-80A, $295; 6MHz Z-80B, $375.

□ Good-deed doers might want to check out *Brainteaser Boulevard* from **California Pacific Computer** (757A Russell Boulevard, Davis, CA 95616; 916-756-2921). As a Scout, your mission is help old ladies across the boulevard without either of you getting flattened by cars and speeding trucks. $24.95.

□ **Discount Software** (6520 Selma Avenue, Suite 309, Los Angeles, CA 90028; 213-837-5141) is offering a catalog full of detailed descriptions of business programs and lists of games. Separate sections for CP/M, Apple, and sixteen-bit software are provided. $5. Catalog and update bulletins, $10 for a year.

□ The eRAM 80 from **Quadram** (4357 Park Drive, Norcross, GA 30093; 404-923-6666) gives you eighty-column display and 64K of extra memory. The display screen can be programmed for either standard forty-column or extended eighty-column text display, and you can switch back and forth between the two formats. $159.

☐ As a follow-up to its successful *Deadline*, **Infocom** (55 Wheeler Street, Cambridge, MA 02138; 617-492-1031) has introduced *Witness*, a whodunit patterned after classic mystery novels. You've just witnessed a death, and your job as detective is to sift clues to determine if it was murder, suicide, or an accident. No two-word commands here. Infocom's Interlogic programming system, with its more than six-hundred-word vocabulary, requires you to use complete sentences. $49.95.

☐ **Technical Education Research Centers** (8 Eliot Street, Cambridge, MA 02138; 617-547-3890) will be hosting its Microcomputers in Education seminars in math, science, and computer literacy at Trinity College. The seminars run July 11–22, with a choice of one-week or two-week sessions. Topics include Logo, Basic, and the use of microcomputers in algebra and geometry, trigonometry and calculus, natural sciences, and physical sciences. TERC's second annual Summer Workshop Series will be held at the TERC offices in Cambridge. The four-day workshops will provide intensive training in several topics for teachers and administrators at all levels, elementary through college. Subjects include software development, Logo, simulations, Pascal, and micros in the science lab. For more details, contact TERC.

☐ **The Boston Company** (One Boston Place, Boston, MA 02106; 617-722-7960) has released a new feature for *Micro PMS*, its portfolio management system that includes menu-driven investment software, monthly data disks covering 1,500 popular stocks, and investment strategy recommendations. The new feature, Screen, lets you quickly find stocks that meet your own investment criteria. More than fifty items are available for selection, including growth, yield, quality, price performance, book value, and others. Screen typically reviews five hundred stocks per minute. Price is still $595.

☐ Apple's Lisa is no longer the only Apple with a detachable keyboard. You can have one, too. **Executive Peripheral Systems** (800 San Antonio Road, Palo Alto, CA 94303; 415-856-2822) has introduced a detachable keyboard that plugs into the motherboard and gives you all ASCII characters. Modules that give you one-stroke editing and programming commands plug into the keyboard; when used with popular word processing and business packages, they replace complex commands with single keystrokes. Most keys have auto repeat, and a twenty-one-key numeric pad is built in for data entry. Includes the Basic and DOS module and interface card. $399.95. Extra modules, $32.95 each.

☐ **Micro Lab** (2310 Skokie Valley Road, Highland Park, IL 60035; 312-433-7550) has lowered the price of *The Learning System* from $150 to $75. This program, which prepares drills and tests for students, is also now unprotected.

☐ **Plain and Simple Software** (9003 Lexington N.E., Albuquerque, NM 87112; 505-293-2448) has released *ASP*, a program for the Apple III that combines the features of a word processor, database, and mailing-list program. *ASP* can write a word processing program at your request that will allow *Apple Writer III* to access the files and provide personalized letters. It allows each record to have ten fields, four of which can be designed to be continually sorted. Field labels, field entry lengths, and record lengths can be changed at any time. Two printing formats are maintained in memory. Requires 256K. $99.95.

☐ *The Assembler* is an assembly language from **MicroSparc** (Box 639, Ten Lewis Street, Lincoln, MA 01773; 617-259-9710) documented for beginners, with features for the expert. It gives you up to 29.5K of usable memory for source programs and includes a global editor with search and replace features. *The Assembler* lets you write your own subroutines and call them by name. $69.95. You can program Applesoft-like programs that are converted directly into machine language with MicroSparc's *MacroSoft*. It has commands that don't exist in Applesoft, and programs run up to ten times faster than compiled Applesoft programs. Requires *The Assembler*. *MacroSoft* and *The Assembler* package, $99.95.

☐ *The Personal Computer—An Industry Source Book* helps you locate products, companies, and personnel. From **Chromatic Communications** (Box 3249, Walnut Creek, CA 94598; 415-945-1602), the book contains more than twenty-five hundred company listings that include product descriptions, industry contacts, and location. Classifications are by hardware, software, and accessory manufacturers; franchisers; support services; distributors; publishers; and periodicals. $42.50.

☐ **State of the Art** (3183-A Airway Avenue, Costa Mesa, CA 92626; 714-850-0111) now has its *Sales Invoicing* module, which integrates with the *Accounts Receivable*, *Inventory Control*, and *General Ledger* modules. *Sales Invoicing* produces invoices directly from shipping documents and automatically generates back orders and sales journals. The module also includes a reporting system that tracks sales, analyzes gross profits, and monitors sales commissions. Apple II version, $395; Apple III, $495.

☐ Seven new products from **Sophisticated Software** (650 Foothill Boulevard, La Canada, CA 91011; 213-790-9052) are available. *The Little Black Book* keeps lists of names, notes, appointments, and a daily agenda. You can add any number of notes to client files in any form you choose. No manual; just turn it on and go. $249. *Micro Mass Mailer* sends lots of messages without lots of work on your part. The computer prints your message right inside the envelope, while addressing them on the outside. You just add postage. $129. *Le Menu* eliminates run and brun commands when you want to use programs. With just two keystrokes, you can get programs up and running. The program works with floppy or hard disk systems. $49.95. *The Scheduler* lets you schedule events as far in advance as you like. You can review your schedule onscreen or print a hard copy. $99. *Roll-a-File* stores, searches, and retrieves information in your electronic card file. You can search for clients by age, name, account number, zip code, or any other criteria. $79. *Print-a-File* works in conjunction with *Roll-a-File*, letting you print mailing lists and phone lists using the same selection techniques built into *Roll-a-File*. $99. Finally, *The Labeler* lets you design and print your own custom labels. Print any quantity you like, any way you like. It prints large type and centers lines automatically. $19.95.

☐ **Proximity Devices** (3511 North East Twenty-second Avenue, Fort Lauderdale, FL 33308; 305-565-2188) now has an extended version of *Word Challenge*, a word-search game previously available only on IBM Personal Computers. This version makes use of an eighty-nine-thousand-word lexicon and has twenty-six levels of difficulty for players of all ages and abilities. The game includes a score keeper, automatic timer, three different board sizes, the option to create your own boards, and the ability to rotate boards for different perspectives. $39.95.

☐ A replacement speaker for the Apple II Plus is available from **The Alien Group** (27 West Twenty-third Street, New York, NY 10010; 212-741-1770). The Alien speaker generates a much louder volume and clearer tone than the computer's speaker, making it a must for dedicated noisy game players and music enthusiasts. The speaker's cable comes with the same connector used by the Apple speaker for simple installation. A built-in high-frequency filter switch is included. $24.95.

☐ Following games with games, **Broderbund Software** (1938 Fourth Street, San Rafael, CA 94901; 415-456-6424) has released *Gumball*. Your job in the gumball factory entails sorting different gumballs while zapping explosive-laced gumballs that are left by dental hygiene vigilantes. $29.95. In *Lode Runner*, you play a galactic commando responsible for uncovering the Bungeling Empire's (remember those guys?) secret gold depository. Jumping and drilling skills are a plus. In case you get bored, *Lode Runner* lets you create your own screens and design your own *Lode Runner* game. No programming knowledge necessary. $34.95.

☐ From **Micro Program Designs** (5440 Crestline Road, Wilmington, DE 19808; 302-738-3798) come the following: *Dr. Grafix* is a hi-res shapes-and-graphs utility designed for easy use by beginning and intermediate programmers. Shape tables can be custom-designed, edited, and reorganized to suit specific programming needs. Included is a hi-res character set, plus instructive demonstrations of animation techniques. The disk contains routines and tutorials for converting numeric data into line, bar, and pie charts. $19.50. *Mr. Krypto* is a word game for gamers ages six and up. Games involve unscrambling words, coded words, crazy sayings, secret messages, and cryptograms. $29.50. *Stock Market Tycoon* is a money game in which one or more players watch the accelerated stock price board and price charts, then call the broker to buy, sell, or sell short. No knowledge of the stock market is needed. $29.50. *Stock Watch* is for the investor of the real stock market. Up to forty-five user-selected stocks can be watched. No hook-ups are involved. High, low, and closing prices and volume data are entered from the newspaper stock-market pages. Graphic data displays include exponential moving averages. $59.50. All of Micro Program Designs's programs come on unprotected disks.

All About Applesoft

by Doug Carlston

Last month we began the design of a program that tracks a person's weight over an eleven-week period and then plots the results on the hi-res screen. The program mixed text and graphics on hi-res page one, thanks to a special machine language subroutine and its data file, both of which we poked in and saved as a pair of binary files called Character Generator and Character Table.

Because these two machine language routines occupy the same space in memory that an Applesoft program does, we had to force the Apple to load the Applesoft program at an address higher than the usual $800. We did this by changing the start-of-program pointers in the Apple before loading our main program.

If we were to name our main program *Diet*, the *Diet* hello program might look like this:

```
10  D$ = CHR$($): PRINT D$"BLOAD CHARACTER GENERATOR"
20  PRINT D$"BLOAD CHARACTER TABLE"
30  HGR
40  XX = PEEK ( - 16302)
50  POKE 103,1: POKE 104,64: POKE 16384,0
60  PRINT D$"RUN DIET"
```

This little program will load in the machine language routines, turn on the hi-res graphics, and tell the Apple to load and run the next Applesoft program, *Diet*, up in high memory (above hi-res page one).

We started designing our main program last month too. Here is our progress to date:

```
10   DIM A%(200):HC = 1: GOSUB 1000: GOSUB 3000: GOTO 100
19   REM ** KEYBOARD STROBE **
20   POKE - 16368,0
21   X = PEEK ( - 16384): IF X < 128 THEN 21
22   IF X = 155 THEN FLAG = NOT FLAG: IF FLAG THEN POKE -
     16303,0
23   IF NOT FLAG THEN POKE - 16304,0
24   A$ = CHR$ (X - 128): POKE - 16368,0: RETURN
100  REM ** MAIN PROGRAM **
110  GOSUB 20: IF A$ = "L" THEN 150
120  IF A$ = "E" THEN 300
130  IF A$ = "S" THEN 500
140  GOTO 110
300  REM ** ENTER DATA **
301  VTAB 5: CALL - 958: PRINT : PRINT TAB( 5)"N(EW PERSON":
     PRINT : PRINT TAB( 5)"A(PPEND DATA": GOSUB 20: IF A$ =
     "N" THEN K = 0:FG = 1
305  PRINT : PRINT "ENTER 0 TO QUIT": PRINT
310  PRINT "ENTER WEIGHT FOR DAY ";K + 1;: INPUT ":";WEIGHT:
     IF WEIGHT = 0 THEN A%(K) = 0: GOSUB 3000: GOTO 100
315  WEIGHT = WEIGHT + (100 - WEIGHT) * (WEIGHT < 100) -
     (WEIGHT - 190) * (WEIGHT > 190)
320  A%(K) = 20 + (190 - WEIGHT) * 1.6:K = K + 1
330  IF K = 1 OR FG = 0 THEN FG = 1: HPLOT 28 + 3 * K,A%(K - 1):
     GOTO 340
335  HPLOT TO 28 + 3 * K,A%(K - 1)
340  GOTO 310
1000 REM PRINT GRAPH
1005 GOSUB 2000: HCOLOR=3: PRINT CHR$(12): VTAB 1:HTAB
     1: PRINT "LBS"
1010 FOR X = 0 TO 9: VTAB 22 - (X * 2 + 1): HTAB 1: PRINT 100 +
     (10 * X): HPLOT 23,X * 16 + 20 TO 27,X * 16 + 20: NEXT
1020 VTAB 23: HTAB 7: FOR X = 1 TO 11:PRINT X" ";: HPLOT 28 +
     21 * X,168 TO 28 + 21 * X,172: NEXT : VTAB 24: HTAB 19:
     PRINT "WEEK";
1030 HPLOT 25,0 TO 25,170 TO 270,170: GOSUB 2010:
     RETURN
2000 REM TURN ON HI-RES CHARACTER SET
2005 PR# 0: IN# 0: POKE 54,0: POKE 55,20: VTAB 24: HTAB 1:
     PRINT" ";: POKE 972,16: POKE 974,32: HTAB 1: PRINT " ";:
     CALL 43089: RETURN
2010 REM TURN OFF HI-RES CHARACTER SET
2015 VTAB 24: HTAB 1: PRINT " ": PR# 0: IN# 0: CALL 43089:
     RETURN
3000 REM ** SET UP TEXT PAGE **
3005 GOSUB 2010
3010 HOME : HTAB 10: INVERSE : PRINT "WEIGHT TRACKING
     CHART": NORMAL
3020 VTAB 5: PRINT TAB( 5)"L(OAD DATA FROM DISK"
3030 PRINT : PRINT TAB( 5)"E(NTER NEW DATA FROM
     KEYBOARD"
3040 PRINT : PRINT TAB( 5)"S(AVE DATA TO DISK"
3100 RETURN
```

As you can see from the menu in lines 3020 through 3040, there are three major segments of this program. One permits you to enter new data from the keyboard (this is the section we worked on last month). The other two permit you to load or save data to disk. We have already done this several times in other programs, so it should be pretty routine by now. But let's take a look at a solution anyway. There are always a few new bells and whistles we can add.

Let's design the routine to save data to disk first:

```
500  VTAB 5: CALL -958: GOSUB 30
510  PRINT D$; "OPEN";NAME$: PRINT D$; "WRITE";NAME$
515  KT = 0
520  PRINT A%(KT):KT = KT + 1: IF A%(KT) <> 0 THEN 520
530  PRINT 0: PRINT D$; "CLOSE";NAME$
540  GOSUB 3000: GOTO 100
```

Line 500 jumps to a subroutine (which we'll write in a minute) that selects a file name. We are putting this function in a subroutine so that we'll be able to select a file name when we retrieve a file as well. If we didn't put the code in a subroutine, we'd have to write essentially the same code in both sections of the program.

The subroutine returns the file name to us in a variable called NAME$.

Line 510 opens a file by the name in NAME$ and prepares us to write to it. Then line 520 prints all of the data from the array A% into the text file until it comes across a zero (which we used to indicate the end of data). Line 530 then writes that final zero into the text file (so we know when we've gotten to the end of it) and closes the file.

Pretty simple. And the subroutine for entering file names should be old hat by now too. We've used it before:

```
29   REM ** DISK FILE NAME **
30   PRINT "ENTER NAME (";: INVERSE : PRINT "RET";: NORMAL :
     PRINT " FOR CATALOG)";: INPUT ": ";NAME$
31   IF NAME$ = "" THEN PRINT : PRINT D$; "CATALOG": GOSUB
     20: PRINT : GOTO 30
32   RETURN
```

The disk read program is just as simple:

```
150  VTAB 5: CALL -958
160  GOSUB 30
190  PRINT D$; "OPEN";NAME$: PRINT D$; "READ";NAME$
195  K = 0
200  INPUT A%(K): IF A%(K) <> 0 THEN K = K + 1: GOTO 200
210  PRINT CHR$(4);" CLOSE";NAME$
```

However, once we've read in the data, we still need to plot it. (Perhaps we should have put the data-plotting routines in the data-entry portion of the program into a subroutine so we could have used them here as well, but we didn't.)

```
220 HCOLOR= 3: HPLOT 28,A%(0): FOR X = 1 TO K - 1: HPLOT
    TO X * 3 + 28,A%(X): NEXT
230 GOSUB 2000: VTAB A%(X - 1) / 8: HTAB ((X - 1) * 3 + 28) / 7
    + 1: PRINT NAME$
250 GOSUB 3000: GOTO 100
```

Line 220 is a straightforward plot of the data points we've read in. However, line 230 is a little added touch. Since we know the name of the file we just read in, we can use that name to label the line we've just drawn on our graph. So we now go to subroutine 2000, which switches printing to the graphics page, and then all that remains is to figure out where on-screen to print the variable NAME$. The vtab and htab formulas in line 230 may look complicated, but they aren't, as long as you keep in mind that all text characters are exactly eight pixels (dots) high and seven wide.

There is one more little detail we should add to this part of the program. If someone using this program types a file name incorrectly, the program will search the disk for a file that isn't there. Eventually it will give up and the program will bomb.

Having this happen is annoying, particularly when you have something on-screen that you don't want to lose by starting over. So let's try a little sophisticated error-trapping. The first thing to do is to warn the Apple—*before* it starts searching the disk—that we want to handle error messages ourselves:

```
180 ONERR GOTO 290
```

If any sort of error occurs after the Apple has read line 180, control will jump to line 290. No error message will be printed. So it's up to us to figure out what is wrong and to advise the program user as to how to correct the problem.

Whenever an error occurs, the Apple places a number indicating the type of error in memory location 222. To figure out what's going on, we can peek at location 222 and then look up the number in a chart. There are two charts in the manuals that come with the Apple hardware. One covers Applesoft errors and can be found on page 136 of the *Applesoft Basic Programming Reference Manual*. The other covers DOS errors and is located on pages 114 and 115 of *The DOS Manual* (DOS 3.3).

Let's not worry about most of the possible errors. What we are looking for is the file-not-found error, which our chart says is error code number 6. If other errors crop up, we'll just have to deal with them one by one.

```
290 IF PEEK (222) = 5 OR PEEK (222) = 6 THEN PRINT : PRINT
    "FILE NOT FOUND": GOTO 292
291 PRINT "ERROR #"; PEEK (222)
292 POKE 216,0: GOSUB 20: GOSUB 3000: GOTO 100
```

The important thing about controlling your own error codes is that when you've finished notifying the user of the problem, you maintain control of the machine and can keep the program running. In this case, we'd print an error message, turn off the error flag (that's what the poke 216,0 does), wait for any key to be pressed, restore the text page menu, and then jump to the main menu part of the program. It's a clean solution.

That's the end of the *Diet* program. May it bring you hours of entertainment. Now let's get back to basics and take another look at the hello program that we had to run before *Diet* would work.

In the *Diet* hello program we loaded in two machine language routines, turned on the hi-res graphics, set lomem, and then ran the main program, *Diet*. Whenever you have a large number of administrative tasks of this sort, there is an alternative way of handling them, that being to use an *exec* file. An exec file is a text file containing a list of instructions from you to the computer. Your Apple executes them one at a time until the list is finished.

Let's create an exec file to run *Diet*. First, though, let's simplify our *Diet* hello program so that all it does is turn on hi-res graphics and reset lomem. Delete lines 10, 20, and 60 and then type *save Diet hello*.

Next let's create our exec file. The following Applesoft program will write out a text file to the disk.

```
10 D$ = CHR$ (4)
20 PRINT D$"OPEN SETUP": PRINT D$"WRITE SETUP"
30 PRINT "RUN DIET HELLO"
40 PRINT "BLOAD CHARACTER TABLE,A4352"
50 PRINT "BLOAD CHARACTER GENERATOR,A5120"
60 PRINT "RUN DIET"
70 PRINT D$"CLOSE SETUP"
```

The text file is called Setup and contains the four commands printed in lines 30 through 60. To see how it works, first run the program above to create the text file. Then type *exec Setup*. This command instructs the computer to execute the commands in the Setup file one by one until all are carried out. It's now possible to create a very simple prehello program that looks like this:

```
10 PRINT CHR$(4);"EXEC SETUP"
```

One of the most common uses for the exec command is to retrieve an Applesoft program that's been captured in a text file. Capturing a program in a text file allows it to be loaded into a word processing program. Try this. Load in any Applesoft program, such as the *Diet* program we just wrote. Then add the following line to the beginning of the program:

```
1 D$ = CHR$(4): PRINT D$;"OPENPROGRAM": PRINT D$;"WRITE
  PROGRAM": LIST 2,: PRINT D$:"CLOSE PROGRAM": END
```

Then run the program. Your disk will whir for a minute and then stop. If you catalog the disk, you will discover a new text file on it called Program. You can load this file into most word processors, which makes it possible to edit programs with all of the powerful features of a word processor, such as universal search and replace. Some of *Softalk*'s writers use such a system to incorporate programs into their articles, which cuts down on errors and keeps them from having to type in the programs twice.

It's even easier to convert a text file back into an Applesoft program. Try this. Type *new* and then *list*. Your program is gone, right? Now type *exec program*. Type *list*. Your program has been read back into memory. Once you start editing your Applesoft programs on a word processor, don't be surprised if you wonder how you ever managed before.

By now you have been exposed to virtually every command in Applesoft—there are only a couple of unusual ones left, and you will pick them up in the next couple of lessons. In addition, you have started to learn a great deal about the internal organization of your Apple and how to peek or poke it when you want it to do something Applesoft isn't equipped to handle. We have even included a few DOS commands, such as the text file open, close, read, write, and exec commands, since DOS is such a fundamental part of your programming equipment.

What remains is practice, and so we'll try over the next few months to design a number of useful and interesting programs that give you an opportunity to try out the vocabulary you've learned.

Each month we'll propose a project and analyze it together. The following month, we'll offer a solution (not the only one or, for that matter, the best) to help if there are any areas that stumped you. And we'll occasionally offer short machine language routines that you can incorporate into your Applesoft programs to give them enhanced power.

One of the most basic machine language routines you'll need (at least if you write any games) is a sound generator. As you may have read, the Apple's speaker can be toggled by any reference to location −16336, for example, *X = peek (− 16336).* If you reference it repeatedly from Applesoft, you can even get sound, in a manner of speaking:

```
10  FOR Y = 1 TO 10: S = - 16336: X = PEEK(S) - PEEK(S) +
    PEEK(S) - PEEK(S): NEXT
```

If you want to create sounds more sophisticated than these low burps, you'll have to have a machine language routine toggle the speaker; Applesoft is just too slow. Here is one such routine, poked in from Applesoft.

```
10  FOR X = 770 TO 788: READ Y: POKE X,Y: NEXT
20  DATA 173,48,192,136,208,4,198,7,240,8,202,208,246,
    166,6,76,2,3,96
```

This program pokes a very short, very simple machine language routine into page three of memory (right below the text area). If you'd like to see what it looks like, enter the Monitor by typing *call −151.* You should see an asterisk on the left margin where you usually would see the Applesoft bracket. Then type *302L.* This is a command to the Apple to list whatever is in memory, starting at hex location 302 (the numbers in the Monitor mode are all hexadecimal).

The numbers at the left are the hex equivalents of the decimal numbers you poked into memory. For instance, AD is the hex equivalent of 173. To the right, you will find a *disassembly* of the hex numbers, which means that the numbers are converted into simple assembly language commands—that is, three-letter mnemonics that stand for very elementary machine language instructions. The first one in this listing, LDA, stands for load accumulator. You can find a longer description of assembly language and mnemonics in the *Apple II Reference Manual.*

Type control-C to get back to Applesoft, and then let's play a little with this routine. It works like this. You have to poke a frequency into memory location 6 and a duration into location 7. Then you jump to the machine language subroutine with a *call 770.* Try this in direct mode:

```
POKE 6,200: POKE 7,200: CALL 770
```

You should hear a short, low tone. Try changing the values you poke into locations 6 and 7 and see how this alters the note. Then you might want to experiment with loops. You can easily get all sorts of interesting sound effects. Here are a few examples:

```
FOR X = 3 TO 155 STEP 4: POKE 6,X: POKE 7,10: CALL 770: NEXT

FOR X = 1 TO 5: POKE 6,90: POKE 7,120: CALL 770: POKE 6,180:
POKE 7,120: CALL 770: NEXT

FOR X = 1 TO 2: FOR Y = 200 TO 50 STEP - 10: FOR Z = Y + 10 TO
Y - 10 STEP - 3: POKE 6,Z: POKE 7,10: CALL 770: NEXT :
NEXT : NEXT

FOR X = 1 TO 15: POKE 6,70 + RND (1) * 50: POKE 7,10: CALL 770:
NEXT
```

You've probably got the hang of it by now. You may find that a few good sound effects liven up even the most serious piece of programming.

Speaking of which, let's talk about a routine for you to work on over the next month. Most of us, at some point, are required to memorize large quantities of material. These may be definitions of terms or words in a foreign language. One of the most effective ways of learning such items is by using flash cards. Each flash card has a word or phrase on one side and the definition or translation on the other. You put the cards in a pile and run through them. If you correctly define the word or phrase, you put that particular card aside. If not, you put it at the back of the pile. This way you can study the more difficult ones and put the others aside.

We can write a flash card program on the computer allowing us to enter phrases and their definitions and then present them to us, in random order, until we get them right. Such a program will require three major sections—one to permit entry of phrases and definitions (and to edit them later in case we got them wrong), one to permit us to save the lists to disk (or to retrieve them), and one to test us with the list. In short, a master menu of functions might look similar to that of the following flash card program:

```
Edit Functions:
    A(dd words
    R(emove words
    E(dit words
Disk Functions:
    L(oad from disk
    S(ave to disk
Study Functions:
    V(iew list
    F(lash cards
```

Writing this program will give you ample opportunity to review your use of arrays and of string manipulation commands. When you start writing the program, start with the parts that will function on their own so that you can test them as you go. In other words, first write the main menu portion (the part that displays your options and then sends you off to the various menu items). Then start with the add-words section, which you will need in order to load your arrays with strings. At this point, jump down to the view-list option, which is a very simple part but one that will allow you to check and see if everything is functioning as it should.

Good luck. Enjoy. Come back next month, and we'll compare notes.

DON'T MISS APPLEFEST FOR 1983 THE WORLD'S LARGEST EXPOSITION EXCLUSIVELY FOR APPLE OWNERS

Each show features hundreds of exhibits of the newest, state-of-the-art products for the Apple. You can see and try out software for every conceivable application—from arcade games to investment programs, music to machine language, teaching systems to accounting packages, word processors to graphics processors. You can sample hundreds of different peripherals, including printers, hard disks, modems, memory cards, video displays and synthesizers, plus accessories, publications and invaluable support services.

Applefest is **the** place to view the most technologically advanced products for the Apple.

At Applefest you can try out and compare hundreds of products in an exciting, information-filled environment. You can learn more in two days than you could in months of visiting computer stores and reading trade journals.

And, best of all, everything on display at Applefest is for sale at special show prices, so you can save hundreds—even thousands—of dollars by making your purchases at the show.

This year a whole new conference program is being introduced to Applefests nationwide. The program will show you how to squeeze absolutely the most power, versatility and usefulness out of your Apple.

Seminars and workshops will teach you the ins and outs of buying software intelligently, using spreadsheet and database programs, putting Apples to work in classrooms and using the Apple as a management tool. You'll learn about new programming languages, important applications for telecommunications, exciting ways to use graphics and more.

No matter what you do (or want to do) with your Apple, the Applefest seminars and workshops will help you do it better.

Software Spotlights will provide an in-depth, understandable look at hundreds of different software packages. Each Spotlight will cover the features, capabilities and limitations of a group of packages, to help you find the software that's best suited to your applications. Experts will be on hand to answer all your questions.

So plan on attending Applefest for 1983—the biggest and best Apple-user show ever. It'll be a mind-expanding experience for both you **and** your Apple.

Daily Registration Fees
Exhibits-only badges are $8 per day, and the Conference Program is $15 per day.

Special Pre-Registration Discount
If you plan to attend Applefest save now with advance registration. Three-day Exhibits and Conference badges are $48, you save $21. Three-day Exhibits only badges at $18, you save $6.

Additional Information
To receive more information about attending or exhibiting at Applefest, including the Conference, Seminar, Workshop and Panel Discussions Program, or information on local hotels call 617-739-2000 or 800-841-7000 (Boston).

Cosponsored by *Softalk* magazine.

Applefest/Anaheim:
Friday-Sunday, April 15-17, 1983
Anaheim Convention Center
10:30AM-5:30PM daily

Applefest/Boston:
Friday-Sunday, May 13-15, 1983
Bayside Exposition Center
10:30AM-5:30PM daily

Applefest/San Francisco:
Friday-Sunday, October 28-30, 1983
Moscone Center
10:30AM-5:30PM daily

Any Questions? Call 800-841-7000 (Boston)


9
10
11
12
13

HERBIE HANCOCK FUTURE MUSIC

BY DAVID HUNTER AND RON RENNELLS

Dust swirls around the interior of the sunlit tent. Portable air-conditioning units rattle and sputter like malfunctioning disk drives. Outside, the temperature is about a thousand degrees.

It's Saturday afternoon, September 4, 1982. A couple of hundred music enthusiasts and shade-seekers have jammed into the Us Festival's speaker tent to see and hear presentations by Chick Corea, Herbie Hancock, and Bob Moog. Looking cool as always, Herbie Hancock steps up to the front of the tent and gives a short, entertaining demonstration of the Fairlight digital, computer-controlled synthesizer.

As he talks about the Fairlight, the alphaSyntauri, and the Apple, Hancock's great enthusiasm for electronic music is apparent. Someone asks Hancock if this means good-bye to acoustic pianos and what we've had in the past.

To which Hancock replies, "No. We're adding to what we have. We're not giving up anything. My next album is acoustic." But what about the one after that?

Herbie Hancock, holding the Clavitar portable keyboard, and Bryan Bell, Hancock's computer maestro. Hancock is recording his new all-synthesizer album at his Beverly Hills home, the first time he's had that luxury.

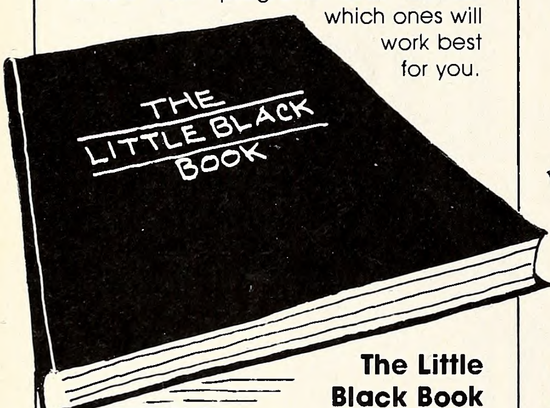
Software Any way you like it!

Sophisticated Software introduces 7 new software packages that enable you to maximize your Apple computer investment.

Simple to operate, yet sophisticated in performance, these new programs put your computer to work where you need it most.

Designed by a team of experts who have been developing software for the professional community for years, **Sophisticated Software** draws on that experience and makes this "miracle" software available for the first time at prices everyone can afford.

So take a look at **Sophisticated Software's** new programs and decide which ones will work best for you.

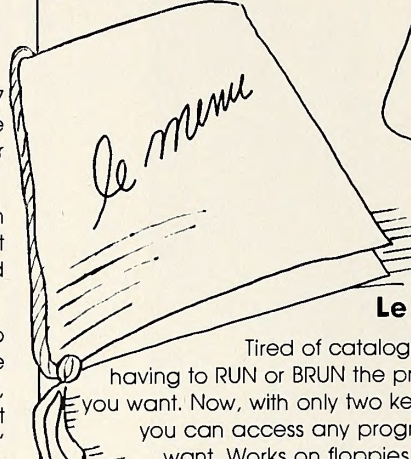


The Little Black Book

. . . the ultimate in sophistication and simplicity. Ideal for the professional, the man about town, or anyone who wants to maintain a "little black book" of names and notes electronically. A must for anyone who maintains client lists, but does so much more. You can add any number of notes to client files in any form you choose, all indexed by name and date. Simple to use and lightning fast, just turn it on, and go. Available for floppies or hard disk systems. . . **$249.00**

The Labeler

. . . is a program engineered to let you design and print your own custom labels. Great for making your own return address labels too. . **$19.95**

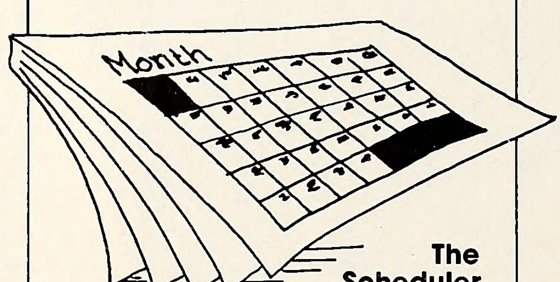


Le Menu

Tired of cataloging and having to RUN or BRUN the programs you want. Now, with only two keystrokes you can access any program you want. Works on floppies or hard disk systems. Le Menu makes it easy. . . . **$49.95**

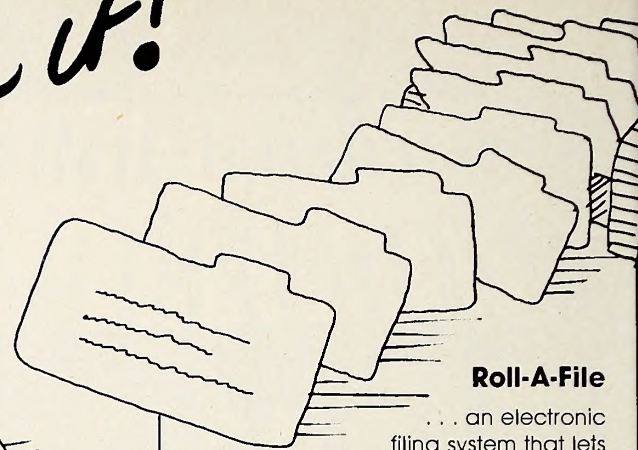
Micro-Mass Mailer

. . . sends your message with the impact of a telegram, in a customized impressive business-like form. The computer prints your message inside the envelope and addresses the outside all at the same time. This is an essential sales tool for anyone conducting business. **$129.00**



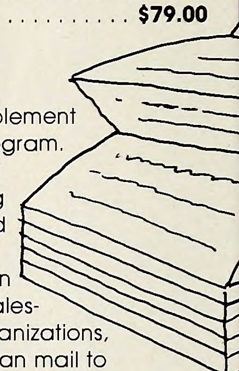
The Scheduler

Keep your life simple with the Scheduler. Don't forget those important dates and events. Schedule as far in advance as you like. Review your day's schedule at the touch of a few keys or print out a daily schedule. . . **$99.00**



Roll-A-File

. . . an electronic filing system that lets you store and retrieve client information by any criteria you select. Organize according to age, address, account number, date or whichever method best suits your needs. Roll-a-File is ideal for salesmen, businessmen, professionals . . . anyone. **$79.00**



Print-A-File

. . . a powerful supplement to the Roll-a-File program. It allows anything in your electronic filing system to be printed out using the same customized selection techniques. Now, salesmen, churches, organizations, and professionals can mail to selected groups of people at the touch of a button. **$99.00**

To Order

. . . by phone, call (213) 790-9052 and use your Visa or MasterCard. To order by mail send your check to: **Sophisticated Software**, 650 Foothill Blvd., La Canada, California 91011. California residents add 6½% state sales tax.

Sophisticated Software

Dealer Inquiries Invited

Two Places at the Same Time. The recording and performing career of Herbie Hancock has been compared to the dual personality of Dr. Jekyll and Mr. Hyde—with Jekyll being the acoustic jazz side and Hyde being the funky synthesizer side. Supreme jazz pianist, revolutionary electronic keyboardist, technical innovator, Hancock has influenced many contemporary musicians, but he consistently defies classification.

In his twenty-two-plus years as a professional musician, Hancock has played and recorded with George Benson, Paul Desmond, George Duke, Carlos Santana, Freddie Hubbard, Chaka Khan, Joni Mitchell, Raydio, Wayne Shorter, Keith Jarrett, Quincy Jones, Stevie Wonder, Oscar Peterson, Phil Woods, and Oliver Nelson, to name a few. Rooted firmly in jazz, Hancock has built up quite a reputation as an experimenter.

When you realize that this is the same cat who jammed with Miles Davis in the sixties, cooked with Chick Corea in the seventies, composed two film soundtracks, *Blow Up* and *Death Wish*, and recorded recent funk and disco albums like *Feets Don't Fail Me Now*, *Magic Windows*, and *Light Me Up*, it's clear that this reputation is well earned.

Hancock's latest album is *Quartet*, a two-record live performance in which he plays acoustic piano. Playing with Hancock are bassist Ron Carter, drummer Tony Williams, and trumpeter Wynton Marsalis. Joe Blum, in his review of *Quartet* in a recent issue of *Musician*, says of Herbie Hancock's performance: "Hancock emerges marvelously undamaged from any of his more commercial ventures."

A visit with Hancock in his garage-turned-recording studio—surrounded by a king's ransom worth of synthesizers, recording equipment, and computer hardware—brings the man and musician into focus. Centered inwardly, following his own mysterious muses, exploring realms that more commercially popular (and more pressured) musicians leave alone, Hancock is part artist, part devoted Buddhist, and part computer jockey.

Mr. Hands. Hancock is heavily into Apples. In fact, he's something of a fiend on the subject. Hancock's bold explorations in search of good-sounding synthesizers led him to discover Apples and the alphaSyntauri in the late seventies. Currently Hancock has more than a dozen different synthesizers, but he has only one brand of personal computer—Apple.

Hancock is just now finishing up a solo, all-synthesizer, all-Herbie record. A Saturday afternoon is the only time to catch him at his Beverly Hills home. In the bottom-floor room of the garage/studio are all the instruments—Oberheim, Clavitar, Clavitron, Yamaha, Rhodes, alphaSyntauri, Arp, Fairlight, Sequential Circuits, Moog, Emu, and Lynn. A half-million-dollar Trident mixing board (rented) takes up one whole wall. There is very little room to move around, but it's just enough.

Out of this fifteen-by-twenty-foot room comes music that sounds like it requires a whole orchestra to play. Drums, bass, brass, strings, keyboards, vocals—they're all there, electronically tinged but no less powerful than the real thing. And it all comes from one musician.

With the help of a custom-built computer patch bay, an Apple II, and the never-flagging assistance of sound technician and computer programmer Bryan Bell, Hancock has networked (so to speak) all his instruments together so they are playable from one keyboard. The Apple II acts as a terminal (they're currently transferring that function to the Apple III) to the system, giving Herbie easy access to the electronics of all the different instruments. This multiple-keyboard configuration allows Hancock to use any "voice" he chooses without having to reconfigure and reswitch the equipment and relocate his body each time he wants a different sound.

(A voice is the sound a synthesizer makes. Monophonic synthesizers have only one voice. If you hit two keys at once, only one note comes out. An eight-voice synthesizer like the alphaSyntauri allows you to press eight different keys at once, blending notes as if you were playing a piano. A completely polyphonic keyboard is one on which you can press all the keys at once and get all the notes.)

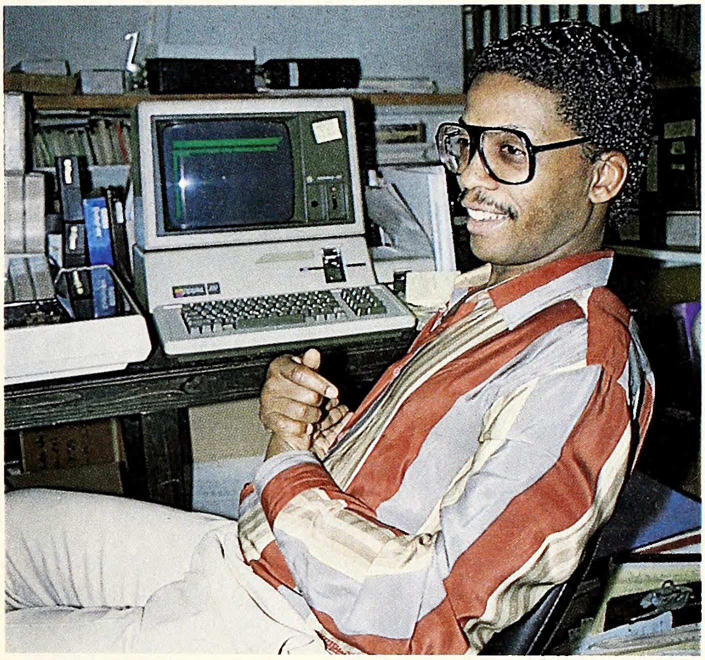
The Analog-to-Digital Blues. Hancock's cache of synthesizers ranges from the analog Minimoog and Arp 2600 to the digital Emu and Fairlight. Tying all these diverse machines together is a formidable task. And it's likely to continue to be a challenge as new synthesizers are introduced each year.

"Stevie Wonder has to have two of everything new that comes out," Hancock relates with a smile. "It's a game between us—to see who gets the newest things first."

Hancock has spent a lot of money over the years on his electronic music quest, and it's really just beginning to pay off. He's excited about his new album, especially because of the new instruments he's using—the Memorymoog, the Fairlight, the Rhodes Chroma, and the Yamaha GS-1. But it's taken hard work from Hancock and a host of others to get this far.

Bryan Bell has been working with Herbie Hancock since 1976. Bell's a guitarist, and he's a software and hardware engineer. Herbie's his boss, and Bell has worked wonders in pursuit of "Herbie's demand. His need. He'd ask me, 'How come it's not like this?' " And Bell would be off writing Z-80 code or hand-wiring patches.

"Herbie has got to have everything. The truth is," Bell says, "no one's ever made a synthesizer with the features of next year's product." And, Bell adds, there is no industry standard for combining synthesizers, drum machines, sequencers, clocks, and all the other attendant hardware in the way Hancock wants.



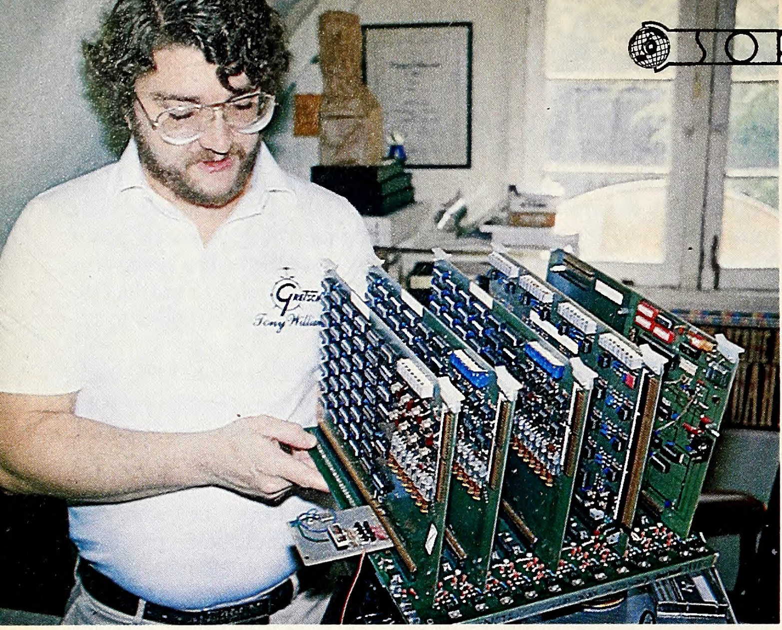
Hancock is an avid Apple user who uses the Apple III and *VisiCalc* to plan tours and keep track of recording budgets. He's developed his own programs and subscribes to the Source; and he'll be getting a Lisa later this year.

There are some signs of standardization, with the MIDI multisynthesizer controller and the Rhodes Chroma, which comes with a standard Apple interface. But in 1979, when Hancock and Bell first started to combine instruments so they could be played from one keyboard with the Apple as a terminal, it was a different story.

Monster. The first problem Bell tackled was modifying the Emu keyboard so that Hancock could control any one of his synthesizers through its sixteen channels of digital output. This involved increasing the control voltage so that the gate output reached a standard that the Oberheim synthesizer could use.

(In synthesizers, all you're ever dealing with is electricity. It comes from the wall and the synthesizer assigns a certain amount of voltage to each octave—usually one volt per octave. Hancock uses the Emu like an Apple with sixteen expansion slots, sending sequences of notes—voltage—to different synthesizers.)

In addition to modifying the Emu, Bell built a tuning device that increased or decreased the voltage coming from the Emu. This permitted each voice of all the synthesizers to be tuned independently to a standard, so any stock controller could be used. The assignment of channels to the various synthesizers was done by hardware patching. Analog engineer John Vieira of Waves Company helped Bell construct the tuning interface for this early configuration.



Bryan Bell shows off Hancock's custom-built, automated patch bay. This impressive piece of hardware acts as a digital switching matrix, configuring and reconfiguring the patches (interfaces between instruments) for each song.

Hancock and Bell convinced the company that makes the Emu to send them a prototype disk drive. They had quickly used up the Emu's sequencer memory (the sequencer stores note files—a whole song if you want—in the sequence originally played). Eventually, they got a Z-8000 sixteen-bit master computer for storing note sequences in virtual memory.

Hancock wanted to be able to control all this from a standard Applesoft menu. Bell, hardware and software designer Michael Larner, and a mysterious character known only as Universal Patchcord wrote a ton of code to make the various machines talk to each other. First they needed a sixteen-bit interface to allow the Apple to talk to the Z-80-based Emu and the Z-8000 master computer.

Custom FIFOs (first in, first out), basically programmable buffer chips, were required in order for the Z-8000 to handle the signals coming from the Emu. They waited "half a year" for them, says Bell. They also built an eight-bit, high-speed, bidirectional parallel port for the Z-8000. Now 6502 machine language could pass to the Z-8000, instructing it to send a sequence of notes to the master clock, which Bell and Vieira had built. (A clock controls the tempo of a sequence of notes. A master clock is the easiest way to make sure all the instruments play in synchronization. The voltage comes from the Emu, passes through the master clock, and ends up being played automatically on the desired keyboard.)

Feets Don't Fail Me Now. Hancock and Bell had to seek the services of electronic drum wizard Roger Linn himself to interface the Emu and his fancy drum machine, the LM-1. Then they interfaced the Clavitar and Clavitron portable keyboards to the Apple, with the help of Wayne Yentis, who designed those keyboards. During a concert, Hancock just presses a key on the Clavitar, which sends a signal to the Apple, which sends a signal to the Emu, which sends a sequence of notes to a particular synthesizer.

One last piece of equipment makes Hancock's system complete, more or less, till later in the year. The "automated patch bay" was custom-built by Bell and Universal Patchcord. (Patches are what allow one piece of equipment to interface with another. You use a patchcord to connect a monitor to your Apple. A particular song may require several different instruments in a unique setup. The automated patch bay handles the configuring and reconfiguring of patches for each song.)

"It's an ultraswank, thirty-two-channels-in, thirty-two-channels-out (in stereo), software-driven, digital switching matrix," says Bell. "To the Apple it looks like a printer port."

The voltage from the Emu goes through the automated patch bay and is directed to the proper instrument, in much the same way the MIDI and Garfield Electronics's Doctor Click Rhythm Controller synchronize several synthesizers at once.

The result of all this time and effort (four years and many long nights) is what Bell calls "the ultimate composition machine." As such, it is contributing significantly to Hancock's creative style.

"If I hear something in my head, I just play it on the Emu keyboard, which stores it in memory. Then I save it on floppies so I can always play it back if I want it later," Hancock says. "If I want to make a change in the bass tracks, I just tell the program on the Apple to shift to the bass instrument. I change it, then type run, and then it plays it, so there's never any need to write anything down. Notation is so cumbersome."

Hancock might use the Oberheim eight-voice for cellos, the Arp 2600 for flutes, the Rhodes Chroma for the bass sounds, and so on. Each of the different synthesizers has its particular strengths and weaknesses. Variety is the name of the game. Hancock is especially fond of the Chroma, which is touch-sensitive (the harder you press a key, the louder the sound). He uses the Emu as the main keyboard because it is a digital controller for the other instruments.

New Perspective. Hancock felt the need for a master clock for controlling the tempo of many different instruments at once and for a central switching device like the automated patch bay long before anything resembling these devices was commercially available. Now the music industry is starting to wise up.

The next piece of equipment that Hancock needs for his system is still to be delivered. That's a Lisa. The Lisa will replace the Z-8000 computer and "is very much part of the picture," according to Bell.

Eventually programs will be written so that Hancock can use the mouse and icon-based software to move from instrument to instrument. The Lisa will combine the easy-access terminal function previously performed by the Apple II and the Apple III with the mass storage and high-speed processing of the Z-8000.

At the moment, Hancock is pursuing a fascination for light pens. The Fairlight system comes with a light pen, used to draw wave patterns. (Remember that sound is voltage that comes into the synthesizer as a wave. Synthesizers normally come with oscillators for changing the wave shape. With a light pen and a digital synthesizer like the Fairlight, you have access to a virtually unlimited range of sounds.)

Light Me Up. "The Fairlight light pen is like a big crayon," Hancock says. He's currently enthralled with the Gibson light pen. "I needed this one cat [Steve Gibson] for a project. It turns out the guy is a synthesizer freak. He knew Pat Gleeson and John Vieira. What he's doing is fantastic."

During the past three years, Hancock has become an avid user of the Apple II and, more recently, the Apple III. He's spent many fine hours working with Neil Konzen's *Program Line Editor* and many long hours battling with program listings in computer magazines.

"I enter the programs and learn from the experience. If it doesn't run, I debug it. But then is it my error or is it in the material? If it's in the program listing and the debugging doesn't work, I don't know. . . . You feel like you've got brain damage."

Hancock is on the leading edge of the electronic music phenomenon. He's enthusiastic about computer technology and possesses an active, inquisitive mind. He's also willing to learn about new technology and is not afraid to experiment.

Hancock's Apple III is set up on the upper floor of the garage-turned-studio. At ease, talking at the computer like it's got a personality, Hancock demonstrates his practical, business uses of the Apple III. That's right, *VisiCalc*.

Hancock is basically a small business (his secretary actually answers the phone "Herbie Hancock!"). Hancock has an accountant, a manager, an agency, and characters like Bryan Bell hanging around. Given such a small operation, Hancock, with a little help from the Apple, is able to keep very close tabs on the business. Hancock has developed his own *VisiCalc* templates for keeping track of tour and record budgets. Sometimes, like on his last tour, knowing so much only heightens the misery.

"Five big concerts fell through, canceled because of unemployment, I guess, and the economy," Hancock recalls. "First it looked like we'd lost $30,000." Hancock used *VisiCalc* to plug in up-to-the-minute figures in his tour template and knew the bad news long before his accountant did.

"I imagined what the accountant would say. 'We got troubles!' Then we were losing a whole lot more than that. It blew my mind."

Free Form. "Lots of people are in music to make a profit," Hancock says. "They're after a big gross and are real serious about it."

One gets the impression that making lots of money is not always Herbie's goal in music or in life. But it takes no small amount of funds to accumulate and experiment with all that fancy equipment. Hancock is gearing up for a tour to support the new album he's working on.

Hancock believes that other bands are in a similar situation of having to tour to make money and needing to watch costs carefully. He envisions everybody using computers and sharing information on "cheap hotels and good all-night restaurants," among other things.

Hancock is just plain enthusiastic about the Apple and his ability to be in control of his business. He'd like to see more standardization in favor of the Apple. "Look at the Chroma from Rhodes," Hancock says. "They're owned by CBS and they went with Apple. The Apple's reliability makes it perfect for taking on the road."

Hancock cites Earth, Wind, and Fire as a band that has found the Apple invaluable in several ways. The band has an impressive synthesizer setup and Bell worked its 1982 U.S. tour. The group also uses the Apple for business and for automating the pyrotechnics in its act.

Hancock has taken the Apple along on previous tours, but his next one will be the first for the automated patch bay and the Z-8000 master computer. It should be quite an adventure for Hancock and Bell.

"My dream is to be able to walk into the concert hall with my music already playing," Hancock muses. "I'd just sit down and join in."

As it is, with the automated patch bay, Herbie will simply press a key on the Apple's keyboard before each song and all of the patches will occur automatically. Hancock uses a vocoder (which analyzes voice characteristics and uses them to control an input sound—like a synthesizer—and filters it to sound like your voice) for most of his vocals, and this too will be patched through the system. What is normally a hassle—physically repatching instruments for each song—Hancock and Bell won't have to worry about.

"When we go on the road we have tons of backup equipment," says Bell. "Spare Apples, spare disk drives, even an extra vocoder."

Master Telecommunicator. At home or on the road, Hancock is a frequent user of the Source. He sees big things in telecommunications. If you can send sequences of notes from one synthesizer to another, why can't you send them all the way across the country?

"You know, as long as we had the same equipment, I could send sounds to Stevie Wonder by modem. Eventually you'd have a library of sounds." Herbie's getting way out there now. "You could also have concerts or jam sessions by modem with musicians in Florida and New York." Stay tuned.

Bell is pleased with the support he's gotten from Apple Computer. "Apple is laying the groundwork for a personal computer revolution. Getting microcomputers into the hands of someone like Herbie really makes sense."

Bell will no doubt have his hands full when Hancock's Lisa arrives. He does not call himself a "kill" programmer. "I'm bad at 6502 and Z-80 machine language. I try to design the concepts and depend on Michael Larner for the machine codes."

Thrust. They say that computers are the first thing that has really excited Herbie Hancock since he discovered jazz at age seventeen.

Some people might still be mad at him for moving away from acoustic piano, even if it's only temporary, but Hancock's an innovator and he sees electronic music as a supreme challenge.

"We're not robots. Take a drum machine. There the technology is taking us away from slavery.

"People say to me that I'm putting orchestras out of work and taking away jobs. You know, it is possible for a piano player to learn to play a synthesizer." Look at Hancock.

But not right now. He's busy finishing up his album, the first he's recorded at his home. It's scheduled for release later this year. If you like funky pop and dance music, with generous portions of jazz thrown in for spice, then Herbie's the man with the music.

Plug 3,000 new applications into your Apple.®

THE CP/M Card™ plugs CP/M Plus™ into your Apple. The CP/M Card gives you the option of running your Apple II with the speed and capability of a professional Z-80 system with CP/M®-compatible software. You plug in the CP/M Card. Then choose CP/M or your standard Apple software at your option.

Plug into a big, new world of software.
The CP/M Card gives you instant access to the world's largest selection of microcomputer software—more than 3,000 CP/M-compatible applications, languages, and programming utilities. So, you, too can use professional business programs such as WordStar,® SuperCalc,™ Condor,™ and other high-performance software from Day One. Yet, you still have access to your present library of Apple software.

Plug into incredible performance.
Together, the ultra-fast CP/M Card and CP/M Plus run applications up to 300% faster than your Apple system! The CP/M Card is the only Apple II performance package that offers the speed and efficiency of CP/M Plus.

A plug about quality.
The CP/M Card was designed and built by Digital Research, the creators of CP/M, and Advanced Logic Systems, the most respected manufacturer of Apple performance products. So you know the CP/M Card is the most perfectly integrated Apple performance package you can buy.

Why just keep plugging along? The CP/M Card provides everything you need—including 64K of on-board memory, CP/M Plus, CBASIC,® GSX™-80 and full documentation—for just $399.

Now available through the CP/M library.

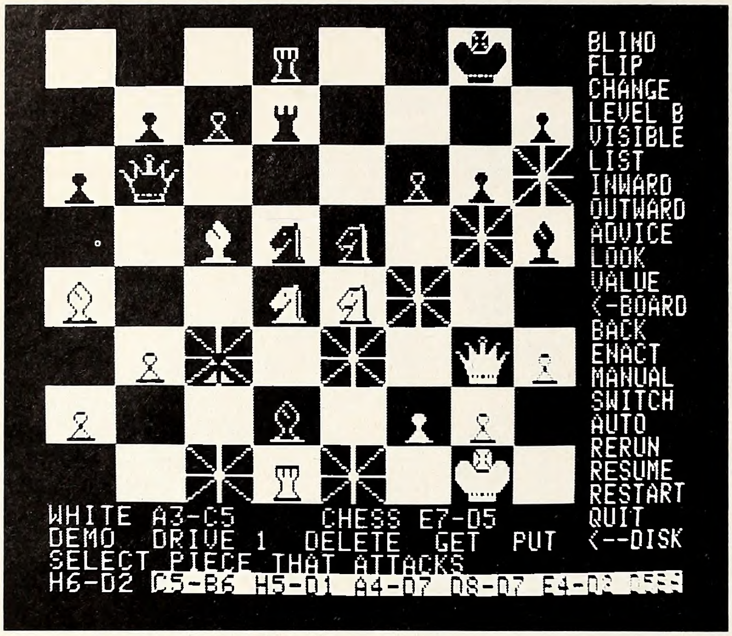
See your local microcomputer dealer today. Or contact Advanced Logic Systems, 1195 East Arques Ave., Sunnyvale, CA 94086 (800) 538-8177. (In California (408) 730-0306.)

Advanced Logic Systems

The CP/M Card for your Apple II.

Also available for the Apple IIe.

Explore the Frontiers of Intelligence



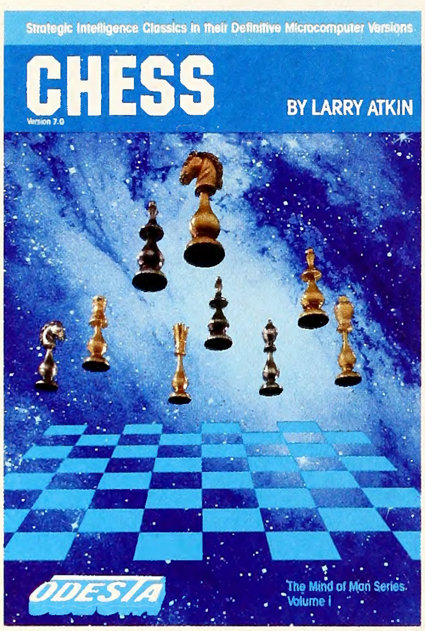
- ◀ Variations of blind-fold play—camouflaged or invisible pieces
- ◀ Invert board to play black on bottom
- ◀ Change pieces on board during game, or set up position
- ◀ Change between 15 levels of play, plus postal and mate-finder modes
- ◀ Show move that Chess is thinking about
- ◀ List played moves for each side
- ◀ Lines of force in: attacks and defenses on a square
- ◀ Lines of force out: squares attacked and defended
- ◀ Chess suggests a move
- ◀ Show moves Chess thinks you will make, and its responses
- ◀ Evaluation of a position
- ◀ Return to board or switch to command menu
- ◀ Take back a move (repeatable)
- ◀ Play move suggested by look-ahead search
- ◀ Chess plays neither side
- ◀ Switch sides
- ◀ Chess plays against itself—one level against another
- ◀ Replay through most advanced position
- ◀ Skip to most advanced position
- ◀ Start new game
- ◀ Leave program
- ◀ Save, get, and delete games to and from disk

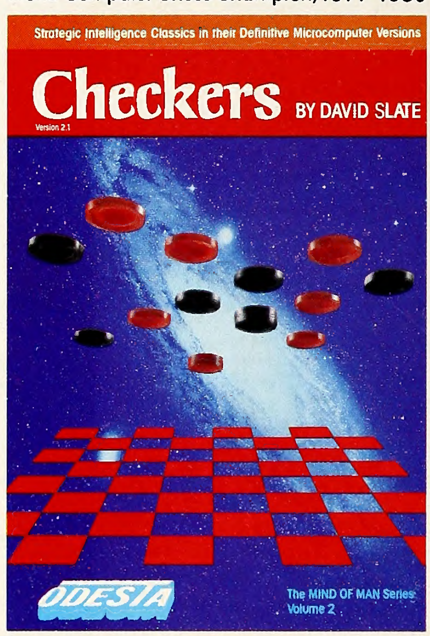
All features self-documented; all choices cursor-controlled
Screen shows "outward" and "look" features being used

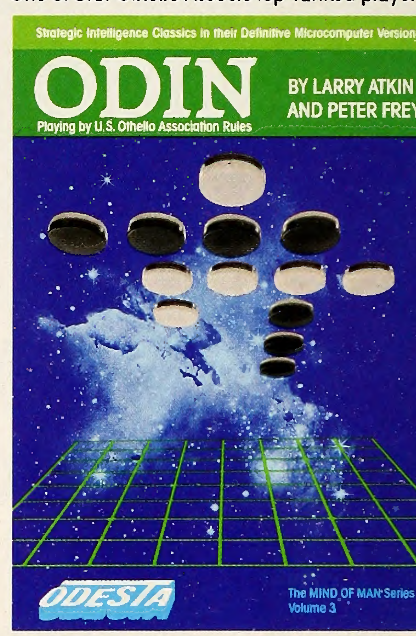
THE PEOPLE BEHIND THE PROGRAMS:

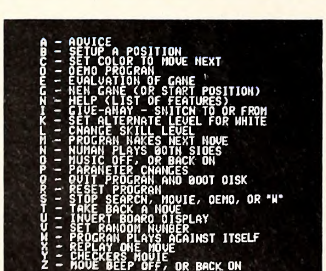
Larry Atkin & David Slate: Authors of the Northwestern University Chess 4.7 program—World Computer Chess Champion, 1977-1980

Peter Frey: Northwestern University professor
Editor: **Chess Skill in Man and Machine**
One of U.S. Othello Assoc.'s top-ranked players

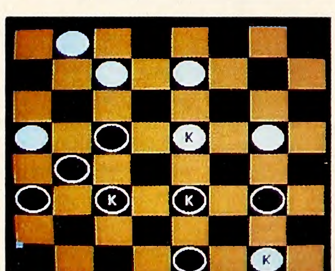




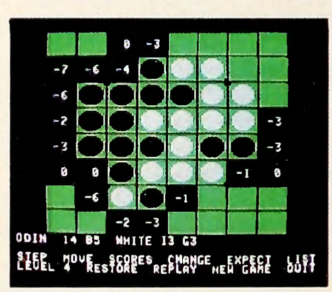




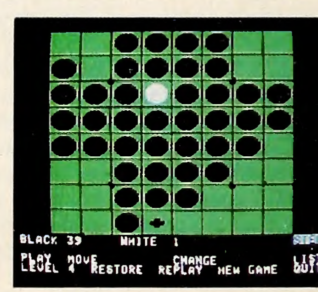
Checkers' features



Black to move and win
(From Checkers documentation)



"Scores" feature in Odin



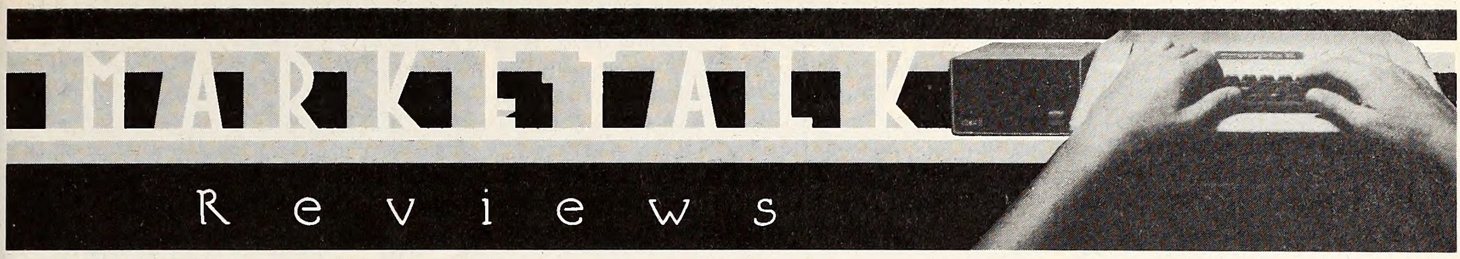
A clue to the secret of Odin:
Black is destined to lose.

ODESTA
930 Pitner
Evanston, IL 60202
(U.S.A.)

Chess: $69.95
Checkers: $49.95
Odin: $49.95

See your local software dealer, or order (Mastercard or Visa):
800-323-5423
(in Illinois, call 312-328-7101)

For **Apple II, Apple II Plus 48K** disk systems, and **Atari 48K** disk systems. Odin is also available for **TRS-80 Model 1 & 3 32K** disk systems.



Unless otherwise noted, all products can be assumed to run on either Apple II, with 48K, ROM Applesoft, and one disk drive. The requirement for ROM Applesoft can be met by RAM Applesoft in a language card. Many Apple II programs will run on the Apple III in the emulator mode.

The Composer's Assistant. By Kentyn Reynolds. Two technofantasies keep recurring to musicians. The first is that it will someday be possible to produce music by brain waves alone, without having to play an instrument of any sort; the second is that it will be possible to derive a score in conventional musical notation directly from a musical performance, without having to go through the arduous task of writing the music out by hand. The brain-music fantasy is doomed to remain imaginary, except perhaps for some specialized types of experimental music. The automatic score-generating problem, on the other hand, has been addressed by a host of computer music researchers. As a result, several score-writing programs now exist. Among the more recent of these is *The Composer's Assistant.* This software package enables music keyboard performances played on the alphaSyntauri computer music system to be printed out as conventionally notated, hi-res scores.

Syn-Dump, a printer initializer disk, derives the control codes that your printer needs to generate the graphics and embeds them in the score-writing software.

Next, the musician uses either the *Alpha Plus* or the *Metatrak* software to record a keyboard performance. Recording for score generation requires a somewhat different playing technique than recording for listening. Begin by selecting a tempo at which the music can be played accurately, then turn on the software metronome; this provides the time reference for the score-writing. The metronome clicks from the Apple's internal speaker. Using *Metatrak*, any mistakes can be corrected by "punching in." If the music score is longer than a thousand notes, it should be divided into two or more performance files.

Enter score-formatting information by using key and time signatures, metronome setting, and timing resolution. The program reads the performance file, then calculates each note's starting time and duration. These calculations are stored in an analysis file.

The graphics-generation routine works from the analysis file. You may look at the score on the monitor, one measure at a time. At this point, lyrics, titles, and performance instructions may be added to the score, directly from the Apple's keyboard. When all text is entered, the score is ready to be printed. The staves are printed vertically down the paper. All measures are the same length; notes and rests within a measure are precisely spaced in time. With an Epson MX-100 it takes slightly more than a minute to print out one measure of music. Graphic symbols are defined to a resolution of one matrix dot. Thus the printout, although slow, produces easy-to-read music.

The Composer's Assistant accurately measures note timing and duration and follows a complex set of rules and algorithms to format the score. However, the final products differ in several ways from traditional sheet music. Series of eighth and sixteenth notes are printed with separate stem flags, instead of one continuous bar. Notes of a given time value are always the same distance apart, whether or not there are accidentals (sharps and flats) between them. All note stems point up, and all note bodies are on the same side of the stem. The differences detract somewhat from the readability of some scores, especially scores of dense, rhythmically complex material. On the other hand, exact spacing of notes and rests and markings indicating all metronome beats are aids to readability that don't exist in conventional scores.

The Composer's Assistant is a useful tool for assistance in performing a job that is often complex and difficult. Like any other tool, using it looks simple at first glance but it takes practice to master. It doesn't correct playing mistakes, compensate for sloppy timing, or make similar musical decisions in your favor. Neither does it exercise artistic judgment to stretch and bend the rules under which it works. It transcribes accurately executed keyboard performances and spotlights exactly what is wrong with less-than-perfect playing. As such, *The Composer's Assistant* is a powerful addition to the alphaSyntauri's extensive musical repertoire. RH

The Composer's Assistant, by Kentyn Reynolds, Syntauri Corporation (3506 Waverley Street, Palo Alto, CA 94306; 415-494-1017). AlphaSyntauri keyboard and software, dot-matrix printer with graphics capability required. $295.

Kaleido-Sound. By Robert McNelly. Do you remember the days of speakers that came with color boxes? The kind that flashed different colors, depending on the pitch and intensity of the music being played? *Kaleido-Sound* is from the company that developed the Soundchaser Music System keyboard synthesizer for the Apple. *Kaleido-Sound* is the Apple's answer to those visual stimuli popular years ago.

By connecting the audio output of a stereo system to the Apple, various patterns are generated on the monitor screen in sync with the music being played. The visual effects range from lo-res kaleidoscopic patterns to graceful hi-res colored bubbles that pulsate in time with the sound. You can limit the colors to suit any theme and control the response time of the graphics to the sound.

The patterns vary according to the type of music being played. Classical music tends to produce very fluid patterns; modern music of the heavily amplified variety can produce sharp bursts of intense color.

Kaleido-Sound is a sensual enhancement of music. It engages more than your sense of hearing for enjoying music. When was the last time you watched Beethoven's Fifth? DA

Kaleido-Sound, by Robert McNelly, Passport Designs (116 North Cabrillo Way, Half Moon Bay, CA 94019; 415-726-0280). $39.95.

Music Maker. By Jim Baldridge. *Music Maker* makes it easy to transcribe music into song files that can be played through an Apple's built-in speaker. Although its capabilities are somewhat limited for playing multipart music, the final results sound very good indeed. You can realize much better sound quality by choosing the cassette port option and playing the music through a stereo system or tape recording it.

The program is comprised of several modules, all residing in memory during use. Thus you can play a piece you're composing at any time, even while entering or editing. As the piece is playing, a display shows the current note number and all the parameters of that note. Stepping through the notes one by one makes it easy to spot errors, even in long works or very fast passages.

The editing program works a lot like a word processor. The notes are numbered automatically and any changes due to insertions or deletions are reflected immediately. You can alter the tempo any time for the whole or for sections, but you must choose the key in advance. You can plan to have the piece change key in the middle; all subsequent note entries will have the appropriate accidentals added automatically.

Tempo note values can range from whole notes through 256th notes. The range of possible frequencies is equally impressive, beginning at low F on the bass clef and continuing to F sharp an octave above the treble clef. Triplets and even quintuplets are available, too. When you enter dotted notes in a row, you don't have to add the dot each time.

A well-written instruction manual contains a minicourse for those inexperienced in the use of musical terms. Also included are complete instructions for using song files from within programs in Applesoft, Integer Basic, or machine language. A special assembler enables the song to be run as a standalone module, independent of the *Music Maker* program.

Being able to repeat portions of a song is a super feature of *Music*

Maker. It can save hours of entering long, redundant passages, yet this feature is lacking on other, more expensive composing aids. Less useful, though no less impressive, is the Kaleidoscopic Maestro module, which creates a visual color display in step with the music.

The only drawback to this program is its method for creating multipart harmony. Chords must be approximated by the use of arpeggiation or other single-note patterns; they come across with a honky-tonk flavor. Several excellent demos on the disk illustrate the best use of this technique. EW

Music Maker, by Jim Baldridge, SubLogic (713 Edgebrook Drive, Champaign, IL 61820; 217-359-8482). $29.95.

High Rise. By Joe Calabrase. In the last couple of years, a marvelous educational game has been making the circuit of finer toy stores. The game involves a multitude of wooden blocks, each of a different wood, color, and shape. The shapes vary widely. Some are the regular cubes, cylinders, and rectangular blocks; others are very strange. One odd shape perches at an angle on the side of another odd-shaped piece.

The object of the game is to build a tower; the player who adds a piece that causes the tower to collapse loses. Because of the varying woods and shapes, the pieces have different weights and centers of gravity. In the process of constructing block towers, a child learns to differentiate subtle variations in shape and weight and gains a sense of proportion and balance.

High Rise from Micro Lab expands on and computerizes this delightful game. A warehouse worker attempts to build a rickety structure of oddly shaped boxes high enough that, when he climbs it, he can reach a ladder in the rafters. He selects one shape at a time from four self-feeding chutes. As he takes a shape out, a different shape slides into place. The man then carries his crate to a hydraulic spring and carefully positions it on the wide-spring platform, which can accommodate several different widths at once. When he's satisfied with the arrangement, he releases the spring, and the crates rise in a graceful arc and land in corresponding positions on the soaring edifice. When the highest point of the structure reaches the ladder, the man scrambles up the crates and the ladder to reach the next level.

The complexity of the game increases and a time limit for building the structure decreases with each level. The new crates on each level assume more and more bizarre shapes. Determining how to put the pieces together in such a way that the structure is stable stretches the ingenuity of the player.

High Rise is being marketed primarily as an educational shape-learning game for children and it's an excellent one; but the pace and scope of the game are well enough done to absorb most adults. People who enjoy solving challenging puzzles will find excitement and delight in each new level.

Hard on the heels of *Miner 2049er*, *High Rise* is another entry in the new high-quality line from Micro Fun. RRA

High Rise, by Joe Calabrase, Micro Fun/Micro Lab (2310 Skokie Valley Road, Highland Park, IL 60035; 312-433-7550). $39.95.

What's Where in the Apple. By William F. Luebbert. It is a fairly universal axiom that the more you learn about anything, the more you realize you don't know. If you're an Apple programmer, struggling to gain control of a rapidly changing, complex environment that didn't even exist a decade ago, you may find this to be a frustrating truth. Particularly when the reference books begin to procreate before your eyes and you can't remember where to find that reference to DOS bload address memory locations.

A well-organized reference like *What's Where in the Apple* can be just the thing when the amount of information needed to complete a major programming project becomes overwhelming. The book consists of three sections. The Atlas is a list of all the meaningful locations in the Apple II indexed in hexadecimal (with decimal equivalents) by memory location. There are thousands of such locations, including all the zero-page pointers, I/O hooks, and Monitor, DOS, Applesoft, and Integer routines.

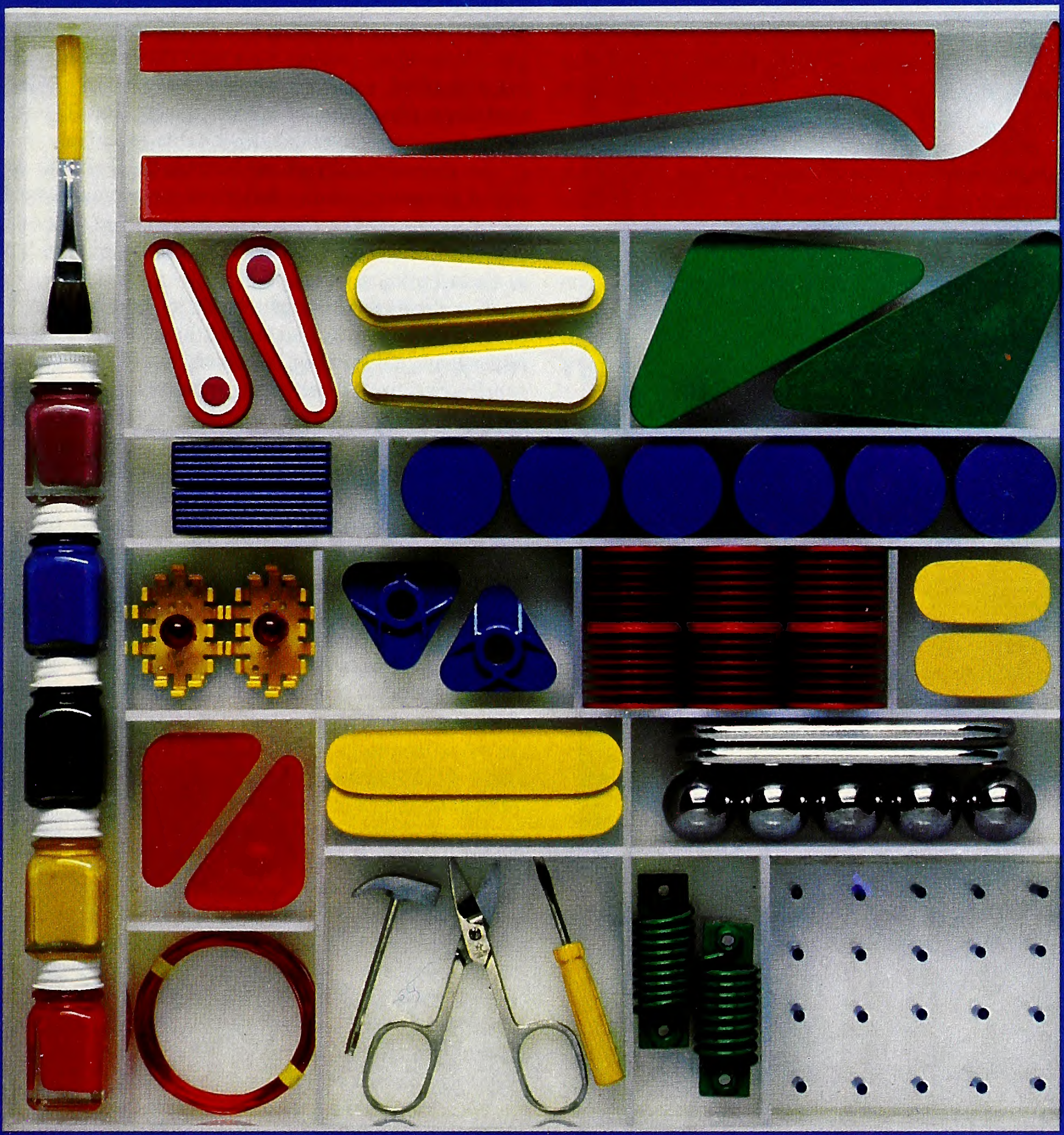
The Gazetteer section is the same list indexed by the standard name for the location or routine, such as the often used COUT and KEYIN and the more obscure BXSAV. The Atlas and Gazetteer sections are completely cross-referenced, and each item in the listings contains a brief description of the function of whatever is at that location. The Atlas section, by giving a meaning to all those anonymous hex numbers, is great for someone trying to understand an unfamiliar machine language disassembly. The Gazetteer is more useful for a programmer who knows the various internal locations and routines by name but doesn't know offhand where to find them.

An earlier version of the book consisted almost solely of these two sections, and in and of itself it was a powerful reference tool for the programmer. The most recent update, however, also contains twenty chapters that would be of further use to anyone attempting to write advanced applications. The early chapters contain comprehensive discussions of the various uses of peeks, pokes, and calls. This isn't a simple list of various specific commands and what they do, like the one in Appendix J of the Applesoft manual, but an encyclopedic treatment of advanced uses for such commands. For instance, the poke section includes a description of how to do a double poke—that is, to poke a sixteen-bit address into two consecutive bytes of memory.

The more advanced chapters discuss the hi-res memory layout, machine language addressing modes, how the Applesoft interpreter works, and useful but hard-to-find aspects of DOS and the Monitor. For those who bought the earlier edition of *What's Where*, the new section is available in a separate edition called *The Guide*.

Makers of Raster Blaster winner of the Softalk Award for most popular program of 1981.

The First Software Toy



BudgeCo announces The Pinball Construction Set, the first entertainment software that has the simplicity and freedom of interaction of a toy. You don't use this program—you play with it.

The Pinball Construction Set allows you to build your own video pinball games by providing a library of conventional (and unconventional) pinball pieces and a set of video tools. Use the video hand to put library pieces on the game board—as many as you want, where you want them.

Use the polygon tools to make borders and obstacles. Add game logic and scoring rules with the wiring kit. Create hi-res designs and logos using the BudgeCo magnifier and paintbrush. Change gravity, time, elasticity and bumper strength. Load and save designs to/from disk.

Available for the Apple II®
Suggested retail price $39.95

BudgeCo
428 Pala Ave.
Piedmont, CA
94611 415-658-8141

What's Where in the Apple is not for the beginner. It will by no means teach programming. What it will do for anyone who already programs—from the tenderfoot just beginning to understand how much there is to know to the Eagle scout who knows too much to keep track of it all—is provide a wealth of information in an accessible format.

What's Where in the Apple, by William F. Luebbert, Micro Ink (34 Chelmsford Street, Box 6502, Chelmsford, MA 01824; 617-256-5515). $24.95.

S.A.M. By Mark Barton. The acronym stands for Software Automatic Mouth, but it should be stated at the outset that to use *S.A.M.* with the Apple you need some hardware, which is included. This isn't a speech synthesizer chip but is instead a simple digital-to-analog converter, which accounts for the relatively low price of the product. *S.A.M.* seems to have been created first for the Atari, which has the necessary hardware built in.

The caveats aside, most of the credit for *S.A.M.* goes to software. By itself, a D-to-A converter can make some interesting noises, but speech requires some sophistication. *S.A.M.* does a good job of making the converter's buzzes, clicks, and tones sound like a voice.

There are two different machine language modules included, both of which can be called from Basic programs or from other machine language ones. One program takes phonetic input, for example, "kumpyuw3ter" for "computer." The second operates from normally spelled English text, interpreting it with standard pronunciation rules. The second program has some difficulty with oddly spelled words, of which there are many in English. The translating software also takes up about 6K of memory in addition to the 9K or so used by the synthesizer; so if you want to do anything serious, you have to get used to the phonetic alphabet.

In an old joke about Henry being spelled "Hen3ry," the three is silent. For *S.A.M.*, the three denotes the stress on the syllable. Any digit from one to eight after a syllable (in the phonetic spelling) will accent that syllable accordingly. This gives *S.A.M.* a nice human quality.

S.A.M. is highly recognizable in two senses—it's reasonably intelligent, and when you hear it speak you know it's *S.A.M.* and not some other speech synthesizer.

You see, *S.A.M.* has a foreign accent. There's a just-off-Ellis-Island quality to its voice that's hard to pin down to any specific ethnic group. One listener will detect a strong Russian accent, another will swear it's more like Swedish. Knock up the pitch and it's Count Dracula doing a Monty Python falsetto.

Perhaps *S.A.M.* is the key to the next generation of synthesized speech. In a few years, maybe we'll be able to control the accent.

S.A.M., by Mark Barton, Don't Ask Software (2265 Westwood Boulevard, Suite B-150, Los Angeles, CA 90034; 213-397-8811). $124.95.

Transtar 130. The wonderful thing about computers and their accessories in today's economy is that their prices tend to fall rather than rise. Daisy wheel printers have been slower than most peripherals to follow this trend; until recently any daisy wheel costing less than a thousand dollars has been of suspect quality. But Transtar's new printer, at $895, is an attractive and dependable machine.

At sixteen characters per second, the Transtar won't win any races with a dot-matrix printer, but it's in the normal range for daisy wheels. It offers ten or twelve characters per inch or proportional spacing, and standard print wheels are available in a number of typefaces for all three of these options.

Two features you might expect to find only on more costly printers are automatic paper load and settings for variable paper thickness. The paper load makes life a lot easier. Just set the paper up on the platen, touch a button, and the paper glides into place at the top of the page—or one, one and a half, or two inches down, depending on dip-switch settings. The paper thickness can be set from one to four sheets by moving a lever that has a very fine control over the distance from the print head to the platen.

The Transtar does lack two features that are common on dot-matrix printers and not unheard-of on daisy wheels. They are tractor feed and bidirectional printing. Unidirectional printing affects the speed and little else. Friction feed is usually okay at a daisy wheel's slower speed. Paper sometimes shifts one way or the other, but it usually isn't noticeable unless you're printing ten or twenty pages at a shot.

The sounds a printer makes are very telling, and usually they can influence the way you feel about the machine as much as all the other factors. The word *clunker* doesn't connote a cheap piece of equipment for nothing. The Transtar sounds clean. The even tapping of its hammer is almost restful next to the high-pitched wail some dot-matrix printers make. But the other noises are important too. The hum of carriage movement, the smooth, sliding sound of the line feed, and the distinct click when the print wheel fits into place give the satisfying feel of a precise piece of equipment.

The Transtar's performance bears this out. And it does graphics too. DD

Transtar 130, Transtar, a division of Omega Northwest (Box C-96975, Bellevue, WA 98009; 206-454-9250). $895.

Ken Uston's Professional Blackjack. By Jack V. Briner, Jr. Blackjack is unique among casino games in that it is the only game in which a skillful player can consistently make money. About twenty years ago, enterprising mathematicians used computers to determine just the best way to play every possible blackjack hand. To their amazement, they were able to devise a simple basic system that, theoretically, could beat the house. These avaricious savants headed straight for Vegas and made a substantial amount of money—just to prove their theory correct. Several of them made even more money by publishing books about their system.

The end result, after much confusion, was that the Vegas casinos changed their rules to restore their edge against basic system players. Unfortunately for the casino execs, the mathematicians, joined by hordes of computer programmers, had come up with an even more powerful strategy. They noticed that as cards were used in the play of hands the character of the deck changed; it would constantly swing between being favorable to the house and favorable to the player. For example, if all four aces have been played, then no one can get a blackjack. This is unfavorable to the player, who gets paid extra for them. By keeping track of the cards that have been played, it's possible for the knowledgeable player to determine when the deck is favorable and bet more in those situations, thus magnifying the effect of the favorable situation and gaining an edge on the house. Such players are known to casinos as "$#@$%!! card counters."

Ken Uston is one of the top ten blackjack players in the world; he's so feared by casinos that he is barred from playing in almost every casino in the world where card counting is illegal. In Atlantic City, where counting is legal, casinos have been known to pay him to put on demonstrations; they figure it's cheaper. Ken has written several books on blackjack and the blackjack life. The best of these is *Million Dollar Blackjack*, probably the most comprehensive collection of strategies and anecdotes about blackjack in existence.

Professional Blackjack attempts to teach three of the strategies found in the book. Which one you should learn depends on what your objectives are: If you want to have a good time and not lose too much money, you should learn a basic strategy. If you want to have a good time and not lose any money, you should learn a "simple plus/minus" card-counting strategy; and if you want to work at it and make some money, you should learn the "Uston Advanced Point Count" strategy.

A molded plastic case holds the disk and two manuals. The first manual tells you how to use the program; it's the one negative aspect of the package. This less-than-comprehensive manual is very skimpy and seems to have been written with the assumption that the program is so self-documenting that the manual would never have to be read. Well, the program is almost that self-documenting—but not quite. Still, the quality of the program more than compensates for the sketchy instructions. The program is menu-oriented and very easy to use and understand.

The second, much larger manual provides an introduction to blackjack and an explanation of the three strategies. Color-coded charts for each strategy are so neat that a smart cookie could do a brisk business just selling the charts in the bookstores.

The program itself is a delight to the blackjack student. At the gaming table, six other players can be controlled by the player or the computer. Computer players use one of the three strategies, and the program monitors your decisions and tells how correctly you played according to the strategy you've chosen to work on. Just for fun, you can also opt to use your own strategy and see how it does.

Stored on the disk are the rules for just about every casino in the

United States, with all their variations. This is one of the best features of the program; you get the feel of play at the Playboy in Atlantic City or Caesars in Vegas. And you can finely tune your playing strategy for the specific casino in which you'll be spending your vacation. If the rules change at the casino, the stored rules can be modified; you can even input an entirely different casino—crucial for all of us planning frequent weekends in Monte Carlo.

Although this package costs more than many other blackjack programs on the market, it's well worth the price to anyone seriously interested in learning how to play blackjack. The program will earn its keep in a few hours in the casino, if the player spends a few hours studying at home. See you at the tables! RW

Ken Uston's Professional Blackjack, by Jack V. Briner, Jr., Intelligent Statements (The Courtyard, Suite 21, Box 2602, Chapel Hill, NC 27514). $89.95.

Black Jack Strategy. By Norman J. Wazaney, Jr. *Black Jack Strategy* presents a basic, "no frills" approach to blackjack, mostly designed for the novice; this is its strongest feature. Neophytes who enjoy playing blackjack still don't like losing. The casinos, of course, love them; casual players provide the bulk of the casinos' revenues. But even the average player can strike back.

Black Jack Strategy comes with a basic strategy installed into the program. It allows development of new basic strategies, simulates those strategies over many hands of play, tutors players by presenting sample situations and monitoring responses, and allows players to experience the feel of head-to-head competition with casino dealers.

The program's only weakness is that it can teach you only a basic strategy; it doesn't provide for advanced techniques such as card counting. In fact, the author remarks that "card counting, even where legal, will soon cease to be a viable betting strategy." But right now, it is. There isn't a single casino in the world where you cannot make money counting cards (although some casinos offer much more favorable rules than others). But the author's philosophy makes most of this program of little use to serious players.

The program does what it says it will do for the beginning player, and it does it very well. The graphic displays are attractive and well thought-out. The manual is in a stand-up binder that's a real aid to learning. *Black Jack Strategy* is terrific for people who are casually interested in blackjack. RW

Black Jack Strategy, by Norman J. Wazaney, Jr., Soft Images (200 Route 17, Mahwah, NJ 07430; 201-529-1440). $69.95.

Logic Simulator/Logic Designer. By Andrew V. Thompson. One of the most tedious aspects of digital electronic design work is building and testing the prototype circuit. Even with a well-stocked parts drawer and a bench full of the latest test gear, there is simply no getting around the chore of assembling all of the components together and trying to make it work properly. If only there were some way to enter the proposed circuit into a computer and have it do the dirty work of checking signal flow and logic before plugging in the soldering iron!

The Logic Series from Spectrum Software does exactly that, in less time and with a whole lot less hassle. This package is comprised of two disks: one containing an interesting, fast-paced demo of what the programs can do, and the other containing the actual designer and simulator programs. The *Logic Designer* module allows drawing a circuit in schematic form by placing the various gates and flip-flops on a gridded pattern on the CRT. The appropriate inputs and outputs are then joined together in a connect-the-dots fashion to complete the final circuit. The completed circuit is automatically saved to disk before being analyzed with the *Logic Simulator* module.

The simulator is where the testing of the final circuit design takes place, and it allows an elaborate series of predefined binary signals to be applied to as many as eighteen input points, or nodes. The seventy output nodes to be monitored are also specified by the user. The resultant ones and zeros are listed on either the CRT screen or a printer.

One of the major features of the *Logic Simulator* module is that the screen drawing phase of entering a schematic may be bypassed entirely if desired. The arrangements of the digital building blocks may be entered directly instead. This has the decided advantage of speeding up the entire process considerably, since the computer doesn't need to see a drawing of the circuit to understand how it works.

When entering data this way, the designer must draw out the circuit by hand and assign node numbers to each input, output, and connection point. These are then entered into the computer as a series of tables prior to running a simulation. When a bug is found (there's always a bug!), the connection tables may be edited and the simulation run again.

Another powerful feature of the Logic Series is its ability to recognize user-defined subassemblies, or macros. When designing digital logic circuits, it is not uncommon to have many repetitive building blocks within one larger circuit. By predefining these macros, the user may recall a fairly elaborate array of interconnected gates with only one or two keystrokes. In fact, since the Logic Series does not support certain circuit elements such as counters, the only practical way to include them into the final design would be to organize the necessary gates and flip-flops ahead of time. Then call them up when needed. The same holds true for gates with more than three inputs; the user must first assemble the required device from smaller blocks.

Unquestionably, the Logic Series can be a timesaver, especially when attempting to analyze a large and complex circuit arrangement. For digital designs using a half dozen components, though, it may be faster just to build the whole thing the usual way. For one component, it is still rather a lengthy process. The grid system of drawing makes lining up the various inputs and outputs easier than it is with some other methods. The program is fairly disk-intensive, and a designer may become impatient waiting to get from one menu to another. Perhaps most exasperating is being asked time and time again, "Which drive will you be using for your data disk?" One would think that answering once should be enough.

With a capacity for one thousand gates, sixteen shift registers, sixteen separate user-defined macros, and more, the *Logic Simulator* is indeed a formidable piece of software. Spectrum is to be commended for successfully completing such a task. EW

Logic Simulator/Logic Designer, by Andrew V. Thompson, Spectrum Software (690 West Fremont Avenue, Sunnyvale, CA 94087; 408-738-4387). $250.

Pro Poker. By Jay Allen. *Pro Poker* is just that: professional. It's by far the best poker program for the Apple. *Pro Poker* allows you to play five-card draw, jacks or better, with wild card. Eight players can play at once; with less, computer opponents fill out the table, and these are some of the toughest computer players ever seen. They play hard and fast, and they even bluff correctly. If you don't play tight, you might even lose your shirt to these electronic cardsharps. The computer is even programmed to analyze your playing style in an attempt to "read" you.

Pro Poker, apart from being a fun game, is also a teaching tool. Jay Allen is a poker professional who has authored several books on gambling techniques. The game even has a kibitz mode that gives meaningful advice on what to do at any point and explains why it thinks its advice is correct. An option allows the player to watch the computer play all hands face-up, so you can see what other players are doing and why.

The program runs very fast and is self-prompting. It will deal an astonishing three hundred hi-res hands per hour.

A minor addition would enhance the game a little. *Pro Poker* plays silently. A bit of atmosphere—the slap of the cards and the clink of the chips—would be fun.

But *Pro Poker* is loaded with little touches that make it a dream to use—for instance, automatically organizing your hands to make them more readable! It's an excellent program that no budding cardsharp should be without. Only hope that you never wind up in a real game with its author! RW

Pro Poker, by Jay Allen, Quality Software (6660 Reseda Boulevard, Suite 105, Reseda, CA 91335; 213-344-6599). $39.95.

Old Ironsides. By Jack Rice and Richard Hefter. Does the lure of the high seas and the thrill of thundering ship-to-ship combat strike a chord within you? One of Xerox's new software publications, *Old Ironsides*, is out to satisfy that longing.

The arena is a square area of sea upon which the wonderfully drawn hi-res ships do battle. The wind is a big factor in maneuvering, and players need to learn to tack a three-masted ship. The program provides an excellent simulation of the difficulty of turning one of those large wooden ships.

When the ships are within range, they can fire broadsides at each other. There are six cannons on each side of the ships. Firing all six cannons at once is a full broadside. Subsequent damage reduces firepower, so players must maneuver to bring their ships' best sides to bear on the enemy. Likewise, the speed of a ship and its ability to turn swiftly changes as masts are lost during combat. If all three masts are blown away, the ship just drifts. A crafty commander can pretend to lie still in the water, only to come to life when a reckless foe goes in for the kill.

Ramming is possible and quite effective. As the ship rams, it can fire broadsides point blank into the enemy, doing devastating damage. Wary skippers are careful not to hit the enemy ships' powder magazines when their own ships are close in. If the magazine is hit, the ship blows up in a fiery hi-res display. The range of the explosion destroys both ships if they are too close together.

Fog plays a large role in this game, but it's a strange aspect of the simulation. Instead of providing fog banks on-screen to confuse the ships, the program declares that everything off-screen is fog. The players are provided with fragile compasses to help guide their ships back on-screen. If a ship stays too long in the fog, it is considered lost forever. This is the hardest playing aspect of the game to learn. The ships turn so ponderously that it's very difficult to keep them on-screen all the time. Once off the screen, even with the compass, determining the proper course on which to return is no small feat. Many a battle may be won just by forcing your opponent into the fog.

The superb packaging is part of the new Xerox look. The soft plastic-covered binder contains bonuses. There's a large poster of a famous painting depicting the 1812 naval engagement between the USS *Constitution* and the HMS *Java*, and there's a finely done Old Ironsides Log Book for chronicling the outcome of many encounters.

The game can be played by two people or by one person against the computer.

While *Old Ironsides* has a certain arcade feel to its play, it's basically a game of skill and strategy. Families looking for games to share would do well to look into this delightful program. RRA

Old Ironsides, by Jack Rice and Richard Hefter, Xerox Education Publications (245 Long Hill Road, Middletown, CT 06058; 203-347-7251). $39.95.

Apple Bunny. Apple Bunny is a neat, quick, little piece of hardware for which we have a neat, quick, little review.

Apple Bunny gives your Apple II or II Plus (no need for it on the IIe) the capability of auto repeat on all keys. The device's greatest virtue is that there isn't much to say about it. It can be installed in about a minute; it has eight little prongs that snap onto eight of the twenty-five pins that project from the keyboard just inside the Apple's cover. It doesn't conflict with the standard shift-key modification.

Another nice thing is that Apple Bunny doesn't lock you into having repeating keys. In some arcade games, auto repeat can be particularly unhealthy. On the Bunny, auto repeat toggles on and off to the simultaneous press of control and shift. Since it reads the keyboard pins directly, no strange software configurations can disable it, and control-shift is such a strange bird that it doesn't affect the operation of most programs. We did note that the control-shift commands in *ScreenWriter II* turned the Bunny off and on inadvertently.

When Apple Bunny is on, it doesn't keep you waiting. When you type normally at a key, it doesn't repeat, but hold the key down three-quarters of a second and it starts to repeat at about the speed the Apple repeat key would cause. The three-quarter-second wait is just about right—short enough to get you moving when you want to move, but long enough that it won't start to repeat when you're not expecting it.

You see? There really isn't much to talk about. Another thing about which there isn't much to say is the price. And that's nice too. DD

Apple Bunny, Accessory Products Company (4542 Palm Avenue, Yorba Linda, CA 92628; 714-970-2031). $24.95.

ORCA/M. By Mike Westerfield. In the grand scheme of things, you might wonder if there really is a need for another Apple assembler. Quite a few already exist, and assembly language programmers are not the largest subgroup of Apple owners that Hayden, or anyone else, might want to address at this point. So why *ORCA*?

ORCA is an assembler for programmers who have grand schemes. Sure, you might use it to impress your friends with your knowledge of assembly or to speed up your Basic programs with short machine language subroutines. It is no worse for that sort of thing, if no better, than any other assembler.

But *ORCA*'s true destiny is to assemble creations of the greatest sort: programs so big we don't even tend to think of them as programs but rather as part of the soul of the machine. Big programs referred to with big words are something we always tend to capitalize in our minds: Operating Systems, Interpreters, Compilers, Languages. Systems programming! Certainly *ORCA* can apply as well to other major undertakings, such as word processors, spreadsheets, databases, assemblers. *ORCA* would be a good assembler to get you started on the Great American Arcade Game (that eighties answer to the Great American Novel) you've been meaning to write.

ORCA has enough features to keep three utilities and a word processor happy as well as a macro assembler. The command level, which they call the Monitor (and with some justification), is the most extensive we've seen on any assembler. The wild-card character from the *Fid* program is available on most DOS commands in *ORCA*, so for instance you can ask for a catalog of all the file names on the disk that contain some common string (such as *mac*, to get a list of macro routines available). You can swap files, alphabetize the catalog of a disk, or restore deleted files, provided they haven't been overwritten. There's even a disk track and sector editing utility. We can't even list all the functions of the *ORCA* Monitor in the space of a review, but we offer these examples as evidence of Westerfield's attention to detail.

The editor is as extensive as the Monitor. In the first place, it frees the user from the bonds of line numbers. Most assemblers do line numbering automatically, but they still make you find out the number of a line you want to edit. *ORCA*'s editor has full scrolling so you can move the cursor to a line and change it. If you want to insert lines, you move the cursor to where you want the new lines to go and insert them. Then you don't have to list the program again to see how they look in context because the rest of the program is still there.

The editing commands take some time to learn. Most of them are somewhat analogous to word processing commands. Unfortunately, they are arranged alphabetically on the reference card instead of by command type. That's one minor flaw in the otherwise very useful and complete documentation.

The assembler itself does its work in two major steps and a couple of minor passes. The first step creates what is called relocatable code. This code isn't yet ready to brun, but it's more compact than the source code and has all the command references resolved. The second step is performed by what is called the link editor, which has the job of putting all the relocatable subroutines together at a given location in memory and making the whole thing ready to run. The value of such a system is that *ORCA* can construct huge chunks of object code. The source code for really large programs would never fit in memory at one time, so *ORCA* has this provision to link many source files together during the assembly process.

Another advantage of *ORCA* is the concept of local and global labels. If you're just writing small routines, that might seem unnecessary, but if you're doing anything large enough to have more than a few loops in it, you'll be glad that you can give them all the local name *loop* and not worry about the assembler getting confused.

The source files for *ORCA* are included on the system source disk. If you want to set *ORCA* up for a particular system configuration, you use it to reassemble itself. The whole process has a sort of chicken-and-egg quality but serves as an example to the truly ambitious. What other assembler could put together an entire system like that in one large operation? DD

ORCA/M, by Mike Westerfield, Hayden Software (600 Suffolk Street, Lowell, MA 01853; 617-937-0200). $99.95.

Microcomputer Circuit Analysis Program. By Andrew V. Thompson. Designing electronic circuits can be a lot of fun. With no more than a pad of paper, a pencil, and perhaps a pocket calculator, a creative engineer can amuse himself for hours on end trying to develop the optimum circuit to perform a particular task. Building and testing the design, however, is much less entertaining, as anyone who does this for a living could tell you.

Computer-assisted design programs have been with us for some time now. Unfortunately, they were available for use only on large and expensive computer systems. The *Microcomputer Circuit Analysis Program* is intended to bring this technology to Apple owners the world over. The package is comprised of several modules: a designer module, three analyzer modules, and, perhaps most important, a program that allows the user to establish parameters for the standard components that are used. All the important analog (as opposed to digital) components are represented here, such as op-amps, bipolar and MOS transistors, and transformers. Also, the waveforms to be applied to the imaginary circuits created by the program may range from a simple sinusoid to a programmable step input used for transient analysis.

The circuit to be evaluated by the analysis program is first drawn on the screen using the keyboard. Standard components are retrieved from a library of shape tables, positioned up, down, or sideways, and then connected together in the correct arrangement. Resistor and capacitor values may be specified at this time, though much of the power of a circuit simulator such as this is realized when component parameters are changed for each of several simulation tests. For each simulation, a graph of the response may be saved to disk, displayed on-screen, or sent to a Silentype printer.

The program can support up to thirty different component interconnection points or nodes, though the actual number of components that may be used is much greater. In fact, one would be hard put to use up all fifteen op-amps allowed, plus fifty resistors and fifty capacitors.

Circuits that have been entered into the program may be analyzed by three different methods: AC, DC, or transient. AC analysis is useful for determining the frequency or phase response of a circuit. A DC analysis will give the user an indication of steady state circuit conditions. Transient tests are essential whenever designing things such as active filters, ensuring freedom from ringing or other signs of circuit instability.

One of the most powerful features of the *Microcomputer Circuit Analysis Program*, an improvement over many of its predecessors, is its ability to recognize that a real diode has some amount of series resistance, that a real transistor always has at least some input capacitance, and that no op-amp can really have unlimited gain. A very wide range of independent parameters is assigned to all the components used by the program and may be changed at any time by the user.

Despite all its features, the analysis program has several shortcomings. Drawing the components onto the screen is a fairly tedious process, and there are certain situations where connecting lines, once drawn, cannot be removed without starting over again. The circuit being developed must be saved to disk constantly, as it is removed from memory every time a new option is selected. The program will run on an Apple II and IIe—and on an Apple II Plus that has two drives as well as 64K memory. Instead of using normal copy-protection methods, this program uses a "key" that plugs into the game paddle socket, precluding its use with an Apple III in emulation mode.

For many consultants and small design groups, this extensive program will be a real godsend. Large amounts of trial and error hours will be eliminated and deadlines might even be met. EW

Microcomputer Circuit Analysis Program, by Andrew V. Thompson, Spectrum Software (690 West Fremont Avenue, Sunnyvale, CA 94087; 408-738-4387). $475.

Hi-Res Architectural Design. By Don Fudge. Push a few buttons and watch a floor plan appear. Neat idea. Don Fudge apparently thought so and came up with a program for drawing fairly detailed architectural plans. The commands are simple and drawing a basic plan is relatively easy. These are choices of different width lines to indicate exterior and interior walls. There is also a menu of seventy-six different house plan shapes such as washbowls, toilets, doors, stairs, and circles. These are accessible with the touch of a key and can be easily rotated and moved around on the screen.

The program allows you to design, store, and print floor plans. Side views of structural drawings or sketches can also be viewed. Printing, though, does require additional screen-dump software if more than control character access is required on your printer for hi-res graphics.

Hi-Res Architectural Design has some convenient features. It provides on-the-move dimensional calculations, which appear at the bottom of the screen, telling the distance in decimal feet from the last point drawn to wherever the cursor is currently positioned. A toggle converts this measurement to feet-inches format. The dimension calculator also figures the angle that would be created by a line drawn from the last point to the location at which the cursor is currently positioned.

Other features include options to change the colors of lines drawn, to insert labels anywhere on the drawings, and to change the screen scale and the shapes scale independently. This makes it possible to create a wide range of drawings, from a whole floor plan to a detail of a room.

The scale of all the shapes may be changed, but, because the two scales are not related to each other in any meaningful way, shapes used in a room may or may not be proportionally correct. Because of this limitation, the program can't be used for final building plans.

As an easy-to-use program for sketching and trying out ideas, *Hi-Res Architectural Design*'s handy architectural style is ideal. SS

Hi-Res Architectural Design, by Don Fudge, Avant-Garde Creation (Box 30160, Eugene, OR 97403; 503-345-3043). $29.95.

Alien. By Hans van Halteren and David Kuijt. Avalon Hill's *Alien* is chillingly faithful to its ancestry—the movie and book of the same name. An alien, captive aboard a starship of the future (a research vessel), escapes its confines. The crew must recapture the alien, but time works against the hunters, for the creature is capable of metamorphoses into increasingly stronger and more deadly forms. Soon the hunters become the hunted. Adding to the problem are harmless lab animals released during the alien's escape; they're indistinguishable from the alien on the ship's sensors. The ship's orders are to bring alien life forms home for study; and around every corner may be a harmless creature or the stuff nightmares are made of.

Members of the crew have different capabilities and must be used skillfully to subdue the alien. Still, better a dead alien than a dead crew—but one of the crew is an android instructed to bring back the alien alive at all costs. He thwarts attempts to kill the alien.

Should all else fail, the ship may be set on self-destruct. The crew must attempt to escape to the ship's shuttles before the reactors overload and explode.

Detailed profiles of the crew, set forth in the manual, provide a rich backdrop for the imagination, as the game opens on a hi-res display of the starship and its thirty rooms and passageways. The graphic display of the starship is well done, but there are no close-ups of specific interior locations. The general locations of humans, animals, and aliens are displayed by the ship's sensors, along with a chart of specific crew locations and weapon status.

You must deploy and use the crew strategically to overcome the evolving alien. Because the game is randomly drawn anew with each play, you never know exactly what capabilities the alien has each time or which member of the crew is the android. The best strategy is merely to be prepared for the unexpected and to act quickly.

Game play is divided into movement and action phases for both crew and noncrew. Action points are expended for movement or activity such as weapon building or computer operation. Each member of the crew acts and moves separately. Using a game paddle to issue most commands brings freedom and convenience to the play of the game. No more mistyped keys and cryptic error messages.

If you liked the movie, you'll love the game. Part horror story and part science fiction, *Alien* combines elements of fantasy role-playing with the challenge of strategy and comes out smelling like total enjoyment. WHH

Alien, by Hans van Halteren and David Kuijt, Avalon Hill (4517 Harford Road, Baltimore, MD 21214; 301-254-5300). $39.95.

The Einstein Compiler. By Dennis S. Goodrow and Shmuel Einstein. There are two major measures of a good Basic compiler. They are how much space the compiled code takes up in memory and on disk and how fast that code runs. Secondary considerations are ease of use, flexibility, documentation, and price. *Einstein* is a winner in all categories.

Actually the speed difference between the various compilers available isn't all that significant. We ran benchmark tests on four other compilers when they first came out. The object of one of the tests was to determine how fast *Brian's Theme*, a hi-res demo on the DOS System Master, could spin out thirty patterns. The Applesoft interpreter took ten

minutes and thirty-three seconds. The fastest and slowest compiled versions took 5:34 and 6:00 respectively. A significant increase over Applesoft, but not a big difference between the two. *Einstein* came in one second behind the faster time.

Most compiler ads claim a speed increase from Applesoft to compiled code of two to twenty times. That upper-end figure is usually exaggerated. If you tried hard, you could probably create a program that was extremely inefficient in Applesoft, but wasn't nearly as inefficient in compiled form. Such a program may run twenty times faster after compilation. But people don't write programs that way, or they shouldn't. A speed increase of two to five times is a reasonable expectation.

It is in space considerations that *Einstein* shows a significant improvement over the competition. All compilers generate code that is longer than the original Applesoft program. It's the price you pay. Another of the programs we used to test those four compilers was *Little Brick Out* from the System Master. The most space-efficient of the compilers created a thirty-nine-sector compilation from the twenty-eight-sector program, but to run the compilation you also needed to bload a seventeen-sector run-time library. *Einstein* compiled the same program into forty-eight sectors, including its run-time library.

The space consideration may be the most important of all, because it may prevent a really large program from compiling at all. *Einstein* fairly consistently creates code that is less than twice the length of the original program, except where the original was very short.

The Einstein Compiler is remarkably easy to use. If you're willing to accept the default parameters, you don't have to tell it anything. It even automatically recognizes the use of hgr or hgr2 commands and reserves the appropriate space in memory. If your program uses pokes to turn on hi-res screens, you have to tell it what memory to reserve, which entails answering a series of questions on parameters.

With some compilers this is a traumatic experience. *Einstein*'s superb manual, written by Michael G. Samet, has a section that systematically describes the meaning of each parameter, under what conditions you might want to change any parameter, and how to do it. The manual, like the program, is so well done that you can expect to compile most programs successfully on the first attempt.

Three of the available options are compressing the code, disabling line trace, and disallowing unstructured for-next loops. Each of these options will make the final code faster and more compact. Nevertheless, there are good reasons for not using them all the time. The default configurations for the latter two favor the less efficient alternatives because programmers aren't perfect. Enabling line trace makes errors easier to find, and allowing unstructured loops makes the compiler more forgiving of spaghetti logic.

Because of the way *Einstein* works, it usually isn't necessary to add compiler directives in the form of rem statements to the source code. For more advanced applications, however, such directives are available. For instance, sometimes using a static storage area for string variables is preferable to allowing the program to store them dynamically and keep track of them with pointers the way Applesoft does. In this case a maximum length for strings must be set. If some strings should have a different maximum than others, they can be so designated in rem statements. By the same means, certain variables may be declared as global and passed on to other compiled programs that are either coresident or loaded and run by the first program.

It's hard to find fault with *The Einstein Compiler*. Some of the other compilers available are good, and if you already have one you're happy with, you probably wouldn't want to replace it. Different needs will certainly beget different choices of compilers, but if you're in the market for a compiler today, this newcomer deserves serious consideration. DD

The Einstein Compiler, by Dennis S. Goodrow and Shmuel Einstein, The Einstein Corporation (11340 West Olympic Boulevard, Los Angeles, CA 90064; 213-477-4539). $119.95.

Chargen V1.1 and **Tellitall**. By David P. Allen. Among the problems confronting video producers is that of producing titles for videotapes. Setting static title cards to be filmed is clumsy, time-consuming, and aesthetically stifling. The usual alternative is to use expensive postproduction special effects. Boston Media Consultants now provides a third choice.

Chargen V1.1 is an updated, licensed version of Darrell and Ron Aldrich's classic *Higher Text II. Chargen*, which is short for character generator, is designed for the professional video producer. Working with this excellent font generator, Allen made major modifications.

Terminology gets changed a bit. *Higher Text*'s Large Font becomes *Chargen*'s Little Print. Allen has devised a special way to expand that font four times. He calls this giant font Big Print. The program can also justify the fonts in a centered measure, something that has not been feasible before but is essential to video production work.

A unique option is the TV Producers Display Program. This option is used in combination with any video genlock board for the Apple II Plus. Such add-on boards make the video signal output from the Apple II Plus conform to U.S. standards (NTSC). This optional program allows the video producer to mesh the title fonts into the large video editing boards. This adds the titles in smoothly to live videotape.

For the home hobbyist, college studio, and video artist, a scaled-down version of the *Chargen V1.1* is available. The *Tellitall* also produces Big Print fonts and can justify the fonts. The price for the *Tellitall* is one-third that of its big brother.

Both programs come with ten fonts, and the company plans to release separate font disks in the future. Would-be game designers, who have been experimenting with *Arcade Machine* or *Pinball Construction Set*, may delight in enhancing their games with these specialized fonts. RRA

Chargen V1.1 and *Tellitall*, by David P. Allen, Boston Media Consultants (19 Damon Road, Scituate, MA 02066; 617-545-2696). *Chargen V1.1*, $125; *Tellitall*, $39.95.

Night Falls. By Bev. R. Haight. With the myriad of arcade-style games for the Apple, a new game has to be something special. *Night Falls* bids for the title with a unique combination of brain-twisting strategy and fast-paced action.

The game's theme is an old, familiar one. Alien invaders swoop down from the top of the screen against one valiant defender. It seems that hordes of robot aliens have wiped out Earth's cities; now they're removing the nitrogen from the atmosphere with devices called Twilight Makers. Lucky for Earth, human scientists have constructed uninhabited decoy cities that glow with an eerie green hue, attracting the aliens, then destroying them with weapons called gravity guns.

When night falls, the only time they can attack, the aliens rain down X-bombs and death rays to destroy the city below. The city is randomly generated via a cosmic reactor core; should the aliens breach the reactor defenses, the emerald city explodes in colorful hi-res.

The trusty gravity cannon may materialize anywhere in the emerald city. Besides destroying alien spaceships, the cannon can steal energy from the aliens to sustain the city.

The alien mother ships are another matter. The mother ships drop multiple X-bombs and have to be shot on sight. And there's more. A sinister stellar vortex can utterly obliterate your city—and even the ordinary alien critters are sometimes invisible.

During daylight, the energy from your reactor core rebuilds the city above. Then night returns, with twinkling stars and deadly aliens!

Night Falls employs excellent hi-res graphics and imaginative sound effects. Attacking aliens come in all shapes, sizes, and colors. Compiled from Applesoft, the game is quick and challenging. On the higher of the nine skill levels, the invaders are aggressive and actively seek the reactor. And, as each day passes, the invaders grow smaller and harder to hit.

To succeed at *Night Falls*, players must strategically circumvent the devilish design of the author. Constant choices and priorities decisions call for advance planning.

In case some players become interested in modifying the game to suit their own needs—or to maintain their sanity—the original unprotected Applesoft version has been included on disk. The modified game can be compiled with any commercial compiler.

Night Falls may be played by one or two players. High scores are saved with the player's initials. WHH

Night Falls, by Bev. R. Haight, Omega Microware (222 South Riverside Plaza, Chicago, IL 60606; 312-648-4844). $29.95.

LAMP. Edited by Mort Wasserman. There must be thousands of people who bought *Visidex* for the sole purpose of organizing reference material out of periodicals. Since then, dozens of specialized database templates have been marketed. Each claims to have the easiest method of indexing printed material. The problem has always been that users have rarely found the time, or had the inclination, to enter the information piece by piece. *LAMP* comes to the rescue.

LAMP stands for Literature Analysis of Microcomputer Publications, representing the first library-quality index to microcomputer publications; the large hard-copy compendium will be published bimonthly. Currently more than one hundred twenty publications are indexed, and more are constantly being added. Even foreign publications are included.

The three major sections of *LAMP* are authors, subjects, and reviews. All technical articles, computer programs, monthly columns, reviews, and features are covered. Each section is completely cross-referenced with the other sections. The reviews include books and periodicals, educational courseware, hardware, software, and computer games.

The first volume, January—February 1983, contains more than three hundred pages of packed information. For December, a book, summarizing the whole year, is planned. Besides the regular publication, Soft Images is also making this index available on microfiche at a discounted rate.

As more information is thrust into the marketplace every day, the task of organizing this data becomes more crucial. Every library, school, publisher, journalist, and public relations firm will find this publication an invaluable reference source. Unlike the usual high price for such detailed services, *LAMP* is priced for general use and should become a widely used reference source. RRA

LAMP, Mort Wasserman, editor, Soft Images (200 Route 17, Mahwah, NJ 07430; 201-529-1440). Yearly subscription: $69.95 (book); $54.95 (microfiche).

Crime Stopper. By Daniel J. Kitchen and Barry Marx. In the beginning there was Sherlock Holmes, Nero Wolfe, the Thin Man, Charlie Chan, and Sam Spade. Now there's Al Clubs, super sleuth! The case: Rescue Cartier Blanche Sizemore from her kidnappers. Weaving through a web of clues, deceptions, and danger, you search for Cartier Blanche Sizemore, hoping to find her before midnight, for that is when time runs out.

Crime Stopper puts the player in the role of Al Clubs, private eye. This all-text adventure has a real flavor of the whodunit. The clues are many, often well disguised. The plot is well developed and the storyline is good. The technical quality of the game is a bit weak, and the vocabulary is skimpy but it serves the player well. There's no continuously displayed status line, traditional in Scott Adams games; whether this is good or bad is a point of some disagreement among adventurers.

A bug remains that makes a newspaper difficult to read: The first of a two-screen article scrolls by without stopping. You can get around it by saving just before you read the newspaper, then hitting reset as the text scrolls on. Restore to your save and go on.

Crime Stopper has many enjoyable features, notable among them the system, complete with map. The main method of transportation in *Crime Stopper* is the subway system. Getting around is fairly straightforward; the location descriptions are well done and colorful. Trying to find the clues is as much fun as solving the game itself. *Crime Stopper* is challenging; solving it is no piece of cake and provides many hours of play time.

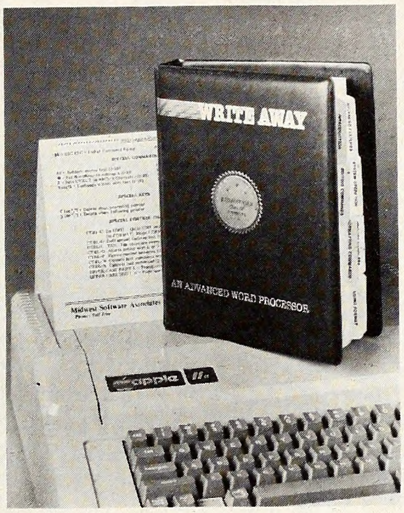
The game includes some limited sound effects, a ringing phone and a gun firing, which help to enhance the mental picture. Al Clubs seems to bear a striking resemblance to the TV sleuth Colombo, right down to the rumpled raincoat.

In comparison with the supersophisticated text adventures from Infocom, *Crime Stopper* is minor league; but compared with pregraphics Scott Adams adventures and their imitators, it's topnotch stuff.

Certainly, starved mystery buffs will find it a good meal; although it isn't the equal of *Deadline*, it does capture the flavor of sleuthing. Incidentally, timing is all. TR

Crime Stopper, by Daniel J. Kitchen and Barry Marx, Hayden Software (600 Suffolk Street, Lowell, MA 01854; 617-937-0200). $34.95.

NOW, USE ALL THE POWER OF THE APPLE *IIe* FOR FAST, EASY WORD PROCESSING



Write Away is a new, full-function word processing system designed specifically for the Apple *IIe*.

It will also run faster than any other word processor when using 48K or 64K Apple *II* systems.

Write Away is ideal for both the advanced professional or novice user. It is fast, versatile, powerful and easy to use . . . Just what you need to produce *any type of document*. Write Away has *all* of the advanced features you would expect from a professional word processing system.

Write Away is easy to learn and use . . . the manual is organized by chapters, and has a complete index as well as reference sections, in which each command is carefully explained with examples. Write away has screen tutorials, a "Help" screen and a handy stand-up command reference card. You'll also use single keystrokes for deleting, searching, inserting, cursor movement and other commands. Write Away allows conditional text ("If . . . Then" text selection).

It's fast . . . Write Away uses S&H's TDE ("the DOS enhancer") for fast system loading. Write Away programs are written in 6502 machine language using advanced programming techniques. As a result, commands are executed and files are loaded with amazing speed.

Extra features . . . Write Away uses standard Apple DOS text files for easy interfacing with popular programs . . . VisiCalc, Sensible Speller and more. It also has a DIF file translator and other utilities for creating a mailing list from DB Master and other data bases . . . mail merge system included.

Automatically utilizes RAM card, upper/lower case and recognizes Videx, Smarterm, Vision 80, Wizard 80, Sup 'R Term, Full View 80, Viewmax-80 and both Apple *IIe* 80-column boards. Write Away will work with any printer.

Awarded the recommended seal of approval by the Professional Software Programmers Association.

ONLY
$175

WRITE AWAY

Midwest Software Associates
P.O. Box 301 St. Ann, MO 63074

CALL TOLL FREE 1-800-835-2246/Ext. 467
(in Kansas) 1-800-362-2421/Ext. 467

Assertiveness Training. By Patrick X. Nidorf. Ever stood in a long line at the bank—only to find that the teller refuses to cash your check because the computer shows you don't have enough money in your account...even though you made a sufficient deposit the day before? Ever feel like throwing that check in the teller's face?

Or have you ever been driving home, after a hard day on the job, and felt the sudden urge to rear-end the guy that just cut in front of you, regardless of the damage to your car? (That'll show him!)

You might expect a program called *Assertiveness Training* to teach you how to recognize responses such as these as unproductive aggressiveness and avoid the unwanted consequences of acting on these impulses. The alternative: positive, self-assertive behavior. But the program's title is misleading. It will not really train you to be more assertive. It's really just a tool to help you be a bit more aware of yourself and think about how you might react when presented with different situations that could elicit aggressive behavior.

Like other packages from Psychological Psoftware, this one contains a multiple-choice questionnaire, the results of which make the claim to tell you something about yourself that you don't already know. This is by no means a professional diagnostic tool. The only assurance of this questionnaire's credibility or value is the word of the program author, who is listed simply as Dr. Patrick X. Nidorf on the title page, with no further references given. (M.D., Ph.D., or D.V.M.?)

The questions asked are all constructed along the same format, with the same four possible answers for each. The kind of responses you are asked to make are pretty polarized, ranging from "always" to "never." Another alternative response you can make is "sometimes." Unless you're an extreme case, the latter will be the most common response to such questions as "Do you find it difficult to make decisions?" The evaluation of your character, based on your answers to the questionnaire, remains very general, simplified and well within the bounds of commonsense advice. Other sections of the program give advice on how to act more assertively, both verbally and nonverbally.

In a section headed "Special Problems," a very reasonable analysis is given on how best to cope with no-win situations. Ways to try to get in touch with the feelings you're having when this occurs, as well as to recognize the temptation to accept guilt wrongly placed upon you by another, are included in a valuable and thought-provoking discussion. A number of assertiveness tools are also pointed out, such as the use of role-playing techniques to practice being assertive. Giving positive reinforcement to yourself in an ongoing way is emphasized as an important aid in persevering in the development of assertiveness along the lines described.

As a stimulus for starting to think about one's assertiveness, passivity, or aggressiveness, this program fills the bill. The first step in making any changes to oneself is awareness. Consider this program as the possible beginning of a longer journey. TC

Assertiveness Training, by Patrick X. Nidorf, Psychological Psoftware (4757 Sun Valley Road, Del Mar, CA 92014). $29.95.

Diversi-DOS. By Bill Basham. If you want to speed up your disk access, there are quite a few fast DOS packages. They're all pretty much the same speed: about two or three times as fast as Apple's DOS. If you want to know which one to buy, don't bother comparing speeds. Ask instead, is it easy to use? Is it well documented? Is it protected?

That last question is more crucial than you might think. DOS 3.3 is the most copyable and copied program there is on the Apple, and Woz knew what he was doing when he made it that way. A DOS you can't copy easily is an impediment to the use of your computer. In the name of copy protection, some fast DOS programs remove the init command, forcing you to use the master disk to format a new disk.

Bill Basham was not so shortsighted. Not only did he leave the init command intact, he also provides a utility to copy *Diversi-DOS* onto a DOS 3.3 disk without erasing the programs on it.

The instructions for *Diversi-DOS* are on the master disk instead of in a book. They can be printed out or read on the screen—remarkably complete for on-disk documentation, even including some well-written technical goodies about customizing *Diversi-DOS* for various uses.

Diversi-DOS offers two useful side shows to go with the main attraction. The DOS mover allows you to put *Diversi-DOS* onto a RAM card. The other program sets up the RAM card as a print buffer and also allows the Apple to buffer keyboard input.

Diversi-DOS's copyability is a major facet of DSR's distribution scheme. The company encourages you to copy the disk and give it to friends and requests that the recipients of the disk send DSR checks if they intend to keep them. Of course, *Diversi-DOS* is also available through normal channels. Low overhead for the company keeps the price down, which should keep people honest. It could be the best try-before-you-buy policy in the industry. DD

Diversi-DOS, by Bill Basham, Diversified Software Research (5848 Crampton Court, Rockford, IL 61111; 815-877-1343). $30.

It's the Pits. By C. Anthony Ray. Pits are the problem, but they're not where this game is.

An eat-'em-up with several fresh twists, *It's the Pits* is the first game from Sagebrush Software. Whatever else the game may have or lack, it has outstanding graphics and animation, especially for first-time authors and publishers. Critters pass over the background as well as in front of and behind each other with no color bleed-through. Everything's neat, clean, colorful, clever, and quick.

Well, maybe not quite quick enough. Although the game would be near impossible to play at high speed, somehow the player still wishes it could go a little faster. A lot of the reasons for the minimum speed are apparently deliberate.

What strange sort of game would require all these paradoxical-sounding qualities? Consider this: As player, you control a grimpet, a small, two-legged, big-eyed (and that's all) critter who absolutely loves to eat plums; and plums abound in this imaginary world. The grimpet has only to run between the maze walls picking up plums. Only trouble is, there are no maze walls. In Grimpetland, the walls must have been uprooted; in their place, and thus forming the maze, are deep fiery pits. Woe to the grimpet who falls in a pit.

To wirlybats, grimpi are far more interesting than plums. Wirlybats fly above the maze watching for grimpi to assume just the right position for the bats to sweep down and capture them by their little heads, carrying them off for heaven knows what mischief. The wirlybats follow distinct patterns, bouncing off walls like *Pong* balls; their main menace actually is one of distraction—and only the utmost concentration avoids the hellish pits.

Grimpi have recourse against the wirlybats, anyway. If in succeeding rounds a grimpet can collect a helmet and a candlestick, it becomes immune to the wirlybats' attacks. With burning candle settled upon the crown of the helmet, the grimpet can turn wirlybats into mere puffs of smoke; then its only problem is staying out of the pits.

At this point, it sounds like a cinch. Hang around as soon as you enter a level, do in the wirlybats (for good strong points), and you're home free; just wind through the maze and go on to the next.

Ha. One misstep is all it takes. One corner anticipated an instant too soon, one misjudgment of the elasticized outside maze wall, one change of mind or pace and it's the pits. And, when a grimpet goes down, helmet and candle go with it; the next of your relay team of three can finish off the current maze in freedom, but it'll have to fend off the wirlybats without protection throughout the next maze and most of the one after that—regardless of how difficult—to regain helmet and candle.

It's the Pits is a welcome harbinger of fine games to expect from a fresh new publisher. The graphics, the animation, the style, and the spirit are here; all that's to come is a tad more gaming value. Let's keep an eye on Sagebrush Software. MCT

It's the Pits, by C. Anthony Ray, Sagebrush Software (39 Carriage Place, Urbana, IL 61801; 217-328-5916). $29.95.

ORIGIN SYSTEMS

Presents a new dimension.

Software without equal.

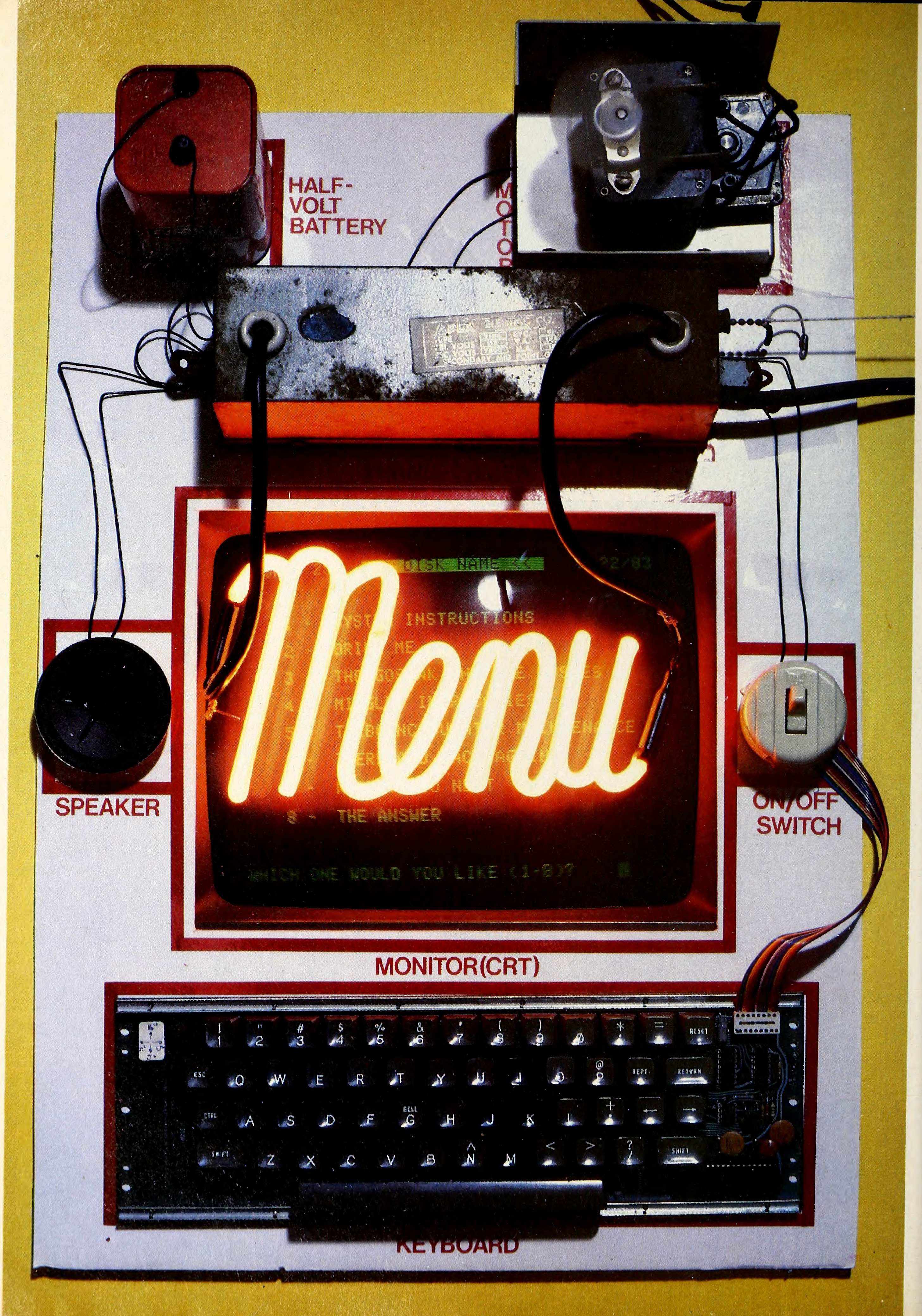
The ultimate in design, execution, and value.

Taking your machine where you want to go.

We invite you to join us.

ORIGIN SYSTEMS 1902 Back Bay Court, Houston, Texas 77058; (713) 333-4716

Dealer and Author Inquiries Welcome.


HALF-
VOLT
BATTERY
MOTOR
Menu
DISK NAME
INSTRUCTIONS
8 - THE ANSWER
WHICH ONE WOULD YOU LIKE (1-8)?
SPEAKER
ON/OFF
SWITCH
MONITOR(CRT)
KEYBOARD

IIND GRADE CHATS

A Custom Menu Generator

by JOCK ROOT

"Whatever did I put on this disk?"

The catalog command, in Applesoft, is good enough when you're still working with a disk; but when all the programs run properly, and you're ready to put the disk in your utility file—or sell it—you may want something different. Something easier to use, and prettier, more user-friendly and professional-looking, as they say in the business. Or something that will allow turnkey operation, so that the user need not know Applesoft to run the system.

Here is a menu program that can be made part of the hello program (or called by it) and thus will come up whenever the disk is booted. It can list as many as nine programs, and you can select any one with a single keystroke. The menu program can handle Applesoft and machine language programs, randomly mixed. Also, the program name offered to the user can be different from the name the program is actually filed under. This means the user can ask for *Dice Game* in order to get *Dice Matrix 4.3*, or some such.

All of this is accomplished by means of a two-dimensional string array, which is loaded with paired data statements. If that sounds complicated, don't worry; it's simply a way of keeping two lists side by side. When the user asks for an item from one list, the program sends out a call to DOS for the matching item, and it is loaded and run (or brun, whichever is appropriate).

The nice thing about keeping the lists of names this way is that it's easy to make changes. You simply add, or remove, one line of the program. Each data line is of the form

```
2000 DATA "USER NAME","DISK FILE NAME"
```

in which the user name is that which the person using your program asks for, and the disk file name is what's on the disk. Thus, if you want to add another program to the disk, you can include it in the table of contents simply by adding a data line to the menu program. The program automatically adjusts the display format to compensate for the number of data lines included, so you don't have to worry about that when you add or remove a program.

Arrays and Indexes. Before we plunge into the program itself, let's do a quick review of array logic (if you know about this already, skip ahead).

An array variable is not actually one variable; rather, it's a set of several related variables that have numbered tags so you (and the computer) can tell them apart. Arrays are useful when you want to do something several times and keep track of the several different results. For example, if you wanted to record the hour of sunrise for every day of the week, you could use an array variable like Sunrise(day), which is actually the set of variables Sunrise(Monday), Sunrise(Tuesday), and so on. Of course, the computer will store the variables as SU(1), SU(2), and so on to save space, but we can go along with that as long as it keeps track of the values.

Array variables tend to hang out with loops a lot. When you do something in a repeating cycle, and each pass through the loop produces a value (or requires one), an array variable is a handy way to keep the various values in order.

So far, we've been discussing one-dimensional array variables—the identifying tag has only one number on it. Sometimes more than one dimension will be needed; for example, if you numbered the weeks of the year from 1 to 52, you could keep track of years of sunrises with the array Sunrise(year, week, day).

Of course, you have to tell the computer what arrangement you're using, so it will know (for instance) whether a particular value belongs with the last day of this week or the first day of next. That's called *dimensioning* the variable, and is done (surprise!) with a dimension statement, *dim* for short. For ten years of sunrises, you could use *dim Sunrise(10,52,7)*. Or, if you wanted an arrangement by months instead of weeks, four weeks to the month, you could use *dim Sunrise(10,12,4,7)*.

Oops! That wouldn't work. Can you see why? It has to do with that four-weeks-to-the-month assumption. In fact, even Sunrise (year, week, day) would need an adjustment every year, since there are not *exactly* fifty-two weeks in a year. You have to watch out for that sort of thing with arrays.

To summarize, an array variable represents a set of values, distinguished by numbered tags of the form (*x*), where *x* is a number. This number is usually called the *index* of the variable. A one-dimensional variable will have one index, a two-dimensional one will have two indexes, a three-dimensional one will have three, and so on.

It's very important, if you have a several-dimensional variable, to keep the dimensions (the indexes) in the right order. You can imagine some of the problems that might result from getting them mixed up. We'll have more to say about this later, when we talk about nested loops.

But that's enough about array variables for the moment. We're now ready to start taking the menu program apart.

The Program Begins. The purpose of the disk data section, beginning at line 10, is to identify the particular disk in use. This information will appear as a header line over the choices shown on the menu. These items are right at the front of the program, so they'll be easy to find when you want to change them for a different disk.

The items in the disk data section are self-explanatory, except for the strange spelling of DAET$; that's necessary because, if you spell it DATE$, the Apple will boggle at the reserved word, *at*, and translate the names as D AT E$.

In line 70, we skip to the other end of the program to pick up some more values. Why are they out there at the end? Several reasons, but one stands out:

Every time Applesoft does a gosub or a goto, it has to check each program line in order, starting with line 1, to find a match for the line

POWERTEXT. THE ONLY FULLY AUTOMATIC WORD PROCESSING SYSTEM FOR YOUR APPLE II.

Now even hunt-and-peck typists produce perfectly formatted letters... memos... presentations... scripts... automatically.

NOW... A COMPLETE SELECTION OF HIGH-PERFORMANCE FEATURES IN TODAY'S MOST ADVANCED WORD PROCESSING TOOL.

The professionals who use PowerText tell us it's changed their whole concept of word processing.

Because this is the system that goes far beyond the write/edit/print/file capabilities of other word processors.

PowerText produces documents that look exactly as you want them to look... *automatically*. From everyday business letters to the most complicated presentations and reports you can devise.

You define the basic formats. PowerText stores them in its style files.

So when you write, you just bat out the words and never worry about where they go on the printed page.

That's PowerText's job.

You just tell PowerText, "I want a letter," and you get a letter...and envelope. With every element just where you want it to be.

Ask for a memo and you get a memo. Ask for a dramatic script and you get a dramatic script...

OK, that's one big PowerText *plus*: the automatic formatting capability that can save you untold time and effort.

But there's still more. A full selection of high-performance features to make your word processing faster, easier, more *automatic* than any system you've ever seen...

Total control editing.

Automatic indents and numbers for outlines, with system-assigned Roman and Arabic numerals, and alphanumerics.

A disk-based editor: file sizes limited only by disk space.

The system automatically maintains a backup copy of the file being edited.

User-definable function keys.

Up to 7 levels of "nested" editing to let you suspend editing one file and move to another.

"Paint mode" editing allows vertical typing for diagrams.

ALL THIS IS AUTOMATIC IN EVERY KIND OF DOCUMENT YOU WRITE.

- Margins
- Indents
- Spacing
- Title page
- Envelope, label
- Justification
- Centering
- Variable pitch
- Pagination
- Table of contents
- Headers and footers
- Footnote Numbers
- Intelligent page breaks

PLUS MANY OTHER VALUABLE FEATURES.

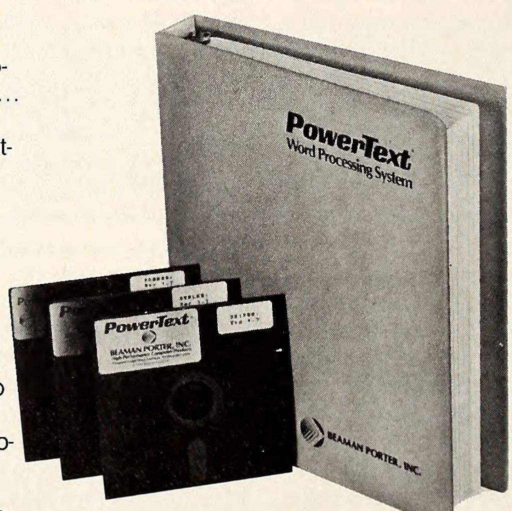
- Built-in form letter capability
- Boxed copy
- Print macros
- Up to 14 columns
- Boilerplate inclusion
- Optional word-by-word cursor move
- Vert. and horiz. border lines
- No limit to document length
- 132-character lines
- Superscripts, subscripts
- Column and line adjustment capabilities
- And much more.

NEW 10-LESSON TUTORIAL.

Our new, streamlined 10-lesson tutorial makes learning PowerText a snap. You'll master it quickly, as you discover how easy and straightforward it is to use. And your system also includes a complete reference manual.

FULL 5-YEAR WARRANTY.

PowerText does everything we say it will do. If it should ever fail to perform as specified, Beaman Porter, Inc. will fix it at no charge, anytime within 5 years after you've bought it.



PowerText

BEAMAN PORTER, INC.
High Performance Computer Products
Pleasant Ridge Rd. Dept. STA-5
Harrison, NY 10528 (914) 967-3504

SYSTEM REQUIREMENTS.

PowerText is a self-contained, bootable system, that supports both serial and parallel printer interfaces. You'll need your Apple II with 64K, 2 disk drives, and a printer.

COMPLETE APPLE II POWERTEXT SYSTEM SOFTWARE PLUS TUTORIAL AND MANUAL $299.

SPECIAL! PowerText *plus* easily-installed PowerCase chip for Apple II upper/lowercase and shift key capability, just $329.95.)

(PowerText for Apple II and III users who have Apple Pascal, $199.)

(PowerText is also available for IBM PC. Write for information.)

Tutorial/manual alone for any version: $25.

Available from selected dealers. Or order directly from Beaman Porter.

VISA and MasterCard accepted. NY State residents add appropriate sales tax.

BEAMAN PORTER, INC., DEPT. STA-5
Pleasant Ridge Road
Harrison, NY 10528

☐ Send _____ complete PowerText system(s) for Apple II @$299. ☐ With PowerCase upper/lowercase chip $329.95.
☐ I have Apple Pascal. Send _____ complete PowerText system(s) for ☐ Apple II or ☐ III @$199.
☐ Send tutorial/manual only, for __________ version @$25.
(NY State residents add appropriate sales tax)
My check or money order for $__________ enclosed.
☐ Or bill $__________ to my: ☐ MasterCard ☐ Visa
(MasterCard only list 4 digits above your name__________)
Card No.__________ Exp. Date__________
Name__________
Street__________
City, state, ZIP__________

number it's looking for. This takes time—the more lines Applesoft has to check, the more time it takes. Thus it makes sense to write your program with as few lines as possible between the front end and the busy parts.

The job we're doing now, loading variables—often called *initializing* the system—only has to be done once, at the beginning of the run; so it isn't a very busy part of the program. Therefore we put it at the end, where it won't slow the program down, and we get to it with a subroutine call.

In this case, it doesn't really matter—our program is so short that any slowing effect would be unnoticeable. But we have to do the initializing somewhere, so why not do it the "professional" way?

After initializing, we print the header. This is where all that disk data gets used.

Next we have to read in the choices from those paired data statements. It's loop and array time.

Filling the Array. What's the first thing you do in dealing with an array? Right, you dimension it—line 210. This is a string array, because we want to fill it with words rather than numbers; otherwise it behaves just like the sunrise array described earlier.

In case you're wondering what the use is of a dimension of 1, as we have here (how can it be an array, if there's only one value in it?), you should note a peculiarity of dimension numbering: it always starts at 0. Thus if you dimension a variable as V(1), it has room for two values, V(0) and V(1). We usually ignore this, using the values from 1 up and wasting the V(0) space; but, when you only need to store two values, you can save a third of the memory space you'd otherwise need by using that value.

So now we have a string variable, N$(9,1), which can hold nine pairs of strings (well, ten really), with each pair consisting of a 0 string and a 1 string. The 0 string is going to be the user name for the program, the name that is offered in the menu display; the 1 string of each pair will be the disk file name of the same program.

Now (in line 220) we have to deal with a side issue: what's this onerr goto about?

One of the classic problems with loops is how to get out of them. The familiar for-next loop has a built-in counter, and when the counter reaches a certain value, you exit. But suppose for some reason you have to exit in the middle? Suppose, for example, you run out of data before the counter is finished?

Normally, that counts as an error, and Applesoft provides an error message to tell you so. But suppose you meant for your program to run out of data? Suppose you wanted to use that data runout as a signal—not of an error condition but merely to tell the computer, "You've got all there is—now do something with it!"

That's one of the clever uses of the onerr goto statement: it does exactly that. It tells the computer, "This isn't really an error—just a signal that this process is finished, and you should start something else."

In this case, it means, "that's all the data you're gonna get; now do line 270."

That's how the program is able to adjust itself to any number of data statements (up to nine pairs) without being told.

We'll deal with what happens at line 270 in a little while, but there's something interesting happening right in front of us. Speak softly, friends, and watch your step: we are approaching a pair of nested loops!

I and J are the indexes (they're a couple of old favorites, among loop fans). Note that J does indeed start with zero, although wasteful I doesn't bother to.

Now the thing to watch out for with nested loops is this: you must never get the inside one mixed up with the outside one. The outside one starts first; then the inside one starts, *and ends*; finally, the outside one ends.

A for-next loop, of course, begins with the for and ends with the next. In our case, the outside loop is the one with the I index, since it begins first (line 230). That makes the J loop the inside one (beginning at line 240), which means it has to end first.

In other words, the good old Apple has to take the next J (if any are left) before it takes another I; the J loop has to end before the I loop can continue.

All of which explains why line 260 is *next J,I* instead of the more natural *next I,J*. And don't you forget it!

Now that we've got our housekeeping squared away, let's take a look at the purpose behind it all—the line that's inside of both loops, *read N$(I,J)*. This line reads the next available data statement into one of the eighteen spaces (that's nine pairs, or I times J) of our array. Each time around, one (or both) of the indexes will be different, so we'll eventually fill all eighteen spaces.

Unless, of course, we run out of data. That will generate an error signal, and . . . well, here we are at line 270!

NC is the number of choices offered—the number of pairs of data statements we read before we ran out of data. The reason it's one less than the I index, instead of equal to it, is that whatever forced the error break would have happened after the I counter was incremented for that cycle, so the counter would read one too many. Even a normal exit from the for-next loop leaves the index reading one high, because of the counter logic: "increment the counter and compare to the preset limit; if too high, exit."

And now the program knows how many choices there are.

Displaying the Choices. The next step (line 300) is for the program to offer those choices to you, the user. That's done with another for-next loop, but this one is simpler: we're displaying only one-half of the pair, so we need only one index this time. We'll use good old X.

Note that good old X appears in two places inside the loop. In line 330 it provides a line number for the display, and in 340 it selects the Xth pair of strings. The other dimension of N$(X,0) being zero, line 340 always prints the user name of the program.

Before we actually get into that loop, there's a sidestep (line 310) to adjust spacing. This subroutine is out toward the end, like the initialize routine, because it only needs to be called a few times during the run—twice, in fact, as we will see in a moment.

What it does is add one or more blank lines, depending on how many choices are to be offered: this will make the display better looking, which is nice, and easier to read, which is essential. Note that this subroutine (line 1200) uses the NC value we just determined.

So now we have a header line at the top of the screen, with an appropriate amount of blank space under it. Now we need to print the choices themselves, so roll the loop! And here again we use NC, this time for the upper limit (line 320). This will give us one display line (one pass through the "print a line" loop) for each choice.

Next (line 380) we call the spacing routine again, this time to get an even space between the choices.

Where the User Comes In. Finally, line 410 prints the prompt line—the line that tells you what to do next—and waits for a response (line 420).

Note the use of *get* rather than *input*. Input is fine when you want the Apple to expect an input of unknown length (for example, in entering a name), but when you know that only one character is coming, why make the user press return after it? Whenever you can be sure how many characters are coming in, the get command is much easier for the user to deal with.

When a response comes in from the keyboard, the program has to evaluate it: there are three possibilities.

The first is an escape loophole, useful mainly for testing the program. If the received character is a space, line 510 does a poke to cancel the onerr goto condition and restore normal error handling and then ends the run.

You should probably change this before turning the program over to a naive user: it's easy to hit the space bar by accident. To change it, type in some other character between the quotes, instead of a space. The asterisk (*) might be a good choice, since it couldn't input by accident (you need to press two keys, neither of them a number). Or you could use *if Q$ = CHR$(27) then...* in line 510 to have it respond to the escape key. Once you've got the program thoroughly debugged, if you'll want a secure turnkey menu so that the user can break out of the program (that is, escape into Applesoft) even by accident, then simply omit line 510 and remove the poke 216,0 from line 270. Then, anything users do will result in one of two things—running a stored program or getting another prompt.

Line 520 would take the input and return a number, if the user typed a number; if anything else was typed, it would return zero. Then line 530 would go to an error routine (we'll get to that later), which eventually would return to the prompt.

Line 540 does the same for an illegal number—one larger than the number of choices offered.

If it's a number, nonzero and not too big, then it's legal input—we will respond politely. Line 550 acknowledges the request by repeating it back to the screen.

Then there's a little housekeeping to be done. We're about to send out a call to DOS, asking for the program the user wants; and every call to DOS using control-D—known to its friends as D$—must be preceded by a return.

That's what the print in line 580 is for, but there's a snag here. If we happened to be on the bottom line of the display at the time—and we would be, with eight or nine choices—a print command would scroll the whole display, eating the header line and generally messing things up.

So line 560 calls −998, the address of an Applesoft routine that moves the cursor up one line. That gives us the space to print without scrolling.

And in the meantime, we begin to format the command to run the requested program.

Getting What You Want. The number you ask for when you request a program is known to the Apple as Q (determined in lines 420 and 520), so the disk file name we want is in N$(Q,1). Line 570 gets it and renames it N$ to simplify matters later.

And now we're almost home, unless something goes wrong.

Line 610 sets up an "errorless" jump in case DOS sends back a "file not found" message. It doesn't indicate a problem yet—we're asking for an Applesoft file the first time, and our target might be a binary file.

Line 620 is the DOS command itself. The purpose of the whole program is to generate this one line!

If we don't score with that one, we'll fall through to line 660, another onerr goto. If the program finds an error here, we'll be in trouble—it means that there was neither a binary nor an Applesoft program under the requested name; but we do *not* want to fall into Applesoft as a result, which is what a normal error break would cause.

What we want, instead, is to go back to the main prompt. We erase the message from line 550 by sending the cursor to the left margin and calling CLREOL (clear to end of line) at −868. A print command would do the same as the call, but since it also might scroll the display, we'd do it this way.

Then we print an explanation and go to the main error routine.

Mistaaak! The program uses a two-part error routine, with a customized "head" and a common "body"; that is, there are two different versions of the first part of the routine, one for each of the possible error conditions (illegal input or file not found). The second part is the same for both situations.

The head section of the routine prints a message to tell you what went wrong. Then the body section beeps to get your attention, waits for you to read the message, and sends you back to the prompt.

The illegal input head (line 800) is a little more complicated than the file not found head (line 700). We want to erase the prompt line but leave the input on the screen so you can see what was wrong with it. This is done with B$, a string of blanks (spaces) just long enough to wipe out the prompt string.

Then we move forward—from the line 700 head by means of a goto, from line 800 by simply continuing—to the body at line 900.

Here you see the advantage of subroutines. Most of the work of this program section is farmed out via subroutine calls. The only thing that's actually done here is to erase the prompt line again, this time with CLREOP (clear to end of page) at −958. This will erase the entire prompt area so we can start over.

And finally, at line 960, we send the program back to give another prompt.

Notice that this, too, is a kind of loop. It doesn't have a counter, like a for-next loop, but every time something goes wrong (bad input or a missing file), you'll get the beep and the wait subroutines and then go back to the prompt again.

That's why line 510, the escape loophole, is in the program. It provides a way out of that endless cycle, which can be very useful if you make any modifications to the program.

We have now finished the main sequence of the program. All that is left is a handful of subroutines, and the data statements themselves (they're out at the very end).

The subroutines begin at line 1000. In a very large program, the routines that are called most often should come first—it shortens the process of counting through the line numbers to find a match—but this program isn't big enough for it to matter.

The Delay Timer. Take a close look at the timer in line 1000. The logic of the timer is sort of folded in on itself, which makes it tricky to follow, but it also provides a lot of power in a small space. If you need a delay timer with a wide range of settings, you might want to use this pattern in your own programs.

Like many programming tricks, the delay timer is easiest to understand if you look at it backward. We'll start with lines 1060 and 1070, the main loop of the routine.

This is your basic for-next loop. Most timer routines have one of these in them, in some form. This one counts up to Time, which is a variable defined in the initialize routine.

Even at microelectronic speeds, counting takes time; the length of time is proportional to the number you count up to. Thus, changing the value of the Time variable is one way you have of controlling the length of the delay.

Now we'll go forward by backing up one step: consider line 1050, gosub 1060.

Huh? Whassat? We're at 1050, and we want to gosub 1060? !##*!????!!

Easy, easy, be cool—it's another programming trick. What's important about this trip is not where we're *going* to, but where we're *coming back* to. Got that? Good.

Line 1060 is the timer itself: every time you get sent there, you count through one cycle and return. *That's* clear enough.

Ah, but watch out for that innocent-looking return. It jumps all over the place! That's why the important thing about this pattern is where you go back to—we're going to get a lot of mileage out of that return statement!

Let's get back to line 1050: from here you go to 1060, count one cycle, and come back. Then you advance to the next line, as usual. Now you're at line 1060, the timer, so you count one cycle—that's two, now—and return; and this time, you return to the line that sent you to line 1050 in the first place.

Let's repeat that: when you're sent to line 1050 on a subroutine call, you go through the timer once on a subsubroutine, and then once more on a fall-through; then you go back to whatever line sent you. Thus, getting sent to line 1050 means doing two cycles and returning.

Suppose—we now advance another step backward—you get sent from line 1040? Why, you do two cycles and return to line 1040, of course.

Yes. And then you fall through into line 1050, and do two more cycles; and after that, you go back to the line that sent you to line 1040. Four cycles this time.

See the pattern? Sure you've got it, now? Okay, then you can advance back to line 1000 by yourself. Take it slow and you won't get dizzy.

The purpose of this arrangement is to give you another way of controlling the time delay. Depending on where you enter the routine (which line you gosub to, from 1010 to 1060), you can get from one to thirty-two cycles. Thus if you pick a large number for Time, so that one cycle takes one second, you can select a delay between one second and half a minute.

You can even fine-tune the routine to get the effect of an entry point between the entry points. For example, this one offers you a choice of sixteen or thirty-two cycles. Suppose you need twenty? Easy; you just call the routine twice, once with the sixteen-cycle entry (line 1020) and once with the four-cycle entry (line 1040). The times will add, and there you have your twenty cycles. In fact, that's how the delay is done here—lines 920 and 930.

By the way, despite the round-and-roundness of it, this multiple-entry pattern of alternating gosub and fall-through is *not* a loop. Remember, the whole purpose of this pattern is to be slightly different at each entry point; and the defining characteristic of a loop is that it's always the same, cycle after cycle.

If you plan to use this timer for several different delays in one program, note that the two control adjustments for the timer (selecting the entry point, and setting the value of Time) have different effects. Picking the entry point will get the delay for that particular call, but changing Time will affect every timer call in the program.

And now, having spent enough time on that, we'll return to the main program.

Last Bits and Pieces. Actually, there isn't much left of it that we haven't covered—only a couple of simple subroutines, and the data statements themselves. The Beep-Beep routine (line 1100) makes a double beep (CHR$(7) twice), then a pause (with the timer), then the double beep again.

Line 1910, in the initialize routine, deserves a mention. The reason B$ is split like that is to make it easy to copy with the cursor, in case you want to change the length. If it were in one piece, Applesoft would break it in two when listing it, without marking the break.

And finally we have the data statements themselves—the program's own reference file. The main thing to remember about these is the format: quotes around each name, and a comma between them. If you leave out any of that stuff, the program is likely to run two of the statements into one and mix up the data from that point on.

However, because of the quotes, you can use almost any characters you like in the name, except another quotation mark or a carriage return (and, of course, the disk file name has to follow DOS rules).

You may have noticed that all the user names in this program call the same disk file name, the program called *Sorry.* That was done for developmental purposes; the *Sorry* program gives you a blank screen with the message, "That program is temporarily out of service." Then it returns you to the *Disk Menu* program.

Hooking It Up. And that brings up the last question we have to deal with. How do you get the menu program in the first place?

There are two ways. The simplest is to store the *Disk Menu* program as the hello program. To do this, set it up before you initialize the disk, edit the disk data section appropriately, put in a blank disk, and type *init hello.*

A better way to make a classy turnkey system is to have a separate hello program, perhaps a fancy graphics display featuring the name of the disk (and, of course, *your* name); something like the title page of a book. Then that program calls the *Disk Menu,* just as the title page in a book is followed by a table of contents.

To do that, make the last line of your title page/hello program read as follows:

```
PRINT CHR$(4)"RUN DISK MENU"
```

assuming you call your version of this *Disk Menu.* (CHR$(4) is the DOS call, control-D.) This line can also be used in other programs whenever you want to get back to the main menu; that's how the *Sorry* program does it.

And that's all there is to it! Enjoy.

```
1  REM -----------------------------------------------------
2  REM        DISK MENU
3  REM -----------------------------------------------------
10  REM DISK DATA
30  NUMBER = 254
40  NAME$ = " >> DISK NAME << "
50  DAET$ = " 2/22/83"
70  GOSUB 1900: REM INITIALIZE
100  REM      PRINT HEADER
110  TEXT : HOME
120  PRINT "DISK # "NUMBER" ";
130  INVERSE
140  PRINT NAME$;
150  NORMAL
160  PRINT " "DAET$
170  PRINT : PRINT
200  REM      READ CHOICES
210  DIM N$(9,1)
220  ONERR GOTO 270
230  FOR I = 1 TO 9
240  FOR J = 0 TO 1
250  READ N$(I,J)
260  NEXT J,I
270  NC = I - 1: POKE 216,0
300  REM      LIST CHOICES
310  GOSUB 1200: REM SPACING
320  FOR X = 1 TO NC
330  HTAB 5: PRINT X" - ";
340  PRINT N$(X,0)
350  PRINT
360  NEXT X
370  PRINT
380  GOSUB 1200: REM SPACING
400  REM      PROMPT & GET INPUT
410  PRINT "WHICH ONE WOULD YOU LIKE (1-"NC")? ";
420  GET Q$
430  IF ASC (Q$) > 32 THEN PRINT Q$;
440  PRINT
500  REM      TEST & EDIT INPUT
510  IF Q$ = " " THEN POKE 216,0: END : REM OPTIONAL EXIT
520  Q = VAL (O$)
530  IF Q = 0 THEN 800: REM ERR
540  IF O > NC THEN 800: REM ERR
550  PRINT "GETTING "N$(Q,0);
560  CALL - 998
570  N$ = N$(Q,1)
580  PRINT
600  REM      RUN BASIC PGM
610  ONERR GOTO 650
620  PRINT D$"RUN "N$
650  REM      BRUN M L PGM
660  ONERR GOTO 700
670  PRINT D$"BRUN "N$
700  REM      FILE NOT FOUND
710  HTAB 1: CALL - 868: POKE 216,0
720  PRINT " -- > SORRY, NOT HERE. < --";
730  GOTO 900: REM BEEP WAIT
800  REM      ILLEGAL INPUT
810  CALL - 998: PRINT B$
820  PRINT "PLEASE TYPE A NUMBER BETWEEN 1 AND "NC".";
900  REM      ERROR BEEP & WAIT
910  GOSUB 1100: REM BEEP
920  GOSUB 1020: REM LONG..
930  GOSUB 1040: REM ..WAIT
940  CALL - 998
950  HTAB 1: CALL - 958
960  GOTO 400: REM PROMPT
1000  REM      DELAY TIMER
1010  GOSUB 1020: REM 32 CYCLES
1020  GOSUB 1030: REM 16 CYCLES
1030  GOSUB 1040: REM 8 CYCLES
1040  GOSUB 1050: REM 4 CYCLES
1050  GOSUB 1060: REM 2 CYCLES
1060  FOR T = 1 TO TIME
1070  NEXT T
1080  RETURN
1100  REM      BEEP-BEEP
1120  PRINT BZ$;
1130  GOSUB 1050: REM PAUSE
1140  PRINT BZ$;
1150  RETURN
1200  REM     ADJUST SPACING
1210  IF NC < 9 THEN PRINT
1220  IF NC < 7 THEN PRINT
1230  IF NC < 5 THEN PRINT
1240  RETURN
1900  REM      INITIALIZE
1910  B$ = "                    " + "           ": REM 21 SPACES
      + 11 SPACES
1920  BZ$ = CHR$ (7) + CHR$ (7)
1930  D$ = CHR$ (4)
1940  TIME = 60
1950  RETURN
2000  REM PROGRAMS INCLUDED:
2001  REM ----------------------
2002  REM
2003  REM --- FORMAT ---
2004  REM "USER NAME","DISK FILE NAME"
2010  DATA "SYSTEM INSTRUCTIONS","SORRY"
2020  DATA "DRINK ME","SORRY"
2030  DATA "THE GOSTAK AND THE DOSHES","SORRY"
2040  DATA "MINGLED INTRICACIES","SORRY"
2050  DATA "TURBOENCABULATOR MAINTENANCE","SORRY"
2060  DATA "THERE AND BACK AGAIN","SORRY"
2070  DATA "WHAT TO DO NEXT","SORRY"
2080  DATA "THE ANSWER","SORRY"
```

FREE!

A six-pack of tasty Apple** posters!

Here's your chance to get any or all of these beautiful 16 x 24 posters Free! except for just a small postage/handling charge! Commissioned by Datamost, and painted by well-known computer artist, Art Huff, each is a unique experience in design that will grace your home or office.

To get your posters, visit your favorite computer store and pick up a coupon where you see the counter display shown below. Fill out the coupon and mail it to Datamost. For each individual poster you want Free, include a registration card from any Datamost software package. Send only $1.75 postage/handling fee with each order. (Note. Without a Datamost registration card, the Posters are available at only $5.95 each, plus same postage/handling charge.)

Be the first to collect the entire series of these magnificent, exclusive, and FREE Apple* posters from Datamost!

Look for this poster display at your computer store.

DATAMOST™ INC

8943 Fullbright Ave., Chatsworth, CA 91311. (213) 709-1202

YES! I Want To Participate in the Datamost

FREE APPLE* POSTER OFFER

Please sign us up and send me a complimentary set of the Six full color Apple Posters.

I would like: __________ window banners.

__________ counter cards w/coupons.

Name: ______________________ Title ______________________

Store/Business Name ______________________________________

Address: ____

City: ______________________ State __________ Zip __________

Mastercard _____ Visa _____ No. ______________________________

DATAMOST™ INC

8943 Fullbright Ave., Chatsworth, CA 91311 (213) 709-1202

*The posters are exclusive products of Datamost, Inc. and are not connected with, or authorized by Apple Computer, Inc.

**Apple is a trademark of Apple Computer, Inc.

Two of mankind's greatest treasures

TIME & ART

Reflected in two outstanding peripherals for your APPLE II

The MBI™ APPLETIME™ Card

$99.00*

- Fully Mountain Software compatible
- Disk included with all software
- Includes Datebook™ - a complete desk calendar
- Time of day
- Calendar date
- Day of week
- Program timer
- International time-keeping ability
- Recharging battery backup
- Complete software formatting
- Offset time/date/day readout

The MBI™ VIP Card

VIP™ Card - "Versatile Interface Peripheral"

(Available for EPSON, NEC, C-ITOH, IDS PRISM, OKIDATA and other graphic printers soon)

VIP™ Card - The Ultimate Graphics Card **$149**

A Centronics Parallel Interface with a Serial Port and Cables featuring:

- Text and graphics screen dump routine
- Graphics with inverse & emphasized modes
- Enlarged picture mode
- Variable line length with left & right margins
- Block graphics
- 90° picture rotation
- Chart Recorder Mode
- A serial port with full RS232 capability
- Software baud rate control from 110 to 9600 baud

*Suggested List Price

MICROCOMPUTER
BUSINESS
INDUSTRIES
CORPORATION

MBI

ADMINISTRATIVE OFFICES: 1019 8TH STREET, GOLDEN, COLORADO 80401 (U.S.A.)
TELEPHONE: (303) 279-8438
TWX: 910-934-0191

Buttonwood Apples

BY KEN LANDIS

Remember back in March when we discussed the sins and virtues of options trading? Well, in this issue, we'll examine a program intended to help options traders to value options and make investment decisions.

OptionX, Crawford Data Systems (350 North Lantana Avenue, Suite 561, Camarillo, CA 93011; 805-484-4159). $145.

Backup policy: Copyable.

System requirements: Apple II with 16K card or Applesoft firmware; Apple II Plus, Apple IIe; one disk drive.

OptionX is designed to help the investor calculate the "correct" value of a stock option, the theoretical value the option should be trading at, based on the strengths, weaknesses, and potential of the underlying security.

Tracking Options. The number of disk drives you have determines the number of options that *OptionX* can track and analyze. With one drive, the program can follow eighty options; with two, the program can follow four hundred options. The more options the program is following, however, the slower it runs, so you won't want to use the two-disk capability unless you need it.

Options are time-sensitive instruments. Their values change daily, for two reasons: Price changes occur in the underlying security and options have a limited life. If an option you own is one day away from expiration and the underlying security would have to move up fifty points in order for you to realize a profit on it, and it's already 3:45 p.m. in New York, you can bet your bottom dollar that the option (if it's a call) is worth virtually nothing.

One of the parameters we need to supply to an option valuation model is today's date, which allows the program to compute the number of days until expiration. *OptionX* does these calculations automatically, allowing the investor to enter the date in any of a variety of formats. When you select the expiration command, *OptionX* displays the expiration dates of the next twelve options you're holding in your portfolio.

Options Analysis. Other options packages we've examined required the investor to enter the options data through an input module and then the analysis was activated. *OptionX* differs from these other options programs in that its analysis process is interactive. The program prompts the investor for information throughout the analysis process and generates an analysis report whenever it has sufficient data to do so.

The investor must inform the program whether it will be evaluating puts or calls. Why? Because calls make money when the price of underlying stock rises, whereas puts make money when the underlying stock falls in price. If *OptionX* doesn't know what to look for, then it can't show the investor what it has found.

Once the investor has supplied the necessary information, the program asks for the symbol of the stock to be analyzed. As mentioned earlier, time is an important parameter in evaluating any option. *OptionX* requests the expiration date of the options under consideration and then asks for the striking price. The striking price of an option is the price at which you may purchase the underlying stock on a call or sell it on a put.

To ensure the accuracy of the program, it's important to make certain that these and all of the other parameters input are correct. One mistake in a crucial variable and the whole study will be wrong, and this could cost the investor a bundle.

The Difference. How else is *OptionX* different from the other options programs on the market? According to the author, "*OptionX* is oriented toward the rapid comparison of either call or put options on the same stock but with different expiration dates and different strikes."

The options programs we've looked at in the past are designed to evaluate a single option or various options strategies. *OptionX* compares and analyzes the time-differentiated options on a single security, so that the investor can choose the best option investment opportunities on that particular security. To facilitate *OptionX*'s analysis, the investor can rapidly enter the required data on as many as nine more options on the same security without having to repeat the stock's symbol each time.

The first report generated by *OptionX* shows the option model being used, the stock being analyzed, the current price of the stock (which was entered by the investor earlier), the stock's volatility (which reflects its past susceptibility to price fluctuations), the current dividend, the market interest rate (which is entered by the investor), the theoretical value of the option, the hedge ratio, and the option's leverage.

What does this report tell us? It shows us if an option is either undervalued or overvalued. Under and overvalued options can present profitable investment opportunities. If we buy something that is undervalued and then resell it at its "true" value, we profit. If we contract to sell something at an overvalued price, and then buy it when the price more closely reflects its actual value and deliver it, we also profit. The theoretical value of an option is the price the option should sell for according to the valuation model being used. This computation helps us identify over or undervaluation.

The hedge ratio illustrates the increase in the price of a call option that results from a one-point increase in the *price* of the underlying stock. Leverage is the percent increase in the *value* of the underlying stock. The hedge and leverage calculations, which the program performs for us, tell us what percentage an undervalued or overvalued option would have to move in order for us to make a profit.

This report relies exclusively on the theoretical, or fair, price predicted by the model. The next step in the analysis is to input the actual closing prices for the options being studied. These prices are available in print from many local newspapers or from *Barron's*. They can also be retrieved from a remote database (such as Dow Jones or CompuServe) using a terminal software package. *OptionX* has no data retrieval capabilities, so any information retrieved electronically must be entered manually, negating any time savings and accuracy associated with electronic updates.

Net Premiums. After the prices for each option in the study have been entered, the computer displays the net premium in points, the net premium in percent, and the implied volatility for each option.

The net premium is the price change in the underlying security that would have to happen in order to profit by exercising (or selling) the op-

Quality Disk Software from SPECTRUM

Apple II

IBM PC

PERSONAL FINANCE MASTER

The premier personal and small business financial system. Covering all types of accoünts including check registers, savings, money market, loan, credit card and other asset or liability accounts, the system has these features:

- Monthly Transaction Reports
- Budgets Income & Expense
- Reconciles to Bank Statements
- Prints Checks & Mailing Labels
- Automatic Year-End Rollover
- Prepares a Net Worth Report
- Searches for Transactions
- Handles Split Transactions
- User-Friendly Data Entry Forms
- Fast Machine Language Routines
- Extensive Error Trapping
- HI-RES Expense/Income Plots

For Apple II (48K) & IBM PC $75.00
Manual & Demo Disk only $15.00

COLOR CALENDAR

Got a busy calendar? Organize it with Color Calendar. Whether it's birthdays, appointments, business meetings or a regular office schedule, this program is the perfect way to schedule your activities. The calendar display is a beautiful **HI-RES** color graphics calendar of the selected month with each scheduled day highlighted in color. Using the daily schedule, you can review any day of the month and schedule an event or activity in any one of 20 time slots.

For Apple II (48K) $30.00

BUSINESS SOFTWARE SERIES

Both Programs $250.00

A user-friendly yet comprehensive double-entry accounting system employing screen-oriented data input forms, extensive error-trapping, data validation and special routines for high speed operation. The series includes these two modules:

GENERAL LEDGER: A complete accounting system with these features:

- Up to 500 accounts and 500 transactions per month.
- Interactive on-screen transaction journal
- Prints checks and mailing labels.
- Produces these reports:
  - Transactions Journal
  - Account Ledgers
  - Income Statement
  - Balance Sheet
  - Account Listings

For Apple II (48K) & IBM PC $150.00

ACCOUNTS RECEIVABLE

A flexible system with these features

- Up to 500 accounts and up to 500 invoices per diskette.
- Prints invoices, customer statements & address labels.
- Interfaces to General Ledger.
- Interactive screen-based invoice work sheet.
- Produces these reports
  - Aged Receivables
  - Sales Analysis
  - Account Listings
  - Customer Balances

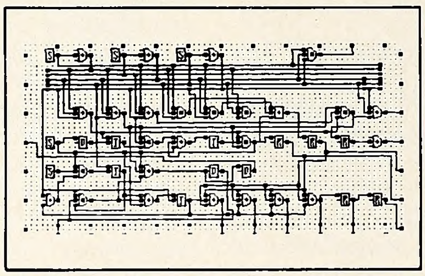
For Apple II (48K) & IBM PC (2 DRIVES) $150.00

LOGIC DESIGNER & SIMULATOR

An interactive **HI-RES** graphics program for designing and simulating digital logic systems. Drawing directly on the screen the user interconnects gates, including **NANO, NOR, INVERTER, EX-OR, T-FLOP, JK-FLOP, O-FLOP, RS-FLOP, USER-DEFINED MACRO** and **N-BIT SHIFT REGISTER** types. Network descriptions for the simulation routines are generated automatically.

The program is capable of simulating the bit-time response of any logic network responding to user-defined source patterns. It will simulate networks of up to 1000 gates. Includes a source pattern editor, **MACRO** editor and network editor. Produces a fan-out report. Simulation output is a string of 1's & 0's representing the state of user selected gates for each bit time of the simulation.

A typical page of a logic drawing looks like this:



For Apple II (48K) & IBM PC (2 DRIVES) $250.00
MANUAL AND DEMO OISK: Instruction Manual and demo disk $30.00

MATHEMATICS SERIES

The Series Includes These 4 Programs:

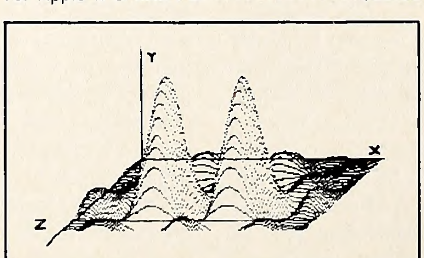
STATISTICAL ANALYSIS I: This menu driven program performs **LINEAR REGRESSION** analysis, determines the mean, standard deviation and plots the frequency distribution of user-supplied data sets.

NUMERICAL ANALYSIS: HI-RES 2-Dimensional plot of any function. Automatic scaling. At your option, the program will plot the function, plot the **INTEGRAL**, plot the **OERIVATIVE**, determine the **ROOTS, MAXIMA, MINIMA** and **INTEGRAL VALUE.**

MATRIX: A general purpose, menu driven program for determining the **INVERSE** and **OETERMINANT** of any matrix, as well as the **SOLUTION** to any set of **SIMULTANEOUS LINEAR EOUATIONS.**

3-D SURFACE PLOTTER: Explore the **ELEGANCE** and **BEAUTY** of **MATHEMATICS** by creating **HI-RES PLOTS** of 3-dimensional surfaces from any 3-variable equation. Disk save and recall routines for plots. Menu driven to vary surface parameters. Hidden line or transparent plotting.

For Apple II & IBM PC $50.00

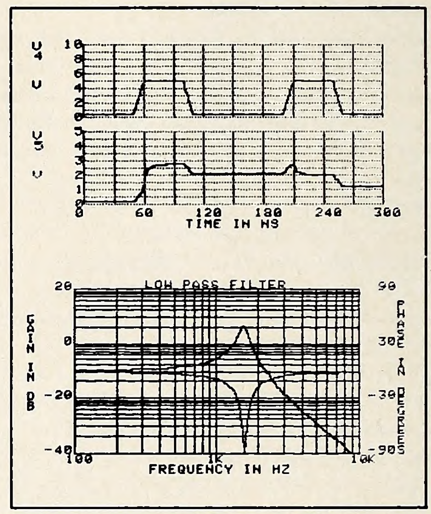


µCAP

Microcomputer Circuit Analysis Program

Tired of trial & error circuit design? Analyze and debug your designs before you build them. With **µCAP** you simply sketch your circuit diagram on the CRT screen and run an **AC, OC** or **TRANSIENT ANALYSIS.** Your circuit may consist of **RESISTORS, CAPACITORS, INDUCTORS, DIODES, BATTERIES, BIPOLAR** or **MOS TRANSISTORS, OPAMPS, TRANSFORMERS,** and **SINUSOIOAL** or **USER-OEFINEO TIME DEPENDENT VOLTAGE SOURCES. µCAP** can analyze any such network containing up to 40 separate nodes. Includes a user controlled **MACRO** library for modelling complex components such as **OPAMPS** and **Transistors.**

Typical **µCAP AC** and **Transient Analysis** graphs:



For Apple II (48K) & IBM PC (2 DRIVES)$475.00
Manual & Demo Disk $30.00

OROERING INSTRUCTIONS: All programs are supplied on disk and run on Apple II (48K) with a Single Disk Drive or IBM PC (64K) with Single Disk Drive unless otherwise noted. Detailed instructions included. Orders shipped within 5 days. Card users include card number. Add $2.00 postage and handling with each order. California residents add 6½% sales tax. Foreign orders add $5.00 postage and handling per product.

SPECTRUM SOFTWARE
690 W. Fremont Ave.
Sunnyvale, CA 94087

master charge

VISA

FOR PHONE ORDERS:
(408) 738-4387
DEALER INQUIRIES INVITED.

tion. The net premium calculation reflects the commission costs the user would incur in executing the transaction (thus, net versus gross). *OptionX* allows investors to input the commission schedules of their own brokers so that the costs used in the calculations reflect those that the investors actually face.

The next report the program generates is a return (or profit) report, which deducts broker commissions from the gross return. The report shows the immediate return to the investor, the maximum return possible, and the maximum return possible as a percent per year. Pressing any of the Apple's keys displays the commissions used in the calculations.

Dividends paid per share and the put conversion value are also included in the profit report. The put conversion value indicates the theoretically "correct" price of a put, based on the price of a call that has the same striking price and expiration date.

At this point, *OptionX* has finished its calculations and has displayed all the information in its reports. The investor may review the results again on the display or send them to a printer.

The volatility calculations performed by *OptionX*, which are used during the analysis, are accessed via the main menu. The volatility update menus are clear and easy to use. The volatility calculations can be done before or after an *OptionX* analysis session. Volatility updates are normally done before the analysis, but if you want to see if there's any significant difference in the valuation you can wait until after the analysis session.

Optional Models. *OptionX* allows the investor to choose either the Black-Scholes model or the Cleeton model to value options. The major difference between the two models is that the Black-Scholes model is based on the mathematical theory of what options "should" sell for, while the Cleeton model is based on what options have *actually* sold for. Black-Scholes is steeped in theory; the Cleeton model is based on past market experience. Although the two formulas are very different, the results obtained by using them are remarkably similar. Of course, it's up to the individual investor to select the method he feels most comfortable with.

According to James C. Moule, the author of the program, *OptionX* is no substitute for "experience, judgment, and hard work. The program makes no attempt to predict the future. I have never discovered a mathematical formula for predicting future prices," says Moule, "and I doubt that one exists."

Moule is right. Any computerized stock market tool is only as good as the skills of the investor who interprets its output. Computers act as an adjunct to the investor, not as an advisor. And as far as predicting the future goes, you can be sure that Moule knows what he's talking about. Accurate soothsayers, whether in the stock market or in a tea parlor, are few and far between.

The program documentation provides a step-by-step walk-through of a typical *OptionX* session. This section is very well written and helps users learn not only the system's operation but some of the theory used in valuing options. The *OptionX* documentation also explains the theory behind the program and its internal workings. A glossary aids the investor in understanding the terms used, and a reference section lists major publications and papers about options. For the serious options trader, this reading is a must.

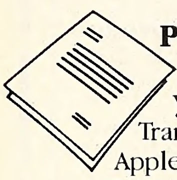
Conclusions. Moule's program is sound. *OptionX* is very well error-trapped and easy to use, and the program documentation is excellent—and, in one section, "final comments," rather humorous as well.

It would seem that the only weak spots in *OptionX* are its inability to use directly captured information from a remote database and its lack of graphic representations. Many investors find it much easier to absorb information from a graph than from a column of numbers. But this is not to say that the report formats of *OptionX* did not present their information clearly. They did.

The outputs of this program can be used by both casual and serious students of options. The theories underlying the analytical models are generally accepted both by academicians and professional traders. Options investors would do well to consider and evaluate *OptionX*'s potential as an analytical adjunct. ❑■

Electronic Communications Made simple for The Apple®: Transpaks™ From SSM.

SSM Transpaks give you all the modem hardware and Transend™ software you need to connect your Apple II/*II*e® to the world of electronic communications.



Public information services. Transend Software lets your Apple do the walking.

Transpak 1 connects your Apple to other Apples, to mainframe systems, or to public information networks. In fact, all Transpaks include valuable subscription offers to three time-saving information services: THE SOURCE℠, Dow Jones News/Retrieval®, and DIALOG's Knowledge Index℠.

THE SOURCE provides instant access to SourceMail business and UPI world news, stock reports, government activity, and private newsletters—even shopping at discount prices.

Dow Jones News/Retrieval keeps you current with the latest corporate news, stock prices, sports statistics, economic surveys, and exclusive news and information from *The Wall Street Journal, Barron's,* and *The Dow Jones News Service.*

The Knowledge Index instantly locates articles, reports or books from an electronic library of over 10,000 journals. You can request information on practically any topic, and in seconds the Knowledge Index will direct you to pertinent articles.

SSM Transpaks will help you do more:

Electronic mail. Transend software gets it there in seconds.

Transpak 2 insures that information you exchange with other Apples will arrive intact. You can send and receive important documents or files of any size over any distance with complete confidence.

With Transpak 3 and a clock card, your unattended Apple can send electronic mail *automatically* to up to *100* other unattended Apples anywhere in the world, at any hour. You'll save time and phone costs by exchanging information at night when rates are lowest.

Each Transpak includes the SSM 300-baud AppleModem™ card with convenient auto-dial/auto-answer capabilities. For the fastest communication available on the Apple, SSM's Transpak 2+ or Transpak 3+ features the SSM TransModem 1200—a 300/1200-baud stand-alone modem.

Choose your Transpak and get started today.

Each Transpak includes all the communications software and hardware you need in one simple package. Your SSM dealer can help you select the Transpak that's right for you. SSM guarantees your satisfaction or your money back.

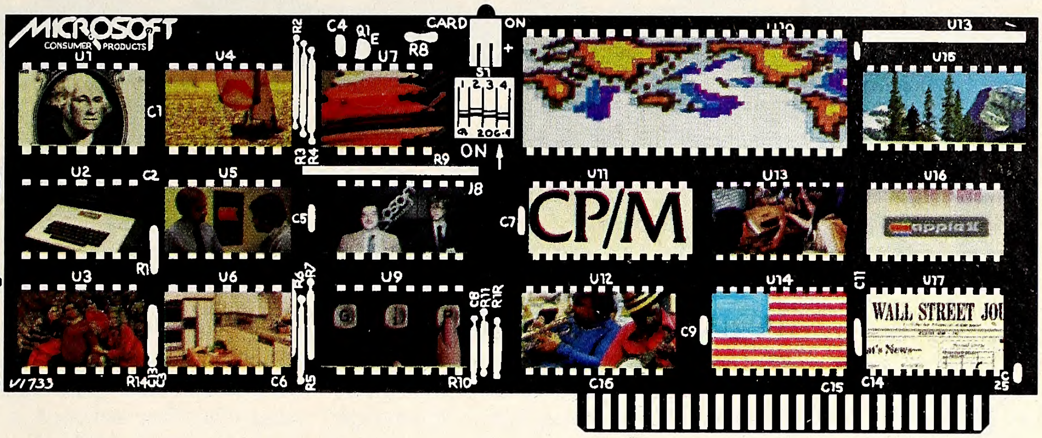
	Software	Hardware
Transpak 1	Transend 1 Terminal Emulator	300-baud AppleModem Card
Transpak 2	Transend 2 Electonic File Transfer	300-baud AppleModem Card
Transpak 3	Transend 3 Unattended Electronic Mail	300-baud AppleModem Card
Transpak 2 +	Transend 2	TransModem 1200 with serial interface and cable
Transpak 3 +	Transend 3	TransModem 1200 with serial interface and cable

Transend it.

SSM Microcomputer Products Inc.
2190 Paragon Drive, San Jose, CA 95131 408/946-7400

SOFTCARD Symposium

by Greg Tibbetts



Welcome to the May installment of SoftCard Symposium. This time we'll deal with the last two character I/O routines, LIST and LISTST, and cover any remaining material related to the handling of character data by the BIOS.

The LIST and LISTST routines control the CP/M logical LST: device category. This category was designed because of the need to produce character output of a printed nature, such as program listings, formatted documents, and even copies of the console I/O data. In speaking of LST:, Digital Research identifies it as "the principal listing device, if it exists on your system, which is usually a hard-copy device, such as a printer or Teletype."

Two things about that statement are worth noting: first, the fact that the LST: device need not be implemented in your system at all; and second, that a physical device in that category need not be a hard-copy device. It's perfectly acceptable to implement LST: as any output-only device (or as the output side of any bidirectional device), though it would take some special circumstances for such a system to be really useful. Normally, then, physical devices that fit into the LST: device category are printers.

Printers are a very diverse class of equipment. They run the range of sophistication, complexity, and cost, from very low to very high. Some, such as the laser printers, even require powerful internal processors dedicated to the task of translating the ordinary ASCII values input to them into the values their hardware must have in order to produce their particular forms of "marks on paper." In all cases, though, the printer as a unit (the word unit is sometimes meant to include both the printer and its interface) is designed to accept simple ASCII code values and to create their human-understood symbolic equivalents on paper.

In addition to creating hard-copy images of the printable ASCII character values, all but the very simplest of printers are set up to accept certain control codes (as we saw when we discussed terminals). These codes, which are either part of the ASCII nonprintable subset or combinations of a lead-in character and one or more printable characters, cause the printer to perform certain operations rather than to print any of the code values themselves.

Printers have in the past, however, been inherently "streaming-type" devices. This concept is best explained by contrasting a printer with a terminal (which is a random access device rather than a streaming or serial device). Terminals deal at all times with presenting a screen of information. At times the difference between one screen of information and the next is only a single character. Terminal displays are extremely alterable (they're completely redrawn sixty times each second), so it became important to increase their efficiency by simply altering only the character or characters that had actually changed, rather than by reprinting the entire screen.

In response to this need, terminal manufacturers provided methods by which the cursor (in this case, the cursor is simply a pointer to the place where the next character will be displayed) could be moved instantly to a given screen location and a character of data altered. This is why we classify terminals as random access; any bit of information on the entire screen can be altered in a random, rather than a serial, fashion. Because of this nature of the screen, manufacturers of system software adapted; and while the addressing codes themselves, for example, may differ from terminal to terminal, the use of cursor addressing is nearly universal.

Printers, on the other hand, have little random access capability. Having produced a character on paper, a printer is generally powerless to change that character completely. Once a character exists in printed form on paper, a simple overstrike is the only thing that can be done to alter it. Printer manufacturers, therefore, were under no compulsion to provide a means to back up the paper-feed mechanism, a way to address the print head within an entire sheet of paper, or other random access features.

Though a few printers today have such features (or limited features of this type), they are still not truly random access in nature because the printed image they produce is still permanent. Consequently, system software has continued to treat the printer as a stream (or serial) device, with the result that only the very standard control codes—such as carriage return, linefeed, tab, and so on—are the least bit universal. As such, no print function tables or other more sophisticated systems exist for the LST: device category, and the system software simply sends characters one at a time, one after the other, to the requested interface in a stream, or serial, manner. It is left to the applications software producers to implement any special codes that may be required, putting them in their business or utility software and outputting them serially when they are to be used. A software producer must, therefore, also know what printer a package is dealing with and must structure codes accordingly.

It's because of all this that the BIOS LIST and LISTST routines are much simpler than their console counterparts.

Essentially, LIST is the routine that takes the ASCII value contained in register [C] (parity bit reset) and sends it to the printer, while LISTST is the routine that tells the program calling it whether or not the printer is ready to receive a character immediately. It's fairly easy to see that LISTST isn't really required in those cases where a printer is able to accept characters and print them as quickly as BDOS can send them, returning control to BDOS and the applications program that is running almost immediately. In such a case, LISTST should always return a ready status for those few programs that use it. On the other hand, if there's no printer connected, or if the LIST routine has not been implemented except as a simple RET instruction, then LISTST should always return not-ready status.

Few, if any, printers are capable of accepting characters as fast as BDOS can send them, so there are very few situations in which LISTST should always return ready. The most common case, therefore, is that of a printer that accepts characters slowly. This case requires a LISTST routine that either accurately checks whether the printer is ready and returns an appropriate response, or always returns not-ready. The latter may seem wrong, since there'll undoubtedly be times when the printer will be ready and LISTST will still indicate the opposite, but because of the way LISTST may be used, this is all right. BDOS does not itself use the LISTST routine to determine whether or not to output to the LST: device; nor should any program accessing LISTST use it to decide unconditionally whether to output characters.

Basically, since the LIST routine won't return when called until it has actually sent the character out, and since printers are slow, the LIST routine is used as a method for programs that have the ability to do other

things while printing to tell whether they should go print a character or handle some other function. The program, however, should understand that it must eventually output the character whether LISTST indicates ready or not.

Because of this last bit of discretion given the program—for example, that after twenty or so calls to LISTST, the program should output the character to LIST regardless of status—it is allowed to keep abreast of other developments and tasks during printing. In this case, programs that make use of this BIOS routine can be more efficient, since not so much time will be wasted looping in the LIST routine while waiting for the slow printer to accept the transmitted character. It must be remembered that no BDOS system call is available to check the status of the LST: device, and that BDOS itself does not use the LISTST routine. This means that a program using LISTST must call BIOS directly.

The final topic in our discussion of printers is that of protocol. This term relates to the *method* of communication rather than to the communication itself. A good simple example of protocol can be found in radio communication, in which some method is needed to identify when one person is finished talking so the other one can begin. In radio transmission, the word "over" is used to indicate the end of a transmission, and when the receiver hears this, he knows that it is now permissible to talk. This convention is said to be the *protocol* used in that communication.

In the case we're dealing with here, the term protocol is used to describe the specific method by which the computer communicates with the printer, from both a hardware and a software standpoint. There are really two types of communication going on—communication between the two pieces of hardware, and communication between the BIOS and the firmware inside the printer or on its interface. Generally speaking, these protocols only pertain to serial devices using the established RS-232 standard for serial communication, although a limited form of hardware protocol is available with parallel interfaces as well.

For our purposes, we can say that the hardware protocol specifies which of the lines (wires) in the serial interface will be used to control the timing and sequence of the data involved in the communications. Since this isn't primarily a hardware column, we won't go into this in detail. Suffice it to say that it is very important in the physical connection of equipment that some hardware protocol be followed to enable the hardware in the computer interface to communicate properly with the hardware in the printer. All of the software in the world won't overcome a basic error in such connections. (Those wishing further information on this subject should consult a good data communications reference work such as *Data Communications for Microcomputers*, by Nichols, Nichols, and Musson, McGraw-Hill, 1982.)

Assuming correct hardware protocol, it is the job of the software in the computer to communicate properly with the software and firmware contained on the interface card or in the printer itself. Normally, with printers and interfaces designed specifically for the Apple, such considerations are unnecessary, since the interface card firmware is designed to handle any protocol measures required. But in situations where the printer interface is a standard serial card and the printer contains a certain amount of intelligence (that is, processing power) that expects a certain protocol, it's up to the software, preferably the system software, to provide that protocol.

The purpose of such protocol is to control the flow of data from the transmitter (the computer) to the receiver (the printer), where the former is considerably faster than the latter. This is done in order to prevent the transmitter from sending so many characters that it overflows the receiver's ability to process them, with the result that data is lost.

Normally, a printer has a memory buffer that is filled as characters are received from the transmitter and emptied as the printer prints them on paper. Since the speed at which the printer can print is so much slower than the speed at which the computer can transmit, it's possible for the buffer to overflow.

With most Apple peripherals, buffer overflow is controlled by using a part of the hardware protocol in the printer or interface to send a not-ready signal when the buffer gets full and a ready signal when the buffer is able to accept more. There are, however, printers which, because they have been designed for use on other machines besides the Apple, do not make use of this type of control. For them, the ready/not-ready signals are reserved for the actual communication with the buffer; for example, a character is placed on the communications circuit and while that character is being received and placed in the buffer, the not-ready status is used. As soon as the character is stored, the ready status comes back on and the next character is sent. No check of overall buffer level is made by the hardware.

It should be noted that if the speed at which data is transmitted is slow enough, the printer will be able to print characters as fast as they are received, using the ready/not-ready states of the communications link to slow the computer down. You may have seen, for example, that with some serial printers, a 1200-baud transmission speed is the highest recommended for use without protocol, even though the printer itself may be capable of receiving at twice that speed or higher. There are two very basic software protocol methods that eliminate such speed mismatch problems; they are ETX/ACK and XON/XOFF.

When the ETX/ACK method is used, the transmitter sends a fixed-size block of data (smaller than the buffer in the printer), followed by a special end-of-text, or ETX, character. This is one of the ASCII control characters, an 03. The transmitter then stops transmitting and waits. The receiver processes the block of data, and when it encounters the ETX at the end it sends back to the transmitter an acknowledge, or ACK, another special control character whose value is 06. The transmitter, which has been waiting, gets the ACK from the receiver and transmits the next block of letters, ensuring that the receiver is ready for the data before it is transmitted.

The XON/XOFF method has an identical purpose but is structured differently. In this case, there are two buffer level values that the receiver can detect—nearly full and nearly empty. As the receiver gets characters from the transmitter and the buffer fills, the receiver is also processing the characters, but at a slower rate. The receiver therefore constantly checks the buffer level, and when it detects a nearly full condition it sends an XOFF character to the transmitter, which quits transmitting in response. As the buffer empties, a nearly empty mark is reached. This is detected by the receiver, which sends an XON to the transmitter and the transmitter begins transmitting again. With this method, the receiver is constantly processing data, but the buffer never overflows.

Since most printers designed specifically for the Apple don't require such protocol (and indeed are not able to handle the protocol by ignoring it if it is sent anyway), the SoftCard BIOS does not implement either of these protocol methods. To have done so would have taken up a not insignificant amount of memory and would have required additional routines to activate and deactivate them as well. In this case, the space that would have been taken up by the protocol routines (and thereby lost to the user) was considered more valuable than the ability to handle the relatively rare occurrence of a printer requiring the protocol.

This sacrificing of adaptability for increased user program space probably has some advocates as well as some critics. Two things do make it unfortunate, though—how difficult it is for users in the field to implement such protocols themselves, and the fact that some printers will not work without them. While it's not impossible to install the protocols and still use the LST: device, it's made especially difficult, since both ETX/ACK and XON/XOFF require the printer to send characters back to the user, and there's no BDOS system call that can be made by an applications program to get a character of input from the printer.

Such a task would require BIOS modification or patching, possibly accomplished through the use of the LISTST routine as a direct BIOS call to return an ACK character generated by the printer, or as a direct call to be constantly polled as a source of XON/XOFF characters. Another method might be to write a specific printer driver that implemented protocol for one's application and to place it in the patch area, substituting it for the normal LIST routine. Users of specific applications programs (such as *Wordstar*) could just write their own direct printer drivers that performed the necessary two-way communications with a peripheral card in a specific slot. This solution would work only for that specific application, however.

A word of warning: The programming tasks in all the cases just mentioned are certainly achievable, but are not trivial. If there's sufficient interest, we might take up such a task in a future column.

The final type of protocol we'll examine involves the use of parity. This protocol is not used for the same purpose as those we have examined so far; rather, parity is used as a limited form of error-checking. As we've said before, the parity bit is the highest order, or leftmost, bit of the eight bits in a byte. CP/M itself does not use the parity bit in normal character output with either the console device or the printer. Some printers, however, may be set to use it or not, and some interfaces are programmable to deal with it.

Basically, the concept of parity is structured around the fact that each eight-bit byte is a collection of binary 1s and 0s. Any collection of bits is said to have either *even parity* or *odd parity*, depending on the number of 1s in the collection. An odd number of 1s indicates odd parity, and an even number indicates even parity.

It can be seen, therefore, that if you wished to transmit only seven bytes of useful data during communication, the eighth bit, or parity bit, could be set or reset as necessary to make the bits all odd parity or even parity, depending on which protocol was in effect. When the parity bit is always reset to 0, the system is said to be "no parity."

When both parties in the communications link are adjusted to transmit in odd (or even) parity mode, a constant check can be made on the bytes being transmitted, and when the wrong parity appears, the receiver can reject the block of data by forcing an error condition and abort the transmission or request another transmission of that block of data if the system allows that to take place. In the cases of printers in general, and printers working with CP/M, there's little point in implementing this activity. CP/M will not accept an error condition reported back from a printer, and indeed, with the short transmission lines that are ordinarily used, errors of this type are unlikely anyway. (When such an error does occur, rare though it may be, it's usually best to ignore it or to repeat the process from the beginning.) For this reason, CP/M doesn't make use of parity, and neither does the SoftCard BIOS.

It's time now to examine the SoftCard BIOS and see just exactly what the LIST and LISTST routines are capable of. Of the two, LISTST is by far the simpler. As the sixteenth entry in the BIOS jump table, LISTST is supposed to return a 00 if the device is not-ready and an 0FFH if the device is ready. Since most printers connected to the Apple are considerably slower than SoftCard, LISTST in the SoftCard BIOS always returns not-ready. This is accomplished by replacing the jump to the list status routine (a three-byte instruction) with an XOR A, which zeros the [A] register, and a RET, to return the BDOS with the zero value. To keep the jump table intact, so that every entry is three bytes long, a NOP (no operation) instruction is placed after the RET. In SoftCard, then, LISTST is always not-ready.

The LIST routine, which is the sixth entry in the jump table, contains a jump instruction to the address 0DB66H in 56K CP/M and 0AB66H in 44K. This routine, like the other character I/O routines, first examines the IOBYTE value to see which of the physical devices is currently the active LST: device.

The four devices available for LST: are TTY:, CRT:, LPT:, and UL1:. Just as we saw in the console device in previous columns, TTY: and CRT: are combined into a single device (which we called TTY:) and are serviced by the TTYOUT driver routine. Therefore, when LIST checks the IOBYTE, values of 00 or 01 cause control to be transferred to TTYOUT. Character output handled in this fashion is identical to that performed by means of the console device, CON:, and by referring back to the column on that device you can see how character output progresses.

An IOBYTE value of 02 designates the LPT: device, commonly called the line printer. This device is serviced by a routine whose address is stored in List Output Vector #1 in the IOCB. If an 02 is found in the IOBYTE, the address contained in the number one vector is loaded into the [HL] register pair and branched to via a JP (HL) instruction.

Had the IOBYTE value been 03, the UL1: device (user-defined list device) would have been the active one. This device is handled by a routine whose address is stored in List Output Vector #2, and so the address loaded into the (HL) register pair would be from that vector location. When the SoftCard is shipped, no user-defined list device exists, and so the same address—that of the LSTOU1 routine (at 0DD2BH in 56K, at 0AD2BH in 44K)—is contained in both vectors.

The LSTOU1 routine—like RDRIN1 and PUNOU1, discussed last month—simply loads the (DE) register pair with the slot value of the

printer device (slot 1) and jumps to one of three hardware routines or to a dummy routine if no card is installed in slot 1. Initially, the address jumped to is that of a simple RET instruction at 0DD3EH. This address is altered by the BOOT routine during initial system boot-up if any usable card has been detected in slot 1. The usable card in this case is a serial card, an Apple Communications card, or an Apple parallel printer card. These cards are serviced by the routines WSER, WCOM, and WPAR, which we've discussed in detail in previous columns. BOOT therefore places the address of one of these three routines in the address field of the jump instruction in LSTOU1 as the system boots up. Whichever routine LSTOU1 branches to uses the slot value in (DE) to find the firmware routines and/or control addresses for the card in that slot and uses these to send output appropriately.

It must be noted once again that in the case of comm cards, the problem encountered before in the PUNCH routine is just as prevalent here. Because of the type of chip in use on the comm card, and the fact that the Z-80 does a preread before every write, the output of a byte to the comm card cannot be done in Z-80 mode. Fixing this problem would involve installing a patch similar to OUTPAT (which we discussed some months ago) into the patch area and altering the List Output Vectors. It should be noted, however, that the use of comm cards as printer interfaces is rare, so such a patch is probably not necessary.

WSER, WCOM, and WPAR all return directly to BDOS after having transmitted the character data to their respective cards. No check is made by BDOS to ensure that the character was sent, however, since control is not supposed to be returned to BDOS until the transmission of the character is complete. (It is for this reason that sending output to a printer that is not on or not-ready causes the system to hang. This does not occur if no printer card is installed, of course, because the address of the RET instruction is left intact by BOOT if no card is detected in that slot.)

We've now covered the LIST and LISTST routines and completed our in-depth discussion of SoftCard's specific character I/O routines. To summarize, then, let's take a look at how BDOS and applications programs use the overall character I/O system. First, there are six specific character I/O routines in the BIOS that BDOS can use to satisfy its nine system calls, and one (the LISTST routine) that, if necessary, can be accessed directly. The nine numbered BDOS system calls follow:

Number	Function	Purpose
1.	Console Input	Get one character from the CON: device.
2.	Console Output	Send one character to the CON: device.
3.	Reader Input	Get one character from the RDR: device.
4.	Punch Output	Send one character to the PUN: device.
5.	List Output	Send one character to the LST: device.
6.	Direct Console I/O	Check and if available get one character from the CON: device or send one character to the CON: device.
9.	Print String	Send multiple characters to the CON: device from memory buffer terminated by ASCII $ character.
10.	Read Console Buffer	Get multiple characters of input from the CON: device and store in designated buffer.
11.	Get Console Status	Return a value of 00 if no character is available from CON: or 0FFH if a character is available.

To perform these functions, BDOS uses the following BIOS routines:

Routine	Used by Numbers	Purpose
CONST	6 and 11	Determine character availability from CON:
CONIN	1, 6, and 10	Get a character from CON:
CONOUT	2, 6, and 9	Send a character to CON:
READER	3	Get a character from RDR:
PUNCH	4	Send a character to PUN
LIST	5	Send a character to LST:
LISTST	unused	Determine readiness of LST:

As you can see from this list, BDOS uses only CONST, CONIN, and CONOUT of the BIOS routines on its own or as part of another function. The remaining BIOS routines are accessed by BDOS only on direct request from an applications program. A minimum BIOS, therefore, only requires that the first three routines be fully functional from BDOS's standpoint, and in fact many older CP/M systems were shipped with only these three routines implemented, leaving the other routines to be implemented by users themselves.

In the SoftCard system, the four devices, CON:, RDR:, PUN:, and LST:, have been assigned specific slots for the sake of simplicity. CON:, if used as an external device (instead of the forty-column screen), is assigned slot 3; RDR: and PUN:, since they each may make up either the input or the output half of a bidirectional device, have been assigned to share slot 2, while LST: has been assigned slot 1.

The BIOS was designed to recognize Apple standard cards of the serial, comm, or parallel type automatically in each of these three slots and to initialize and communicate properly with the card when the appropriate device is called. In this way, for most standard applications, all of the BIOS routines (and consequently the BDOS system calls) were designed to function without user modification. With the exception of the comm card problems and possible problems with protocol we've mentioned, the BIOS fulfills this original design criteria.

Recognizing, however, that individual users might need to implement some different device in one of these categories, BIOS was constructed with two physical devices that could be assigned to each logical device or category. In this way, the user could have an alternative physical device of his own in addition to the standard physical device available for each of the functions.

While in theory this appeared to be an optimum solution, and while all the information necessary to accomplish this task was included in some form in the SoftCard manuals, in practice the task itself required that more tutorial information and guidance be available to most SoftCard users than the manuals could provide. To a large degree, that's the reason for the existence of this column; its purpose has always been to provide additional information concerning the BIOS and the overall function of CP/M, with the goal being a better informed and more capable SoftCard user community.

In coming installments of the column (once we've examined the disk I/O routines, of course), we'll consider examples of installation of routines to handle special peripheral devices. In addition, we'll look at ways the BIOS can be altered to be more effective in dealing with peripherals already recognized. Until next month. . . .

THE VISUAL DOCUMENTATION

SOFTVUE T.M.

Can someone please help us find a program that is right for us?

....Yes.

SoftVue videotapes allow you to watch an expert demonstration of a computer program before you buy the program, or possibly before you buy the computer to run it on. No longer is it necessary to find a salesperson with enough time to show you a program. No longer is it necessary to wade through the manual for hours trying to decide EXACTLY what the program can do. Each tape is a step-by-step tutorial aid that moves you through the computer program at an easy pace. And review is as simple as hitting rewind.

A supplement showing actual print-outs that are made during the demonstration is also included with each program that uses a printer. Thus, you will see exactly how to run the program and exactly what a print-out from that program looks like.

SoftVue demonstration videotapes take the guesswork out of buying computer programs.

Demonstrations available now include APPLE WRITER and the entire PFS series, which includes PFS: FILE, PFS: REPORT, and PFS: GRAPH. New programs in this continuing library of demonstration videotapes will be available soon.

Ask to see SoftVue videotapes at your local computer or software dealer.

Ask your computer dealer for a SoftVue videotape demonstration.

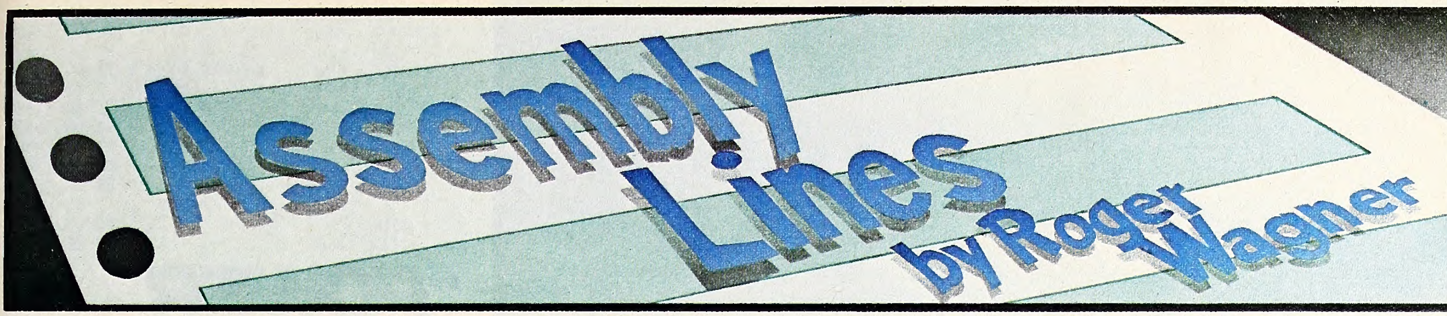
For more information call or write:

SoftVue Inc.
8911 Tavistock
Houston, Texas 77031
(713) 988-2530

VISA MasterCard

PFS: is a trademark of Software Publishing Corp.

APPLE is a trademark of Apple Computer Inc.



Everyone's Guide to Assembly Language, Part 32

Last month's column presented a listing for a hi-res character generator and the theory behind its operation. The generator used an existing character set, loaded at location $9000 in memory, and contained the data for ninety-six ASCII characters.

To create your own character set, all that is needed is a utility for editing the existing character set and creating the new *font*, or character design, that you desire.

Before presenting the listing for the character editor, consider for a moment the information and techniques that must be provided for. This is a very important part of solving any problem, programming or otherwise, and is instrumental in directing and clarifying one's thought processes.

In discussing the character set, you'll recall that each character is represented by a series of eight bytes in the table, and that each dot in the character image is represented by a bit within one of those bytes. The first two considerations, therefore, are how to address the series of bytes that correspond to a given ASCII character and how to identify and alter the bit corresponding to the particular dot in the character image that we wish to modify.

In editing each character, we will want to be able to turn a given bit on or off (set it to 1 or 0) and to move a *cursor* from one bit to another. You'll also recall from last month that each byte of the character data corresponds to one line of the screen image. Within each byte, seven bits are used to map the seven screen dots used to generate a character.

When we edit the individual screen dots, it would be nice if we could use the standard directional keys, I, J, K, and M, to move the cursor around in the box representing the character image.

Speaking of the character box, some thought will have to be given to how the entire character itself will be displayed. We could just print the character on-screen each time a modification was done, but because of the small size this would become tedious after a while. A better approach would be to display a magnified image of the character, upon which our cursor can be positioned to edit any particular bit in the overall image.

To use the editor, we'll also have to be able to specify the character we want to edit, and then to signify later when we are done. To keep things simple, we'll select a character by pressing the equivalent key and store the completed image back in the character table when the return key is pressed.

Loading and saving of the complete table is not provided for in the editor but can be accomplished easily from the immediate mode with bload and bsave. More on that later.

Here, then, is the complete listing, which will be explained in detail. See you at the bottom!

```
                 ****************************************
                 *            CHARACTER EDITOR          *
                 *                2/7/83                *
                 ****************************************
                               ORG   $8000
                 CSW           EQU   $36
                 BASL          EQU   $28
                 CV            EQU   $25
                 CH            EQU   $24
                 CR            EQU   $06
                 CC            EQU   $07
                 MASK          EQU   $08
                 CHR           EQU   $09
                 TABLE         EQU   $9000
                 POSN          EQU   $3C             ;(BAS2)
                 SCRN          EQU   $3E             ;(A4)
                 VECT          EQU   $3EA
                 COUT          EQU   $FDED
                 COUT1         EQU   $FDF0
                 HGR           EQU   $F3E2
                 HCOLOR        EQU   $F6F0
                 HPLOT         EQU   $F457
                 HLIN          EQU   $F53A
                 X1            EQU   $22             ; 34
                 X2            EQU   $54             ; 84
                 Y1            EQU   $17             ; 23
                 Y2            EQU   $58             ; 88
                 VTAB          EQU   $FC22
                 RDKEY         EQU   $FD0C
                 BELL          EQU   $FBDD
                 B1            EQU   %10101010
                 B2            EQU   %01010101

                 CURDAT        EQU   $FFFF

8000: A9 7C      HOOK          LDA   #HCOUT
8002: 85 36                    STA   CSW
8004: A9 81                    LDA   #>HCOUT
8006: 85 37                    STA   CSW+1
8008: 20 EA 03                 JSR   VECT

800B: 20 E2 F3   ENTRY         JSR   HGR
800E: A9 00                    LDA   #$00
8010: 85 06                    STA   CR              ;CR=0
8012: 85 07                    STA   CC              ;CC=0
8014: EA         TITLE         NOP

8015: A9 03      CHRLIST       LDA   #$03
8017: 85 25                    STA   CV
8019: 20 22 FC                 JSR   VTAB
801C: A2 20      START         LDX   #$20
801E: 8A         CH2           TXA
801F: 29 0F                    AND   #%00001111      ;2^4-1
                 * RESULT = VALUE MOD 16
8021: D0 09                    BNE   CONT            ;NOT MULT OF 16
8023: A9 8D                    LDA   #$8D
8025: 20 ED FD                 JSR   COUT            ;PRINT RETURN
8028: A9 14                    LDA   #$14            ;MARGIN FOR NEW
                                                      LINE
802A: 85 24                    STA   CH
802C: 8A         CONT          TXA                   ;RESTORE CHAR
802D: 09 80                    ORA   #$80            ;SET HI BIT
802F: 20 ED FD                 JSR   COUT            ;PRINT CHAR
8032: E8         NEXTC         INX
8033: E0 80                    CPX   #$80
8035: 90 E7                    BCC   CH2

8037: A2 03      MATDSP        LDX   #$03
8039: 20 F0 F6                 JSR   HCOLOR
803C: A2 22      BOX           LDX   #X1
803E: A0 00                    LDY   #>X1
8040: A9 17                    LDA   #Y1
8042: 20 57 F4                 JSR   HPLOT           ;PLOT X1,Y1
8045: A9 54                    LDA   #X2
8047: A2 00                    LDX   #>X2
8049: A0 17                    LDY   #Y1
804B: 20 3A F5                 JSR   HLIN            ;TO X2,Y1
804E: A9 54                    LDA   #X2
8050: A2 00                    LDX   #>X2
8052: A0 58                    LDY   #Y2
8054: 20 3A F5                 JSR   HLIN            ;TO X2,Y2
8057: A9 22                    LDA   #X1
8059: A2 00                    LDX   #>X1
805B: A0 58                    LDY   #Y2
805D: 20 3A F5                 JSR   HLIN            ; TO X1,Y2
8060: A9 22                    LDA   #X1
8062: A2 00                    LDX   #>X1
8064: A0 17                    LDY   #Y1
8066: 20 3A F5                 JSR   HLIN            ;TO X1,Y1
8069: A9 03      MATD2         LDA   #$03
806B: 85 25                    STA   CV
806D: 20 22 FC                 JSR   VTAB
8070: A0 00      GETROW        LDY   #$00
8072: A9 05      GR1           LDA   #$05
8074: 85 24                    STA   CH
8076: B9 64 81                 LDA   MAT,Y
8079: A2 00      SCAN          LDX   #$00
807B: 4A         S1            LSR
```

```
807C: 48                 PHA              ; SAVE RESULT
807D: A9 A0              LDA   #$A0       ; SPACE
807F: 90 02              BCC   PRINTM
8081: A9 FF              LDA   #$FF       ; BLOCK
8083: 20 ED FD  PRINTM   JSR   COUT
8086: 68                 PLA              ; RESTORE ACC.
8087: E8        NXTBIT   INX
8088: E0 07              CPX   #$07
808A: 90 EF              BCC   S1
808C: A9 8D              LDA   #$8D       ; RETURN
808E: 20 ED FD           JSR   COUT
8091: C8        NXTROW   INY
8092: C0 08              CPY   #$08
8094: 90 DC              BCC   GR1

8096: 18        CURSOR   CLC
8097: A5 06              LDA   CR         ; CURSOR ROW
8099: 69 03              ADC   #$03
809B: 85 25              STA   CV
809D: 20 22 FC           JSR   VTAB
80A0: 18                 CLC
80A1: A5 07              LDA   CC         ; CURSOR COLUMN
80A3: 69 05              ADC   #$05
80A5: 85 24              STA   CH

80A7: 20 BD 81  CURCALC  JSR   SCRNCALC
80AA: A4 06     STATUS   LDY   CR
80AC: B9 64 81           LDA   MAT,Y
80AF: A6 07              LDX   CC
80B1: 4A        ST1      LSR
80B2: CA                 DEX
80B3: 10 FC              BPL   ST1
80B5: 90 02              BCC   CLEAR
80B7: B0 04              BCS   SET
80B9: A9 00     CLEAR    LDA   #$00
80BB: F0 02              BEQ   PRNTCURS
80BD: A9 08     SET      LDA   #$08

80BF: 18        PRNTCURS CLC
80C0: 69 6C              ADC   #CURSDATA
80C2: 85 3C              STA   POSN
80C4: A9 00              LDA   #$00
80C6: 69 81              ADC   #>CURSDATA
80C8: 85 3D              STA   POSN+1

80CA: 20 CB 81           JSR   PUTBYTE
80CD: 20 0C FD  CMD?     JSR   RDKEY
80D0: C9 A0              CMP   #$A0
80D2: 90 12              BCC   EDIT       ; CTRL CHAR
80D4: 85 09     CHAR     STA   CHR
80D6: 20 96 81           JSR   POSNCALC
80D9: A0 07              LDY   #$07
80DB: B1 3C     MOVE     LDA   (POSN),Y
80DD: 99 64 81           STA   MAT,Y
80E0: 88                 DEY
80E1: 10 F8              BPL   MOVE
80E3: 4C 37 80  CHRX     JMP   MATDSP

80E6: C9 8D     EDIT     CMP   #$8D       ; RETURN
80E8: D0 14              BNE   E1
80EA: A5 09     ACCEPT   LDA   CHR
80EC: 20 96 81           JSR   POSNCALC
80EF: A0 07              LDY   #$07
80F1: B9 64 81  XFER     LDA   MAT,Y
80F4: 29 7F              AND   #$7F       ; CLR BIT 7
80F6: 91 3C              STA   (POSN),Y
80F8: 88                 DEY
80F9: 10 F6              BPL   XFER
80FB: 4C 15 80  XFX      JMP   CHRLIST

80FE: C9 9B     E1       CMP   #$9B       ; ESCAPE
8100: D0 18              BNE   E2
8102: 38        TOGGLE   SEC
8103: A6 07              LDX   CC
8105: A9 00              LDA   #$00
8107: 2A        SHFT     ROL
8108: CA                 DEX
8109: 10 FC              BPL   SHFT
810B: 85 08              STA   MASK
810D: A4 06              LDY   CR
810F: B9 64 81           LDA   MAT,Y
8112: 45 08              EOR   MASK
8114: 99 64 81           STA   MAT,Y
8117: 4C 37 80  TGX      JMP   MATDSP

811A: C9 8B     E2       CMP   #$8B       ; CONTROL-K
811C: D0 0B              BNE   E3
811E: C6 06     UP       DEC   CR
8120: 10 04              BPL   UPX
8122: A9 07              LDA   #$07
8124: 85 06              STA   CR
8126: 4C 37 80  UPX      JMP   MATDSP

8129: C9 8A     E3       CMP   #$8A       ; CONTROL-J
812B: D0 0F              BNE   E4
812D: E6 06     DOWN     INC   CR
812F: A5 06              LDA   CR
8131: C9 08              CMP   #$08
8133: 90 04              BCC   DX
8135: A9 00              LDA   #$00
8137: 85 06              STA   CR
8139: 4C 37 80  DX       JMP   MATDSP

813C: C9 88     E4       CMP   #$88       ; CONTROL-H
813E: D0 0B              BNE   E5
8140: C6 07     LEFT     DEC   CC
8142: 10 04              BPL   LX
8144: A9 06              LDA   #$06
8146: 85 07              STA   CC
8148: 4C 37 80  LX       JMP   MATDSP

814B: C9 95     E5       CMP   #$95       ; CONTROL-U
814D: D0 0F              BNE   ERR
814F: E6 07     RIGHT    INC   CC
8151: A5 07              LDA   CC
8153: C9 07              CMP   #$07
8155: 90 04              BCC   RX
8157: A9 00              LDA   #$00
8159: 85 07              STA   CC
815B: 4C 37 80  RX       JMP   MATDSP

815E: 20 DD FB  ERR      JSR   BELL
8161: 4C CD 80           JMP   CMD?


8164: 55 AA 55  MAT      DFB   B2,B1,B2,B1,B2,B1,B2,B1 ; 8 BYTE
                                                         WORKAREA
8167: AA 55 AA 55 AA

816C: 7F        CURSDATA DFB   %01111111
816D: 41                 DFB   %01000001
816E: 41                 DFB   %01000001
816F: 41                 DFB   %01000001
8170: 41                 DFB   %01000001
8171: 41                 DFB   %01000001
8172: 41                 DFB   %01000001
8173: 7F                 DFB   %01111111

8174: 00                 DFB   %00000000
8175: 3E                 DFB   %00111110
8176: 3E                 DFB   %00111110
8177: 3E                 DFB   %00111110
8178: 3E                 DFB   %00111110
8179: 3E                 DFB   %00111110
817A: 3E                 DFB   %00111110
817B: 00                 DFB   %00000000

817C: C9 A0     HCOUT    CMP   #$A0
817E: 90 13              BCC   OUT        ; DON'T PRINT CTRL
                                            CHARS
```

```
8180: 48       245            PHA              ; STORE CHAR
8181: 85 3C    246            STA   POSN
8183: 98       247            TYA
8184: 48       248            PHA              ; SAVE Y
               249
8185: A5 3C    250 CALC1      LDA   POSN       ; GET CHAR
8187: 20 96 81 251            JSR   POSNCALC
               252
818A: 20 BD 81 253 CALC2      JSR   SCRNCALC
               254
818D: 20 CB 81 255 PRINT      JSR   PUTBYTE
               256
8190: 68       257            PLA
8191: A8       258            TAY              ; RESTORE Y
8192: 68       259            PLA              ; RESTORE CHAR
8193: 4C F0 FD 260 OUT        JMP   COUT1
               261
8196: 29 7F    262 POSNCALC   AND   #$7F       ; CLR HI BIT
8198: 85 3C    263            STA   POSN
819A: A9 00    264            LDA   #$00
819C: 85 3D    265            STA   POSN+1
819E: 38       266            SEC
819F: A5 3C    267            LDA   POSN
81A1: E9 20    268            SBC   #$20
81A3: 85 3C    269            STA   POSN       ; CHR<96
81A5: 06 3C    270            ASL   POSN       ; *2 = CHR<192
81A7: 06 3C    271            ASL   POSN       ; *4 <384
81A9: 26 3D    272            ROL   POSN+1
81AB: 06 3C    273            ASL   POSN       ; *8< 768
81AD: 26 3D    274            ROL   POSN+1
               275
               276 * POSN=(ASC-$20) *8 BYTES PER CHAR
               277
81AF: 18       278            CLC
81B0: A9 00    279            LDA   #TABLE
81B2: 65 3C    280            ADC   POSN
81B4: 85 3C    281            STA   POSN
81B6: A9 90    282            LDA   #>TABLE
81B8: 65 3D    283            ADC   POSN+1
81BA: 85 3D    284            STA   POSN+1     ; POSN=POSN+TABLE
                                                 ADDR.
81BC: 60       285            RTS
               286
81BD: 18       287 SCRNCALC   CLC              ; ENTER WITH BASL,CH
                                                 SET UP
81BE: A5 28    288            LDA   BASL
81C0: 65 24    289            ADC   CH
81C2: 85 3E    290            STA   SCRN
81C4: A5 29    291            LDA   BASL+1
81C6: 69 1C    292            ADC   #$1C
81C8: 85 3F    293            STA   SCRN+1     ; SCRN=BASL+CH+
                                                 $1C00
81CA: 60       294            RTS
               295
81CB: A0 00    296 PUTBYTE    LDY   #$00       ; ENTER WITH POSN,
                                                 SCRN SET UP
81CD: B1 3C    297 G1         LDA   (POSN),Y
81CF: 91 3E    298            STA   (SCRN),Y
81D1: C8       299 INC        INY
81D2: 18       300            CLC
81D3: A5 3E    301            LDA   SCRN
81D5: 69 FF    302            ADC   #$FF
81D7: 85 3E    303            STA   SCRN
81D9: A5 3F    304            LDA   SCRN+1
81DB: 69 03    305            ADC   #$03
81DD: 85 3F    306            STA   SCRN+1     ; SCRN=SCRN+$3FF
               307 * $3FF TO MAKE UP FOR GROWING VAL
               308 * OF 'Y'
               309
81DF: C0 08    310 DONE?      CPY   #$08
81E1: 90 EA    311            BCC   G1         ; NO
81E3: 60       312 YES        RTS
               313
```

After assembling the listing, bload the character set from last month at location $9000. Then bload the character editor at $8000 (do not brun) and type *call 24576* from Applesoft or *8000G* from the Monitor (Applesoft must be the selected language, though).

When the program is called, the screen will clear and a box with a matrix pattern inside it will appear, along with the complete character set loaded at $9000. If the characters appear scrambled, recheck to make sure you have loaded the character set properly at $9000.

To select a character to edit, simply press any noncontrol key. An enlarged image of that character should appear in the box. To move the editing cursor around, use the left and right arrows to move left and right, and control-J and control-K to move up and down. If you have an Apple IIe, the four directional arrows will also work. Even on a standard Apple II, you may find it easier to hold down the control key with the little finger of your left hand and then press the H, U, J, and K keys with the right hand to move around.

Pressing escape will toggle bits in the character on and off. To save a character back to the table, press return. If you want to start over with a character, simply press the original letter key again.

To save the altered table back to disk, simply press reset, and then type:

BSAVE TABLENAME, A$9000, L$300

You can replace Tablename with any name you wish to give the new character set.

How It Works. Although the listing looks rather long, don't be discouraged. As it happens, much of the listing consists of routines that were presented in earlier issues. For example, lines 243 through 313 (HCOUT) are the character generator that was described in the last issue.

To see how the editor works, let's first consider this overview of the program:

HOOK: Hooks up the character generator, HCOUT, to the output vectors so that the hi-res characters can be printed.

ENTRY: Clears the hi-res screen and initializes the column and row counters to zero.

CHRLIST: This section prints all ninety-six ASCII characters to the screen. We'll examine part of the process in detail shortly.

MATDSP: This section draws the matrix pattern to indicate where the character will be edited. This is also the entry point for the editing loop for each character. This section can be broken down as follows:

> **BOX:** The Applesoft hi-res routines are used to draw a box with four straight lines. This forms the boundary of the matrix area.
>
> **GETROW:** Each byte of the matrix pattern is retrieved here, after which SCAN will process and display the individual bits.
>
> **SCAN:** This section shifts each bit of the row into the carry and, depending on whether it's set or not, displays a solid or an empty block.
>
> **NXTROW:** Increments the row counter (the Y register) until all eight rows have been displayed.
>
> **CURSOR:** Calculates the current cursor position using CC (cursor column) and CR (cursor row).
>
> **CURCALC:** This part, along with PRNTCURS, determines whether the bit at the cursor position is set or not. If it is set, a white cursor is printed. If not, an outline of the cursor is displayed.

CMD?: At this point we are ready to get a command from the keyboard. The general theory is to refresh the screen with the routines in MATDSP each time a command is entered. That way we don't have to update only part of the screen specifically.

If a control character is entered, it is assumed that it will either be a directional key or return, so control is passed to EDIT.

If a noncontrol character is entered, it is assumed that this is a character to be edited. MOVE then retrieves the eight bytes for that character and moves them to the work area (MAT).

EDIT: If the user presses return, ACCEPT will store the character data back in the table. If escape is pressed, the bit within the byte for that row will be toggled.

If one of the directional keys is pressed, the position counters CC and CR are adjusted accordingly.

Pressing a control key other than the legal command characters will generate a bell sound. In any case, after a key is entered, a jump is made back to MATDSP to start the process over again.

And Now with the Magnifying Glass. The preceding overview showed in general how the editor works. Now we'll spend a little more time examining the particular techniques used in each routine. Some of the routines taken from earlier issues will not be described in as much detail as those presented here for the first time. You may wish to refer to previous sections if some parts seem difficult. To help you scan through to just the parts that interest you, each section is keyed to the preceding overview.

HOOK. By storing the address of the HCOUT routine in CSW and then calling VECT ($3EA), all future output will pass through the HCOUT routine, allowing us to print the hi-res characters on the screen.

ENTRY. This is the main entry point to the editor; it serves to clear the hi-res screen and initialize the column and row position of the cursor

to 0,0 (upper-left corner of the matrix).

CHRLIST. To display all the existing characters, CHRLIST loops through the values #$20 through #$7F (32 through 128 = 96 characters). Because we can't print 96 characters on one line, some sort of aesthetic placement is desirable. The format chosen was a group of 6 lines of 16 characters each.

START is the beginning of this loop (X register set to #$20), and CH2 is the top of the printing loop. An interesting problem here is how to determine each time we have printed 16 characters. A separate counter could have been kept, but if it were possible to do a modulo function we could just test for our current character counter for multiples of 16. Because modulo returns the remainder of a division, we would expect a result of zero each time the counter was at a multiple of 16, or in other words, had just finished another line of 16 characters.

As it happens, the AND instruction can be used to perform the equivalent of a modulo for any power of 2. The technique is to do an AND with the value you want minus 1. Because 16 is a power of 2 (2 ∧ 4 = 16), we need only do an AND #$0F followed by a BNE to test for each completed line of 16 characters.

If a line has been finished, a carriage return is printed, followed by the equivalent of an htab 20.

Notice that as each character is printed the high bit is set with an ORA #$80. This is to make COUT happy, as it always expects the high bit to be set on characters to be printed.

MATDSP. This section begins the part that creates the matrix display used in editing the individual characters. This section will be repeated each time a command character is entered.

The first part, BOX, draws a box outlining the character image using the Applesoft HLIN routines.

Once the box is drawn, the individual bytes must be displayed with the status of each bit indicated. The algorithm is to scan through each bit position, printing a space if the bit is clear and printing a rubout (#$FF) if the bit is set. In last month's character table, a rubout was a solid block, so this approach should work. (Note that, if you edit the space character, the matrix pattern will be altered accordingly.)

There are a total of eight bytes to be retrieved and displayed for each character. GR1 is the section that does the equivalent of an htab 5 (for proper screen placement) and then loads a byte from the work area MAT (see line 225). Once a byte is retrieved, SCAN uses the LSR instruction to shift a bit into the carry flag. If the carry is set, a rubout (#$FF) is printed; otherwise a space (#$A0).

Because the accumulator will be used to print a character via COUT, the shifted byte is preserved by pushing it onto the stack on line 98, and later pulling it back off on line 103.

After each seven bits are "printed," a carriage return is printed on lines 107 and 108 and the loop is repeated until all eight bytes have been displayed.

CURSOR. Once the character matrix has been displayed, we need to display the cursor. Lines 115 through 123 use the cursor row and column (CR and CC) to calculate the htab, vtab position. Remember that since we are mirroring actions taken on the text page we can also use the text page as a frame of reference for hi-res screen operations.

STATUS is used to read the particular bit that corresponds to the current cursor position. Note that CR (cursor row) is conveniently equal to which byte in the individual character definition we will need to read, and that CC (cursor column) determines how many bits need to be shifted out to put the one of interest into the carry flag. Depending on whether the bit is clear or set, the accumulator will be loaded with a #$00 or #$08, the purpose of which will become immediately obvious.

PRNTCURS. Since CH and CV ($24,$25) have been set up, we can use a special form of the HCOUT routine, called PRNTCURS, to print a smaller block or a block outline. You'll notice that the hi-res character generator at HCOUT has been modified slightly to use the pointer POSN ($3C,3D) to point to the data table. Our original character generator always assumed that the table would be at $9000. Normally HCOUT sets POSN to point at $9000 on lines 278 through 285.

With POSN set up to point at a special two-character definition table on lines 227 through 243, the PUTBYTE routine will do the equivalent of printing one of the two necessary special characters at the cursor position.

You may wish to compare the HCOUT routine contained in the editor with last month's character generator to see what changes have been made to facilitate the calling by the PRNTCURS routine.

An interesting digression: By avoiding COUT and writing to the screen directly, we are on the verge of being able to do *block shapes*, a technique used in many hi-res arcade-type games.

CMD?. The processing of the command characters is done in this section. The character is read from the keyboard using the Monitor routine RDKEY ($FD0C). This routine will place the ASCII value for the key pressed into the accumulator.

The first major distinction to be made is whether or not a control character has been pressed. Lines 145 and 146 do this, forwarding any control characters to the EDIT section.

If a noncontrol character has been pressed, the user wants to edit that character. CHAR and MOVE use the ASCII value of the key pressed to calculate the position of the data of that character in the table, and then move that data into the work area, MAT. After the move, a jump is made back to MATDSP to refresh the display with the new character and to get the next command key.

EDIT. If a control key is pressed, one of a number of functions must be performed. We will consider these in the order they are executed.

Return: This implies that the user wants to accept the character as displayed and copy it back into the character table. This is done by essentially reversing the process used by CHAR and MOVE (lines 147 through 153).

Toggle: If escape is pressed, the appropriate bit position must be switched to its opposite condition—off to on or on to off. This is done by creating a mask byte with the proper bit set. To do this, the carry flag is set and the accumulator loaded with a zero. When an ROL is done, this set bit will be shifted through the accumulator. By doing the ROL a given number of times (determined by CC) we can set a given bit in the MASK byte ($08).

Once the mask has been created, we need only retrieve the proper byte from the work area (determined by CR) and then mask it with MASK byte (lines 178 through 180). Once this is done, we again jump back to MATDSP to refresh the display and get the next character.

Cursor control. To move the cursor around, we'll use the four directional keys on the Apple IIe keyboard. Even if you don't have a IIe, you can generate the same characters in the manner mentioned earlier in this article. To refresh your memory, the keys we'll use will be control H, U, J, and K, for left, right, down, and up respectively.

The code on lines 185 through 219 is fairly straightforward. The up and down motions are done by incrementing or decrementing the cursor row counter and left and right motions by incrementing or decrementing the cursor column counter. All motions wrap around.

Miscellaneous Notes. You must use reset to exit this program as no exit-key provision has been made.

Although they can be displayed, lower-case characters may not be easy to edit, since they are not easily generated from the Apple II keyboard. Apple IIe owners will have no trouble. It is possible to use the lower-case input routine described in an earlier issue to generate lower-case characters from a standard Apple II keyboard. Simply activate the routine prior to calling the character editor. The escape shift functions will continue to work properly, with presumably no ill effects on the editor routines.

It is worth noting that the character sets used and created by this editor are identical in format to the *DOS Tool Kit Animatrix* character sets, although the character editor provided with that package does have one or two minor, though not inconsequential, features not available in this editor.

Conclusion. This concludes the current discussion of hi-res character generation and editing, and should provide you with the basic principles of these techniques. The idea can be extended into block graphics for arcade-style games or as improvements to the art of hi-res character generation. You might, for example, want to experiment with oversize letters, colored text, or simple animation.

As usual, any comments regarding this or other columns are welcomed and encouraged. ❑

"The fate of the world is in your hands..."

SECRET AGENT:

MISSION ONE

A thrilling ADVENTURE in hi-resolution

An evil scientist
is threatening to
destroy the world
with his awesome
new weapon. And
you're the only one
who can stop him.

An adventure for
the 48K Apple* II/II+
with Applesoft,
Dos 3.3

Available for
$32.95
at a dealer near you
or directly from

Jor-And

P.O. Box 9180
Glendale, California
(213) 247-6658
(California residents
add 6½% sales tax: $2.14)

Dealer/Distributor inquiries welcome

*Apple & Applesoft are registered trademarks of Apple Computer Inc.

THE THIRD BASIC

by Taylor Pohlman

Exploring Business Basic, Part 20

Greetings, Basic buffs! This month's article continues the never-ending (or so it seems) saga of graphics capabilities on the Apple III. We've got a lot of material to go over, and after several months at this it will be assumed that you are pretty familiar with the concepts. This installment will cover the last technique for character graphic animation, and then plunge headlong into the Apple III bit-mapped display, most notably the 140-by-192 sixteen-color mode.

A Last Issue from Last Time. As you may remember, one of the major events from last time was the use of the character download capability of the .console driver to create smooth animation by rapid switching of character definitions. We proved how powerful the technique was by smoothly scrolling hundreds of little bug heads from one side of the screen to the other. The question that somewhat naturally came up was, "How can I move just one head across the screen?" Curiously enough, it's a little harder; so it's worth showing a program in order to help you accomplish it. If you read last month's column, you'll notice a fair amount of similarity in structure with other programs in that article.

The trick last time to moving lots of bug heads smoothly was to print alternating sets of character 128 and character 129 across the screen and then download successive definitions of each character. The progression of definitions showed the head making a step-by-step transition from one character cell to the other. Since a character cell is seven dots wide, it takes eight steps to move a head from completely within one cell to completely within the next one. By reversing the process for every other cell, the illusion of smooth motion is created. To create the character-cell definitions, we used the font and shape editor described in the February issue (part 17). However, in case you missed that issue, or couldn't summon the strength to type the whole thing in, the following program uses data statements to define the bug-head transitions. Feel free to edit your own head definitions and substitute the appropriate routine if you want.

As was mentioned previously, moving a single head across the screen is a little trickier than flipping one character definition to the next. The simplest solution is to print a string of characters across the screen, each one a different character number, but initially all defined to appear as a blank. The program can then redefine the characters, one at a time, and give the appearance of motion across the screen as each successive character is given the head definition. The following program illustrates one way to use this approach to solve the problem:

```
10 DIM a%(511),ctrlist$(7),head1$(6),
     head2$(6)
20 INVOKE"/basic/download.inv","/basic/
     request.inv"
30 q$=CHR$(34):array$="a%":name$=
     ".console"
40 fg$=CHR$(19):bg$=CHR$(20)
50 mblue$=CHR$(6):white$=CHR$(15)
60 bw$=fg$+mblue$+bg$+white$
70 text40$=CHR$(16)+CHR$(1)
```

Line 10 defines various arrays to be used in the program. Ctrlist$ contains the character redefinitions for downloading to the character generator, and the two head$ arrays contain definitions of the bug head coming and going. The other lines establish constants that will be used later.

Now, to initialize the data and get ready to scroll, we will use several routines, which will be discussed in turn:

```
80 GOSUB 700:REM get the head definitions
90 GOSUB 800:REM load the print line
100 GOSUB 900:REM set up control strings
110 GOSUB 600:REM set up screen
```

The first routine loads the head definitions, in this case from data statements:

```
700 RESTORE
710 FOR i=0 TO 6
715   head1$(i)="":head2$(i)=""
720   FOR j=0 TO 3
725     READ a%:h$=HEX$(a%)
730     head1$(i)=head1$(i)+CHR$(TEN
          (MID$(h$,1,2)))+CHR$(TEN(MID$
          (h$,3,2)))
735     NEXT j
740   FOR j=0 TO 3
745     READ a%:h$=HEX$(a%)
750     head2$(i)=head2$(i)+CHR$(TEN
          (MID$(h$,1,2)))+CHR$(TEN(MID$
          (h$,3,2)))
755     NEXT j
760   NEXT i
765 RETURN
770 DATA 7215,32545,838,15360,0,0,0,0
772 DATA 14430,32322,1548,30720,0,
     256,1,0
774 DATA 28732,31748,3096,28672,1,
     769,2,256
776 DATA 24696,30728,6192,24576,258,
     1794,4,768
778 DATA 16496,28688,12384,16384,773,
     3844,8,1792
780 DATA 96,24608,24640,0,1803,7944,
     17,3840
782 DATA 64,16448,16384,0,3607,16144,
     291,7680
```

6 GAMES IN 1...

You start as an amoeba trying to advance up the evolutionary chain to become a human. To attain this goal you will have to go through 6 distinctively evolutionary steps:

- You're an amoeba fighting microbes while trying to eat DNA.
- You're a tadpole eluding fish to catch waterflies.
- You're a rodent chased by snakes while looking for cheese.
- You're a beaver attacked by alligators while building a dam.
- You're a gorilla protecting his food from monkeys.
- You're a man battling genetic mutants in an elastic background!

Do you have the instincts and reflexes to survive and evolve to each higher life form? It's a challenge that grips you as no other game!

- UP TO 99 LEVELS
- EITHER A KEYBOARD OR JOYSTICK CONTROL
- WRITTEN IN MACHINE LANGUAGE
- COLORFUL HIGH-RESOLUTION GRAPHICS
- KEEPS TRACK OF TOP 10 SCORES
- FULL CONVENIENCE CONTROLS
- ARCADE SOUND
- EVEN AN INTERMISSION!

PLAY EVOLUTION ON APPLE II, APPLE II PLUS, COMMODORE 64 OR IBM PC.

Hailed by leading reviewers as one of the most exciting games available. EVOLUTION is truly different. Try it today!

Created by Don Mattrick and Jeff Sember.

THE INCREDIBLE NEW PROGRESSIVE GAME FROM **SYDNEY.**

EVOLUTION

Design: Ian MacLeod, TCA

ONLY THE STRONG SURVIVE.

SYDNEY DATAPRODUCTS, INC.
#129 · 444 Camino Del Rio South, San Diego, CA 92108 (619) 298-5886

Purchase EVOLUTION at progressive computer stores everywhere. Dealer enquiries invited.

The head-loading routine is identical to one discussed last time, so you're on your own. If you want to use a font editor to create the heads, just load up the head$ arrays in the same sequence as found in the data statements (arranged as they would appear in an actual font definition).

Next comes the routine to create a print string for the screen and to redefine all the characters in it to blanks:

```
800  line$=""
810     FOR i=0 TO 31
820     line$=line$+CHR$(128+i)
830     NEXT i
840  cr0$=CHR$(0)
850  ctrl$=CHR$(1)+cr0$+cr0$+
     cr0$+cr0$+cr0$+cr0$+cr0$+
     cr0$+cr0$
860  FOR i=0 TO 31
870     SUB$(ctrl$,2,1)=CHR$(i)
880     PERFORM control(%17,@ctrl$)
        name$
890     NEXT i
895  RETURN
```

Notice in lines 810 through 830 that line$ is built containing all the characters from 128 through 159, which are the displayable versions of character definitions 0 through 31. We could have used more characters, but it is necessary to skip character number 32 (space) or risk redefining every space character on the screen. As long as we just redefine control characters (0 through 31), we won't disturb anything important. Of course, with some thought, you can build line$ to any reasonable length.

Next comes the routine that defines the character download strings, defining the successive versions of the characters:

```
900  FOR i=1 TO 6
905     ctrlist$(i)=CHR$(2)+
        CHR$(0)+head1$(i)+CHR$(1)+
        head2$(i)
910     ctrb$(i)=CHR$(1)+CHR$(0)
        +head2$(i)
915     ctre$(i)=CHR$(1)+CHR$(31)
        +head1$(i)
920     NEXT i
925  ctrlist$(7)=CHR$(2)+CHR$(0)
     +head2$(0)+CHR$(1)+head1$(0)
930  ctrb$(7)=CHR$(1)+CHR$(0)+
     head1$(0)
935  ctre$(7)=CHR$(1)+CHR$(31)+
     head2$(0)
940  RETURN
```

In addition to the ctrlist$ definitions, which define the transitions of the characters in the middle of the screen, there are two special arrays, ctrb$ and ctre$. Ctrb$ is the head definition as it appears on-screen in the leftmost character position, while ctre$ is the definition of the rightmost character, as it disappears.

Lastly comes the screen setup routine:

```
600  PRINT text40$;bw$
605  PRINT CHR$(21);"9";
610  HOME:PRINT:PRINT
620  RETURN
```

This just turns on the forty-column mode, sets the color to blue on a white background, turns off character wrap (line 605), and clears the screen to white.

Now the fun begins:

```
120  PRINT line$;
150  ON KBD GOTO 200
160  FOR i=1 TO 7:PERFORM control
     (%17,@ctrb$(i))name$:NEXT
165  FOR x=0 TO 30:FOR i=1 TO 7
170     SUB$(ctrlist$(i),2,1)=CHR$(x)
        :SUB$(ctrlist$(i),11,1)
        =CHR$(x+1)
175     PERFORM control(%17,
        @ctrlist$(i))name$
180     NEXT:NEXT
185  FOR i=1 TO 7:PERFORM control
     (%17,@ctre$(i))name$:NEXT
190  GOTO 160
```

We stated earlier that the animation would occur by successively redefining our string of characters. Line 120 prints the character string on-screen, and line 150 sets up an on kbd jump to exit from the scrolling. Scrolling begins in line 160 by successively redefining the first character in the string through the seven phases required to bring the head fully into the first character position. Lines 165 through 180 then set up a major loop to proceed through the characters in line$ (remember that they are now printed on the screen), taking each through the seven-phase redefinition necessary for the smooth scrolling. Notice that the same ctrlist$ definition is used each time, with the appropriate character numbers plugged in in line 170. If you are unsure as to how the perform-control statement works, review last month's article, the section in the *Standard Device Drivers Manual* on .console control functions, and the Request.inv documentation on the Basic disk.

All that remains now of our program is the keyboard service routine at line 200 and the cleanup and exit routine at line 500, to wit:

```
200  OFF KBD
210  IF KBD=27 THEN 500
220  ON KBD GOTO 200
230  RETURN
500  PRINT CHR$(21);"="
510  PRINT CHR$(22);CHR$(14);
520  TEXT:HOME
530  nam$=q$+"/basic/standard"+q$
540  PERFORM getfont
     (@nam$,@array$):PERFORM
     loadfont(@array$)
550  PRINT CHR$(15);
560  END
```

Well, that's it. When you run this program, the little creature's head should appear on the left side of the screen, move smoothly to the right, and disappear somewhere around the thirtieth character position, only to reappear again on the left of the screen. One thing is noticeable, however. If you watch closely, the head appears to zip on and off the screen much faster than it chugs across the main part of the screen. This is because the routines at lines 160 and 185 are much faster than the main routine in lines 165 through 180. Although the routine can be speeded up somewhat, more drastic measures are required to make it substantially faster. The tradeoff, as usual, is memory space for tables. By prestoring the results of the various string substitutions in a large string array, the whole sequence can be rewritten as a simple loop. We will declare a string array zoom$ to accomplish this, with the following changes to the program above:

```
10   DIM a%(511),ctrlist$(7),
     head1$(6),head2$(6),zoom$(231)
160  FOR i=1 TO 231:PERFORM
     control(%17,@zoom$(i))name$:
     NEXT
165  GOTO 160
940  FOR z=1 TO 7:zoom$(z)=
     ctrb$(z):NEXT
945  FOR x=0 TO 30:FOR i=1
     TO 7
950     SUB$(ctrlist$(i),2,1)=
        CHR$(x):SUB$(ctrlist$(i),
        11,1)=CHR$(x+1)
955     zoom$(7*(x+1)+i)=
        ctrlist$(i)
960     NEXT:NEXT
965  FOR z=225 TO 231:zoom$(z)=
     ctre$(z-224):NEXT
970  RETURN
```

Notice that we have added code in lines 940 through 970 to put the various string values into the zoom$ array, so line 160 now performs the entire sequence much faster than before (you can now delete lines 170 through 190). The difference in speed should be really noticeable.

The uses of this basic idea are practically unlimited. For example, by splitting up the string and printing it in various places on-screen, you can cause the head to move around, disappear from one spot, and reappear somewhere else. Have fun!

Can't Tell One Pixel from Another without a Bit Map. While the preceding retrospective into character graphics was important to clear up some issues, the real purpose of this article is to get heavily involved in the graphics modes of the Apple III, called a *bit-mapped* display because the image is created by reading out bits of data from certain areas of memory and translating them to dots (pixels) on the display screen. As a reminder, let's look at the program we concluded with last time, which gives a hint as to an important capability of the Apple III graphics driver, the setting of priorities in the color table. Remember that all communication to and from the actual bit map is via the .grafix driver, and the Bgraf.inv routine is used to make the functions of the driver easier to perform. The Bgraf procedure setctab is used to change single color entries in the table in a much easier manner than the .grafix driver permits directly. Type in the following program, and we'll begin our exploration:

```
10  INVOKE"/basic/bgraf.inv"
15  OPEN#1,".grafix"
20  m$=" ** **** ** **** *** * ****  ** *****"
25  PERFORM initgrafix
30  PERFORM grafixmode(%3,%1)
35  PERFORM pencolor(%13)
40  PERFORM fillcolor(%4):PERFORM
    fillport
45  PERFORM setctab(%4,%9,%9):
    PERFORM setctab(%13,%9,%9)
50  PERFORM fillcolor(%9)
55  PERFORM viewport(%0,%6,%0,%191):
    PERFORM fillport
60  PERFORM viewport(%133,%139,%0,
    %191):PERFORM fillport
65  PERFORM fillcolor(%4)
70  PERFORM viewport(%0,%139,%0,%191)
75  PERFORM grafixon
80  FOR j=7 TO-77 STEP-1
85    PERFORM moveto(%j,%180):
      PRINT#1;m$
90    NEXT j
95  GET a$:IF ASC(a$)<>27 THEN 80
100 PERFORM release:PERFORM
    release:PERFORM release
105 CLOSE:INVOKE
110 TEXT
115 END
```

Setting Your Priorities. In the little demonstration of color priorities above, we set the primary colors to be 4 and 13—the fillcolor and pencolor respectively—in lines 35 and 40, and fill the screen with color 4 (dark green). The color-table definition in line 45 says that any time color 4 or color 13 is printed over color 9, the result will remain color 9. Lines 50 through 65 use the window capability to set up two orange (color 9) borders. Without the changes to the default color table, anything printed in the first or last column of the screen would destroy the borders. A fairly complete description of this process can be found in the device drivers manual and in the Business Basic manual (volume 2), but running the program is the best way to see how this works. The loop in lines 80 through 90 makes the print string run backward, one pixel position at a time. As the string runs off the left side of the screen, the priority established in the color table ensures that neither the fill nor pen color will disturb the color 9 bars at the edge of the screen.

This very powerful feature of the Apple III graphics driver is exploited very little by programmers but will be the subject of much of the rest of this article. One comment is important before we proceed, however. If you don't have a color monitor, there is no need for despair. The Apple III automatically translates color output into sixteen shades of gray (or green) on your monochrome display, and the color values in the following examples were arrived at to make sure they would look okay without color. So, get a good grip on your Bgraf.inv module, 'cause here we go!

Becoming a Fan of Hi-res Graphics. The next program illustrates the abilities of the color-priority table on a much grander scale, and even suggests how the capability can be used to give the illusion of depth to an image on the screen. As explained in the Basic manual section on Setctab, every pixel to be plotted onscreen is first passed through the color table, and converted, if necessary, before being actually drawn. The word *priority* is somewhat of a misnomer, actually, since the entries in the table can produce any color as a result of plotting one color over another. It's just as easy to specify that plotting dark green over orange will produce white as to say that orange will be unaffected. In addition, the default is what you would expect; that is, the color you plot with is the color you get on-screen.

Without further ado, let's look at the program:

```
50  REM 7, 11, and 14 are best background
    colors
100 OPEN#1,".grafix"
110 INVOKE"/basic/bgraf.inv"
120 black%=0:blue%=6:orange%=9:
    green%=12:white%=15
130 dgreen%=4:brown%=8:gray%=10:
    yellow%=13
140 vector(1)=dgreen%:vector(2)=brown%:
    vector(3)=gray%:vector(4)=yellow%
150 PERFORM grafixmode(%3,%1)
160 PERFORM initgrafix
170 GOSUB 1000
```

The program opens with the usual initialization and defines a number of variables as color constants in lines 120 through 140. Line 150 sets the graphics mode to three, the 140-by-192 color screen. This is the unrestricted sixteen-color mode, which will be used for the rest of this article. After initializing the graphics screen, we set up the changes to the color table in a routine at line 1000:

```
1000 PERFORM setctab(%dgreen%,
     %blue%,%blue%)
1010 PERFORM setctab(%dgreen%,
     %orange%,%orange%)
1020 PERFORM setctab(%dgreen%,
     %green%,%green%)
1030 PERFORM setctab(%dgreen%,
     %white%,%white%)
1040 PERFORM setctab(%brown%,
     %orange%,%orange%)
1050 PERFORM setctab(%brown%,
     %green%,%green%)
1060 PERFORM setctab(%brown%,
     %white%,%white%)
1070 PERFORM setctab(%gray%,
     %green%,%green%)
1080 PERFORM setctab(%gray%,
     %white%,%white%)
1090 PERFORM setctab(%yellow%,
     %white%,%white%)
1100 RETURN
```

In the case above, we are using the color table to establish priorities for the vector colors to be drawn. As you can see, dark green won't affect anything but the background, since drawing dark green over blue, orange, green, or white will have no effect. Brown, however, will draw over blue but will lose to orange, green, and white. Gray draws over blue and orange but has no effect on green and white. Yellow affects everything but white. These statements are easy to understand and could be programmed in simple background situations, but imagine what it would be like to draw lines or shapes on complex backgrounds. Checking every spot on-screen to see what color table is already there would be incredibly time-consuming. Since the color table is built into the .grafix driver, using it causes the check to be done at assembly-language speeds, without the programmer having to worry about it.

Enough praise for .grafix. To continue:

```
200 INPUT"Background color number: ";a$
210 a=VAL(a$):IF a$="" OR a<0 OR a>15
    THEN 510
```

To see the effect of the color table further, the program allows you to set the overall background color for the screen. You should choose this carefully, since the color table will affect the results of certain choices. Colors 7, 11, and 14 should give good results.

Next, we'll use the viewport-fillport combinations to create various color bars on-screen, after first turning on the screen and clearing to the background color you have just chosen:

```
220 PERFORM grafixon
230 PERFORM fillcolor(%a):PERFORM
    fillport
300 PERFORM viewport(%20,%30,%40,
    %170)
310 PERFORM fillcolor(%black%):
    PERFORM fillport
320 PERFORM viewport(%35,%40,%30,
    %160)
330 PERFORM fillcolor(%blue%):
    PERFORM fillport
340 PERFORM viewport(%52,%65,%40,
    %170)
350 PERFORM fillcolor(%orange%):
    PERFORM fillport
360 PERFORM viewport(%77,%92,%35,
    %175)
370 PERFORM fillcolor(%green%):
    PERFORM fillport
380 PERFORM viewport(%110,%120,%20,
    %170)
390 PERFORM fillcolor(%white%):
    PERFORM fillport
```

Note that normally you would set up the screen and then turn on the display—that is, move statement 220 to beyond 390. In this case, it's worth noticing how the color bars are set up, especially if you use different background colors than those suggested. The lines above can be replaced by some data statements and a loop for more compactness, since each set of two statements is identical except for parameters, but this way you get a feel for exactly what's going on. Next, we draw vectors over our landscape and observe the color table effects:

```
400 PERFORM viewport(%0,%139,%0,
    %192)
410 FOR i =1 TO 4
420   FOR j=1 TO 10
430     horiz=(i−1)*45+j*4
440     PERFORM moveto(%140,%horiz)
450     PERFORM pencolor(%vector(i))
460     PERFORM lineto(%0,%96)
470   NEXT j
480 NEXT i
500 GET a$:IF ASC(a$)<>27 THEN
    TEXT:GOTO 255
```

The routine above sets the viewport to the whole screen and then uses a double loop to draw lines from the point X=0,Y=96 (middle of the left side of the screen) to various points on the right-hand side. The effect is somewhat like a fan (or rays projected from a single source). Because of the color priorities established, the effect is quite dramatic, since the rays appear to go behind some objects and in front of others. Pressing escape in line 500 terminates the program, like so:

```
510 TEXT
520 PERFORM release:PERFORM release:
    PERFORM release
530 INVOKE:CLOSE
540 END
```

Running this program with various background colors and various settings of the color table will allow you to experiment with the setctab procedure enough to get to know its capabilities. You might try setting different result colors, for example, setting some to the background color, and observing the effects.

The Bugs Are Back. Although the program just listed is interesting, even dramatic in its own way, you're paying to see the creatures from space, right? So let's bring on the bugs! Actually, it is useful to look at combining the techniques we have already discussed with the new capabilities of the bit-mapped graphics display. The following program introduces a new creature, somewhat larger than his character graphics ancestors, which was originally created using the Shape Editor from the February issue. We will use this new kind of "bug" (which really resembles a squid) to begin our discussion of animation on the hi-res screen. The next program defines the creature and lets us move him around on-screen. We begin with the usual declarations:

```
 5 DIM mshape%(1,15),x%(255),y%(255)
10 INVOKE"/basic/bgraf.inv"
15 OPEN#1,".grafix"
20 GOSUB 1000
```

Mshape% in line 5 is the array that will contain the creature's shape definition (in two parts). The first part (column) will show the tentacles extended, with the second column defining the tentacles retracted. This allows simple animation, along with movement. The x% and y% arrays will be covered in a minute. After invoking the Bgraf.inv module, we gosub to line 1000 for the shape definition, like so:

```
1000 RESTORE
1010 FOR j=0 TO 15:FOR i=0 TO 1:READ
     mshape%(i,j):NEXT:NEXT
1020 RETURN
2000 DATA 0,0
2005 DATA 0,0
2010 DATA 0,0
2015 DATA 0,1984
2020 DATA 1984,2080
2025 DATA 2080,4752
2030 DATA 4752,4112
2035 DATA 4112,5008
2040 DATA 5008,2080
```

```
2045 DATA 2080,1984
2050 DATA 1984,2336
2055 DATA 2976,2720
2060 DATA 1344,4752
2065 DATA 0,0
2070 DATA 0,0
2075 DATA 0,0
```

Lines 1000 through 1020 read the contents of the data statements into the mshape% array, thereby defining the two versions of our creature. Hang on to these data statements; they will be used in several other programs later on in this article but are really too dull to repeat. Next comes the screen setup:

```
25 PERFORM initgrafix
30 PERFORM grafixmode(%3,%1)
35 PERFORM fillcolor(%4):PERFORM
   pencolor(%13)
40 PERFORM fillport
45 PERFORM viewport(%0,%139,%0,
   %191)
50 PERFORM grafixon
```

The lines above clear the screen to dark green and set the pen color to yellow. Note that we again use the 140-by-192 color mode. After this setup we initialize some general variables:

```
55 f%=4:s%=16:z%=0
60 x%(8)=-3:x%(21)=3:y%(11)=
   3:y%(10)=-3
65 i=70:j=90
```

Of note above is line 60, which establishes some entries into the large x% and y% arrays. A quick check of your keyboard chart should tell you that ASCII 8 is the left arrow key, 21 is the right arrow key, 11 is the up arrow, and 10 is the down arrow. Left and right correspond to movement in the X direction, and up and down in the Y direction. This makes it obvious that we will be using values in the x% and y% arrays to indicate the amount and direction of movement, depending on which cursor key is pressed (left and down both being negative movements). As we have seen before, such techniques waste space but increase speed. We'll see an even more interesting application of this technique in just a minute. For now, on with the program:

```
70 PERFORM moveto(%i,%j)
75 IF r%=0 THEN r%=16:ELSE r%=0
80 PERFORM drawimage(@mshape%
   (0,0),%f%,%r%,%z%,%s%,%s%)
85 GET a$:a=ASC(a$)
90 IF a<>27 THEN i=i+x%(a):j=j+y%(a):
   GOTO 70
```

The lines above constitute the main loop of the program. After moving to the position established by the initial values of i and j, r% is set to alternate between 0 and 16, which will be our bit index into the mshape% array. The drawimage procedure (from Bgraf.inv) is then used to put the appropriate bit pattern on-screen at the current pen location. See your Basic manual for a specific discussion of drawimage, but fundamentally the parameters look like this:

```
PERFORM drawimage(@array,
%Num.Row.Bytes,%X.skip,%Y.skip,
%Dr.width,%Dr.height)
```

This definition parallels, of course, the drawblock capability of the .grafix driver and allows you to specify a source array, the number of bytes in a given row of the array (needed to find the offset for row 2 and so on), the number of bits to skip in the row before drawing, the number of rows to skip in a column before drawing, the number of row bits to draw from that point, and the number of rows to draw. This is sufficient information to define any arbitrary rectangular block of bits in any given array. Any bits in the array that are on (that is, 1s) are drawn in the current pencolor, and any that are off (0s) are drawn in the current fillcolor.

Lines 85 and 90 get keystrokes and modify the values of i and j according to the contents of that character location in the x% and y% arrays, and then jump back to 70 to redisplay the creature at the new location. If the character typed is an escape (27), then cleanup and termination are done:

```
100 TEXT
105 PERFORM release:PERFORM release:
    PERFORM release
110 CLOSE:INVOKE
115 END
```

If you type nothing else in from this article, at least try this program. You should have some fun watching the creature swim around the screen at your command. For a little more excitement, try adding the following:

```
61 x%(55)=-3:y%(55)=3:x%(49)=
   -3:y%(49)=-3
62 x%(57)=3:y%(57)=3:x%(51)=
   3:y%(51)=-3
63 x%(52)=-3:x%(54)=3:y%(56)=
   3:y%(50)=-3
```

The lines above set up additional definitions of possible X and Y movements. Close examination of your ASCII chart will show that the codes correspond to numbers on the numeric pad of the Apple III. ASCII 55 in line 61 corresponds to the 7 character, which is in the upper left-hand corner of the keypad. Both x% and y% are affected, x% being decremented (indicating movement to the left) and y% being incremented (indicating movement up). This combination creates diagonal motion. Quick comparisons with the rest of the characters will show the remaining relationships. Add these lines and run the program again. You'll find that you can control the creature completely from the pad! Note also that changing the constant value will change the amount of movement in any direction.

Onward, Ever Diagonally. Here's hoping that the program above has whetted your appetite for more creature features. The next program will combine creature movement with the windowing techniques of the graphics driver to create interesting motions of several creatures at once. First, however, some fooling around should be encouraged. Try changing the displacement constants in the previous program to values higher than three. For example, try:

```
60 x%(8)=5:x%(21)=-5:y%(11)=
   5:y%(10)=-5
```

Now use the cursor keys. Makes a mess, right? Right. What happens is that while the previous drawblock image had enough fillcolor bits (zero-value bits) surrounding the image to blank out any movement of three pixels in any direction, when we move five at a time, some old bits are left on-screen without being cleaned up by the next occurrence of drawimage. A quick glance at the data statements will show that only three rows on top and three on the bottom are completely zero. Some analysis of the row values will prove that the same is true about zero bits on the left and right sides of the columns. To allow our next program some freedom as to how much displacement an image can have without leaving trash on the screen, we will do the following:

```
10 DIM mshape%(1,15),zshape%(1,31)
15 INVOKE"/basic/bgraf.inv"
20 OPEN#1,".grafix"
25 GOSUB 1000
```

The difference in this program is that we introduce the zshape% array with twice as many rows as our mshape% array. This array is initialized in the routine at 1000, as follows:

```
1000 RESTORE
1010 FOR j=0 TO 15:FOR i=0 TO
     1:READ mshape%(i,j):NEXT:NEXT
1020 FOR j=0 TO 15:FOR i=0 TO
     1:zshape%(i,j+8)=mshape%(i,j):
     NEXT:NEXT
1030 RETURN
```

Please note that we use the same data statements from the last program, and, once the mshape% array is defined, we load the middle of the zshape% array with it. The offset in the rows between mshape% and zshape% gives eight extra blank rows at the top and bottom of zshape%, enough for the tricks we are about to pull.

Next, we initialize and declare a viewport, in which the visible part of our operations will occur:

```
35 PERFORM initgrafix
40 PERFORM grafixmode(%3,%1)
45 PERFORM pencolor(%4)
50 PERFORM fillcolor(%7):PERFORM fillport
55 PERFORM viewport(%40,%100,%15,
   %130)
60 PERFORM fillcolor(%13):PERFORM
   fillport
65 PERFORM grafixon
```

Note above that, although the graphics routines will let us draw anywhere on the screen (and anywhere off the screen from −32768 to 32767), the only visible effects will occur in the 40,130 to 100,15 window. Next, we get to the more elaborate draw section:

```
100 s%=16:t%=32:z%=0:f%=4
105 FOR k=0 TO 100
110   j=COS(k/5+2)*53+88
115   l=SIN(k/10)*30+58
120   m=SIN(k/10)*70+85
125   r%=s%*(r%=z%)
130   PERFORM moveto(%k+28,%j)
135   PERFORM drawimage(@zshape%(0,
      0),%f%,%r%,%z%,%s%,%t%)
140   PERFORM moveto(%k+14,%l)
145   PERFORM drawimage(@mshape%(0,
      0),%f%,%r%,%z%,%s%,%s%)
150   PERFORM moveto(%k,%m)
155   PERFORM drawimage(@zshape%(0,
      0),%f%,%r%,%z%,%s%,%t%)
160   NEXT k
200 GET a$:IF ASC(a$)<>27 THEN 105
```

The effect of the statements in lines 110 through 120 is to create different Y values for each of three creature images. Lines 130 through 155 then move to each unique location, draw the appropriate creature image, and go on. Note that the incremental movement of the first and third shapes is great enough that we need to use the zshape% version. Since zshape% takes longer to draw, mshape% is used where possible. Note also that the X positions are offset for each other, with the boundaries of the window responsible for clipping the images until they are within the display area. Running this program will produce images of bouncing creatures zipping through a boxlike window on the screen.

Like most programs, this one has a cleanup section:

```
300 PERFORM release:PERFORM
    release:PERFORM release
310 CLOSE:INVOKE
320 TEXT
330 END
```

There, have fun with that one!

By the way, if you want to edit your own creatures for the previous program, or for any to follow, you can use the February Shape Editor and make the following changes to the program above:

```
10 DIM shape%(7,15),mshape%(1,
   15),zshape%(1,31)
15 INVOKE"/basic/bgraf.inv","/
   basic/request.inv"
22 INPUT"Shape file name: ";file$
24 IF file$="" THEN 300
28 IF error THEN PRINT"Error number "
   error" in file '"file"' "GOTO 22
30 FOR j=0 TO 15:FOR i=0 TO 1:
   mshape%(i,j)=shape%(i,j):
```

```
     NEXT:NEXT
32 FOR j=0 TO 15:FOR i=0 TO 1:
     zshape%(i,j+8)=mshape%(i,j):
     NEXT:NEXT
1000 ON ERR GOTO 1030
1010 array$="shape%":OPEN#3,file$
1020 ftype=TYP(3)
1030 IF ftype=1 THEN READ#3;filtyp,
     ch,cw,sl:IF filtyp=1 THEN 1070
1040 OFF ERR:CLOSE#3:IF ftype=0 THEN
     DELETE file$
1050 error=1:REM Not a shape file
1060 RETURN
1070 READ#3,1:PERFORM filread(%3,
     @array$,%256,@ret%):OFF
     ERR:CLOSE#3
1080 IF ret%< >256 THEN error=2:
     RETURN:REM shape definition is
     invalid
1090 error=0:RETURN:REM Shape loaded
```

The routine from lines 1000 through 1090 can be used as a general-purpose shape-load routine. Note the addition of the Request.inv invokable to do the reading of the shape file.

Wrapping It All Up and Bouncing It off a Wall. The previous program proves that you can get the shapes to move through some rather elaborate paths. The next program puts together everything we have covered so far and borrows from an idea in the old *Applesoft Tutorial* manual—that of objects (the book called the little square blocks "balls") bouncing off the walls of a video room, not unlike the old *Pong* game. Since we have already shown how to endow our creations with X and Y movements, this should be easy:

```
10 DIM mshape%(1,15)
15 INVOKE"/basic/bgraf.inv"
20 OPEN#1,".grafix"
25 GOSUB 1000
```

The preceding lines comprise the usual warm-up. The routine at line 1000 is the usual and employs the data statements from the previous programs, to wit:

```
1000 RESTORE
1010 FOR i=0 TO 15:FOR j=0 TO 1:READ
     mshape%(j,i):NEXT:NEXT
1020 RETURN
```

Now on to setting up the screen:

```
30 PERFORM initgrafix
35 PERFORM grafixmode(%3,%1)
40 PERFORM fillcolor(%5):PERFORM
   fillport
45 PERFORM moveto(%45,%145):
   PERFORM pencolor(%0):PRINT#1;
   "Bug Box"
50 PERFORM viewport(%40,%99,%15,
   %130)
55 PERFORM fillcolor(%13):PERFORM
   fillport
60 PERFORM viewport( %60,%80,%62,
   %82)
65 PERFORM fillcolor(%4):PERFORM
   fillport
70 GOSUB 600:REM set color table
```

After clearing the screen to gray in line 40, we print the title "Bug Box" above a window of yellow created by lines 50 and 55. Then a dark green square is drawn in the middle of the box by lines 60 and 65, and we go to the routine at line 600 to set up our color-table scheme:

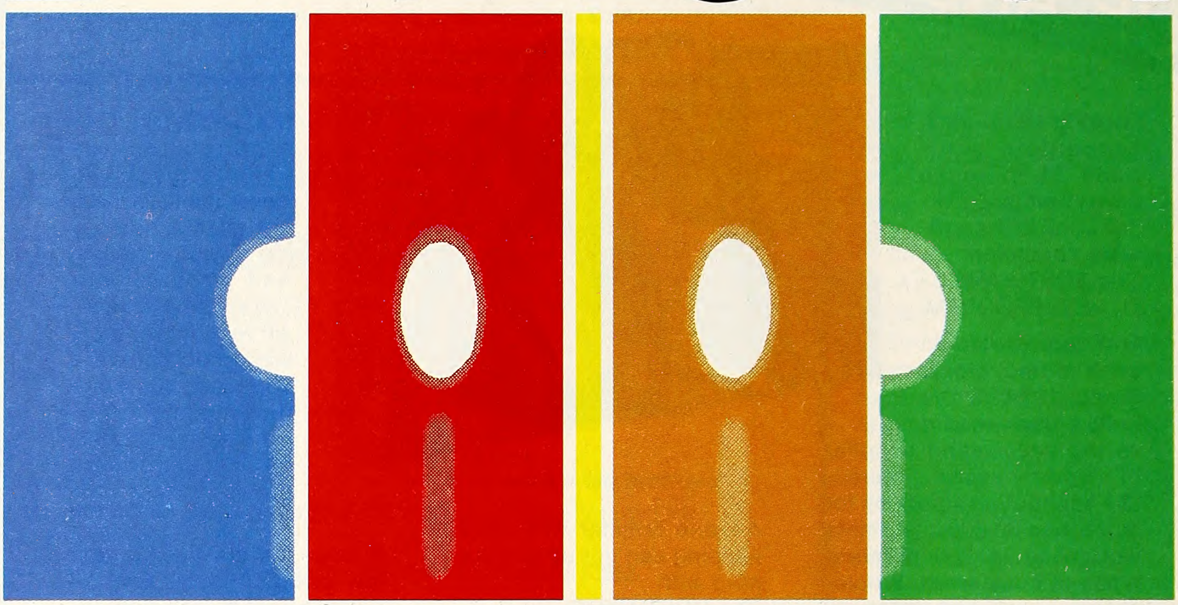
```
600 PERFORM setctab(%7,%2,%7)
610 PERFORM setctab(%2,%7,%7)
620 PERFORM setctab(%14,%2,%13)
630 PERFORM setctab(%15,%7,%13)
640 PERFORM setctab(%14,%13,%13)
645 PERFORM setctab(%14,%7,%7)
650 PERFORM setctab(%15,%13,%13)
655 PERFORM setctab(%15,%2,%2)
660 PERFORM setctab(%2,%4,%4)
665 PERFORM setctab(%7,%4,%4)
670 PERFORM setctab(%14,%4,%4)
675 PERFORM setctab(%15,%4,%4)
690 RETURN
```

As the routine above might tend to indicate, we will employ two creatures in this demonstration, one dark blue (color 2) and one light blue (7). The dark blue creature will use a fillcolor of aqua (14) and the light blue one will use a fillcolor of white (15). As you can see from lines 600 and 610, the dark blue creature will always appear to pass behind (be covered up by) the light blue creature, should their paths cross. Lines 620 through 655 ensure that the background (fill) colors will always translate to yellow (13), the background color of the box in which our creatures will live (and bounce). Study this carefully, until you are sure as to what is going on. Finally, lines 660 through 675 ensure that any movement in the area of the dark green (4) box will be hidden—that is, appear to go behind that object, since all colors drawn onto color 4 will result in color 4 on the screen.

Now that our colors are set, on with the show:

```
75 PERFORM viewport(%40,%99,%15,
   %130)
80 PERFORM grafixon
100 f%=4:s%=16:t%=32:z%=0
```

SOFT DISK



Magazette

TWO FOR THE PRICE OF ONE

Starting with the April issue of **Softdisk**, you'll get two disks of unprotected programs and Apple information for the price of one. You keep one disk and send back feedback on the other. We pay the return postage.

The second **Softdisk** disk will have room for electronic advertising from software companies. The second disk will also include a section for Soft Pals, **Softdisk** subscribers who correspond with each other through the electronic magazine.

A note to advertisers: **Softdisk** is now selling disk ad space initially for $3 per sector. If you hurry, you can reach 1,200 disk drives with a live demo or message for next to nothing. Call (318) 868-7247 for details.

Subscriptions to **Softdisk** cost $10 for the first issue and $5 per subsequent issue when the previous disk is returned.

Stop by the Softalk Publishing booth if you're attending the Boston Applefest May 13-15. You can check out **Softdisk** and meet the folks who bring you **Softalk, Softdisk,** and **Softline.**

Softdisk
3811 Saint Vincent, Department S5
Shreveport, LA 71108

Softdisk requires Applesoft and DOS 3.3

```
105 s1x=INT(RND(1)*40)+40:s1y=INT
    (RND(1)*115)+15
110 s2x=INT(RND(1)*40)+40:s2y=INT
    (RND(1)*115)+15
115 x1m=3:y1m=3:x2m=3:y2m=3
120 ON KBD GOTO 250
```

The lines above establish a viewport for operations and set up the initial random X and Y locations for our two creatures (lines 105 and 110). Line 115 sets the increment of movement in each direction for each creature, and line 120 establishes an on kbd jump to line 250 to get us out of the program. We now start the loop that will bounce our creatures off the walls of their tiny domain:

```
125 x1n=s1x+x1m:IF x1n<37 OR x1n>89
    THEN x1m=-x1m:GOTO 125
130 y1n=s1y+y1m:IF y1n<27 OR y1n>134
    THEN y1m=-y1m:GOTO 130
135 x2n=s2x+x2m:IF x2n<37 OR x2n>89
    THEN x2m=-x2m:GOTO 135
140 y2n=s2y+y2m:IF y2n<27 OR y2n>134
    THEN y2m=-y2m:GOTO 140
```

The lines above add the X and Y increments (or decrements, if negative) to their respective values and check to see if the results are within the ranges of the window. Note that the pen position is always the upper left-hand corner of the drawblock image, which explains the differences in the values from the viewport statement above. Now we get into the actual draw routines, based on the positions calculated above:

```
145 r%=s%*(r%=z%)
150 PERFORM moveto(%x1n,%y1n)
155 PERFORM pencolor(%2):PERFORM
    fillcolor(%14)
160 PERFORM drawimage(@mshape%
    (0,0),%f%,%r%,%z%,%s%,%s%)
165 PERFORM moveto(%x2n,%y2n)
170 PERFORM pencolor(%7):PERFORM
    fillcolor(%15)
175 PERFORM drawimage(@mshape%
    (0,0),%f%,%r%,%z%,%s%,%s%)
180 IF ABS(x1n-x2n)>10 OR ABS
    (y1n-y2n)>10 THEN 200
185 PERFORM moveto(%x1n,%y1n)
190 PERFORM pencolor(%2):PERFORM
    fillcolor(%14)
195 PERFORM drawimage(@mshape%
    (0,0),%f%,%r%,%z%,%s%,%s%)
200 s1x=x1n:s1y=y1n:s2x=x2n:s2y=y2n
205 GOTO 125
```

Notice again that we use r% to calculate the offset into the mshape% array to animate our creature. Lines 150 through 160 move to the calculated position of our first creature, set the colors, and draw the image. Lines 165 through 175 do the same for the second creature. Now comes something interesting. Creature number two has priority over creature number one, if they are in the same space. Therefore drawing over number one will destroy part of its image, which subsequent repositioning of the creatures cannot re-create. Lines 180 through 195 check for this possibility and redraw creature number one to fill in any missing parts before the next erase at the top of the loop. Try the program without these lines to see what happens. In either circumstance, line 200 sets the current positions of the creatures to the positions just used and starts over. The effect is that of the two beasts bouncing off the four walls in quite a regular fashion. Changing the values of x1m, x2m, y1m, and y2m will affect the type, length, and angle of bounce. Experiment and see what you like (random numbers are fun, too.)

A little more and we're finished:

```
250 OFF KBD
255 IF KBD=27 THEN 300
260 IF KBD=13 THEN POP:GOTO 40
265 ON KBD GOTO 250
270 RETURN
```

This on kbd routine allows termination (with escape (27)); if return is pressed, the program starts over with new random locations for the creatures. This is useful if you want to see how the color table affects the creatures when they cross over, but the random values put them on paths that don't cross. You just press return and start over with a new scenario.

Finally:

```
300 TEXT
310 PERFORM release:PERFORM
    release:PERFORM release
320 CLOSE:INVOKE
330 END
```

This does the visual cleanup.

A Cheerful Farewell. This month's article should give you plenty to work on. As you might imagine, the color table presents all sorts of possibilities that this missive could only hint at. Next time we'll wrap up our discussion of graphics and get on to more interesting doings with the Apple III. Until then. . . .

Lee McFadden makes learning to use your computer and its programs an easy, rewarding experience.

Mastering a new system or even a new program is not easy. It takes a lot of study, a lot of trial and error, a lot of back-and-forth between dense manuals and your computer. Wouldn't it all be so much easier if you had a knowledgeable friend sitting by your side?

Lee McFadden is that friend. He's been producing widely acclaimed teaching tapes for over ten years. He knows what information you need, and exactly when you need it. Team his skill with the know-how of technical experts, and you get a faster, easier way to master your computer and all its power.

Patented Audio Cassettes

Lee guides you patiently, step-by-step, while you're sitting at your keyboard. No connection between the computer and your cassette player is necessary. You learn by *doing*, instead of by reading. And even the tapes are special—a patented FlipTrack system lets you go at your own pace, covering as much or as little detail as you want.

Ideal for Training

Use the courses over and over again. Share them with friends, family, students and staff. Make "computer literacy" a reality in your local schools. Bring a new employee "up to speed" in just hours, instead of weeks. The fully indexed Guide included with each course is, by itself, an invaluable, on-going reference.

Moneyback Guarantee

You may order any of the FlipTrack courses from us on a 15-day "right of return" basis. Try a lesson or two yourself. Share them with others. If you're not delighted at how quickly and easily you begin to learn, simply return the program in its original condition for a full refund. No questions asked.

How to Order

Drop by your local dealer for a demonstration. Or Visa and MasterCard holders may add $2.50 shipping and handling to the prices, and order toll-free:

1-800-227-1617, Ext. 439

(In California, call 1-800-772-3545, Ext. 439.)

Choose from these rewarding courses:

How to Operate the Apple II® (specify "e" or "Plus")
3 audio cassettes & Operator's Guide $49.95

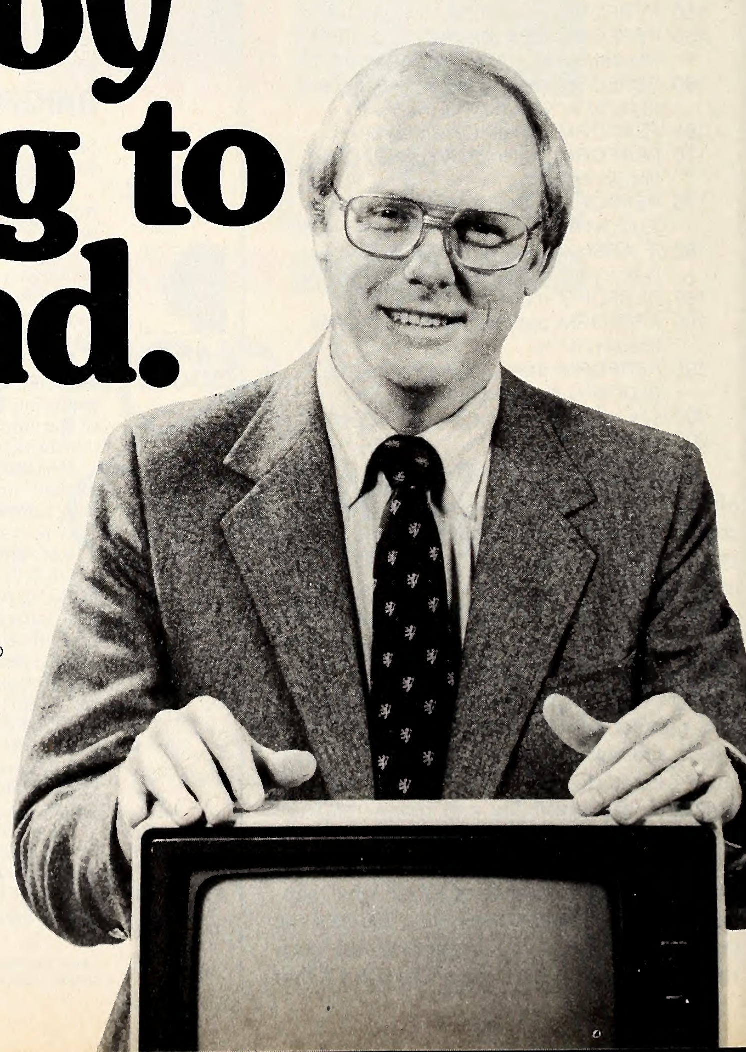
How to Operate the Apple///®
4 audio cassettes, 1 diskette & Operator's Guide.. $95.00

How to Operate Your Computer Under CP/M®
3 audio cassettes & Operator's Guide $49.95

How to Use VisiCalc®
4 audio cassettes & Lesson Summary $65.00

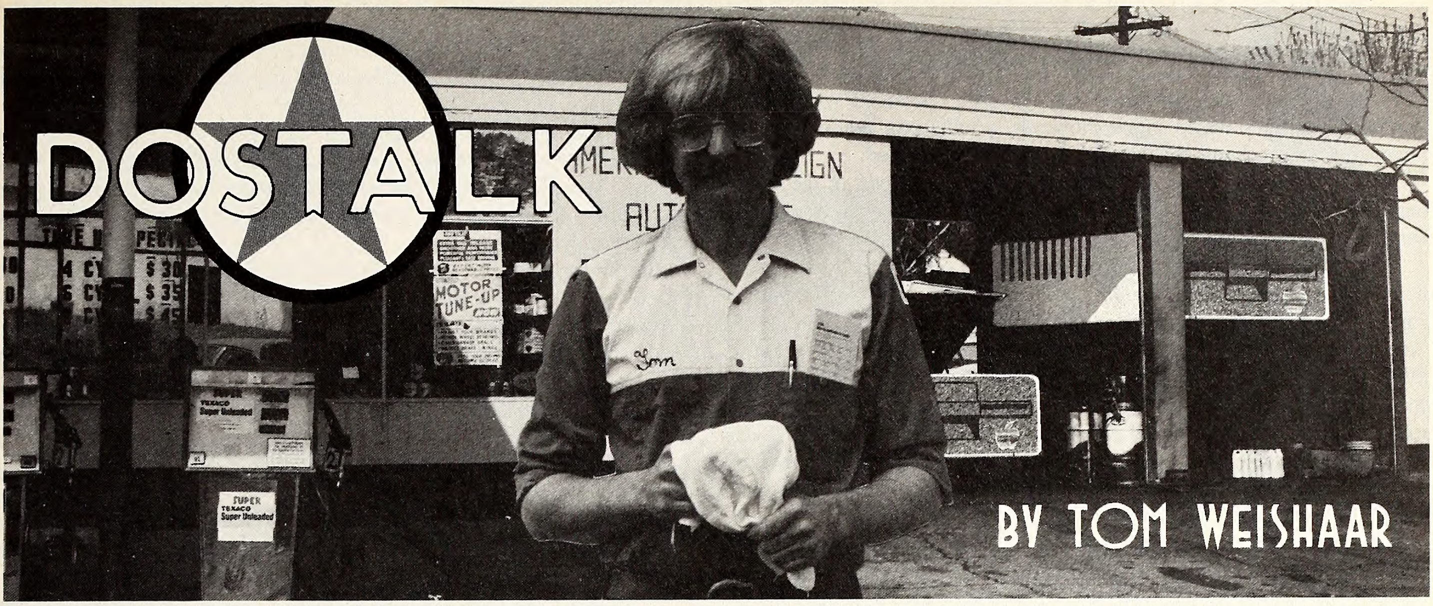
How to Use WordStar®
3 audio cassettes & User's Guide $49.95

Learn by listening to a friend.



FlipTrack® Learning Systems

A Division of Mosaic Media, Inc.
526 N. Main St., Dept. ST- B
Post Office Box 711
Glen Ellyn, IL 60137
312-790-1117



The main function of a computer's operating system is to get your programs and data into the computer and get the results back out to you. In computer jargon we're talking about input/output or I/O functions.

Let's look at Apple DOS as an operating system. We'll see why it is friendly to us common folks. You'll find out what those two noble DOS commands pr# and in# really do and how they do it. You'll also discover what's so special about control-D and why it doesn't seem to work properly sometimes. And you'll get an overview of a special part of the Apple II's memory, complete with maps.

About a Small Conspiracy. Operating systems are usually written in assembly language. They provide a computer with very elementary instructions for dealing with the outside world (you). Most operating systems were written for people who have sand (er, silicon) for brains.

CP/M, for example, the most widely used microcomputer operating system after Apple DOS, seems to have been written to make small computers as difficult to use as large ones. There are those who say the real meaning of CP/M is Conspiracy to Protect the Ministry. If that's true, it doesn't seem to be working—the power and influence of the computer priesthood is clearly dwindling.

The microprocessor in a computer like the Apple communicates with the outside world by looking at certain special memory locations designed to be built into the computer. In the Apple II, for example, byte number 49152 is the primary memory location for dealing with the Apple's keyboard. This byte is usually called KBD. You can create your own little minioperating system by means of the following program:

```
10  KBD = PEEK (49152)
20  IF KBD < 128 THEN 10
30  POKE 49168,0
40  PRINT KBD
50  IF KBD <> 155 THEN 10
60  END
```

How It Works. If the value at KBD is equal to or greater than 128, that means a key has been pressed. And until a key is pressed, this program is locked in a loop at lines 10 and 20. (We'll talk about line 30 in a moment.)

Line 40 prints the value we found. Line 50 tests to see if the key we pressed was the escape key (this key returns the value 155). If not, we continue the program; else we END.

When you run this program, nothing happens until you press a key. The value associated with that key then appears on-screen. Try typing in the ABCs. If the program works, go to the next paragraph. (*Global Program Line Editor*, and probably some other programs as well, plays tricks with KBD, so run the program without *GPLE.*)

Line 30 pokes a value into another special memory location, usually called the *keyboard strobe* or KBDSTRB. A reference to this location (it can be a peek or a poke; if a poke, it doesn't matter what value you poke there) resets KBD so that it's below 128 (referencing this location actually subtracts 128 from the value at KBD).

If you drop line 30 from the program you'll find that all the keys on your computer (except for escape) repeat automatically, whether you want them to or not. If you put the poke in line 30 before the test in line 20, your keyboard will operate somewhat erratically.

Switch Softly. The purpose of all this is to show you how a microcomputer deals with a simple device, such as a built-in keyboard, by peeking and poking at its own memory locations. The locations a computer uses in this manner are called *soft switches*, so named because they are electronic switches you can control with software.

Different computers use different memory locations for these soft switches. Thus you cannot use the Apple's operating system on an Atari computer, even though both use the same microprocessor known as the 6502. Part of the explanation for this is the fact that these two computers use different areas of memory for their soft switches.

The Apple II comes with a built-in operating system. Apple DOS is an extension of that built-in system, known as the Apple Monitor. The Monitor lives at the very highest addresses in your Apple's memory, from 63488 (in hexadecimal that's $F800) to 65535 ($FFFF).

The Monitor includes programs that sense when you touch your keyboard and that display characters on your screen. It also contains software routines for dealing with the built-in speaker, game controllers, and a cassette tape recorder.

The Monitor doesn't include any routines to operate devices connected through your Apple's slots, such as a printer or a modem (a device that connects your computer to a phone line). And it doesn't include much that relates to DOS. But the way the Monitor provides for these items is extremely clever and is responsible in large part for the Apple's friendliness.

Programs Written in Sand. The Monitor's cleverness is closely related to the design of a very special 4,096-byte area of Apple memory, from location 49152 ($C000) to location 53247 ($CFFF). The first 256 bytes of this area contain the soft switches (the keyboard locations we just used are at $C000 and $C010). The remaining 3,840 bytes hold assembly language programs that extend the Apple's operating system.

What is clever is that these programs are actually contained on your peripheral cards. Each card comes complete with its own operating system. Since the programs on these cards are permanently stored there, you needn't worry about loading special programs in from a disk or concern yourself with the technical aspects of interfacing your Apple with outside devices—everything required is included right on the card itself.

Adding a new peripheral (or removing one) always involves reprogramming the operating system. The Apple II is one of the few computers you can reprogram with the power off. You simply push a card in a slot or remove it from a slot.

Draw Me a Map. The accompanying figure will give you a better idea of how things are organized inside your Apple's memory. The leftmost bar of the figure represents the addresses in your Apple's memory, with the lowest addresses at the top of the bar and the highest at the bottom. Some of you may think the map is upside down; if you use the Monitor's memory dump much, however, you'll soon realize that it's correctly presented and that everyone else is out of step.

The center bar of the figure is a magnification of the part of the Apple memory dedicated to input and output. The center bar shows one-sixteenth of your Apple's memory addresses.

The rightmost bar is a further magnification of the first 256 bytes of the input/output area. It represents 1/256th of your Apple's memory. This area is where all the soft switches live.

Because the numbers are round and make a great deal more sense in hexadecimal, the map presents them that way. If you've never understood hexadecimal numbers, try it now. They are just like decimal numbers, except that each digit can range from 0 to 15 instead of 0 to 9. To represent the digits from 10 to 15, hexadecimal notation uses the letters A through F. Any number you see that begins with a $ is a hexadecimal number.

Let's start with the rightmost bar. Again, this bar represents the 256-byte memory area, from 49152 ($C000) to 49407 ($C0FF). All of your Apple's soft switches are in this area. The map shows that half of the soft-switch addresses are dedicated to built-in devices, such as the screen display, speaker, and game controllers; the other half are associated with your Apple's slots.

Look What They Did to the IIe. In the Apple II and II Plus, just 30 of the 128 addresses available for built-in soft switches were actually used. In the Apple IIe, the number of available addresses was nearly doubled, to 58. There are still many addresses left for built-ins in the IIe Plus.

Some of the new soft switches in the IIe control new features—a built-in alternate character set (the alternate set has inverse lower-case letters instead of flashing capitals) and provision for eighty-column display and 64K auxiliary memory. Other new switches give you the ability to find out how other switches are set. For example, in older Apples there's no way to tell from inside a program whether the screen is set to text or graphic display. On the IIe, activating one of the soft switches will tell you this.

Last month, in our discussion of the IIe DOS changes, we mentioned that two of these new soft switches are poked whenever DOS is booted. According to the new (thirty-dollar!) reference manual, one of them turns off the alternate character set and the other turns off eighty-column display.

The lower half of the rightmost bar in the figure shows the addresses of the soft switches associated with your Apple's slots. Each slot gets sixteen addresses to use for soft switches. Some peripheral cards use just a few of them, but your disk controller card, for example, uses all sixteen.

A language card, which expands the amount of memory available in your computer by 16K and slips into slot 0, uses eight of the sixteen locations in the $C080 to $C08F area to control its various functions. (The Apple IIe has no slot 0—$C080 to $C08F in a IIe controls its language card memory, which is built in.)

On the other hand, your printer, if it's plugged into slot 1, receives characters and is controlled with soft switches in the $C090 to $C09F area.

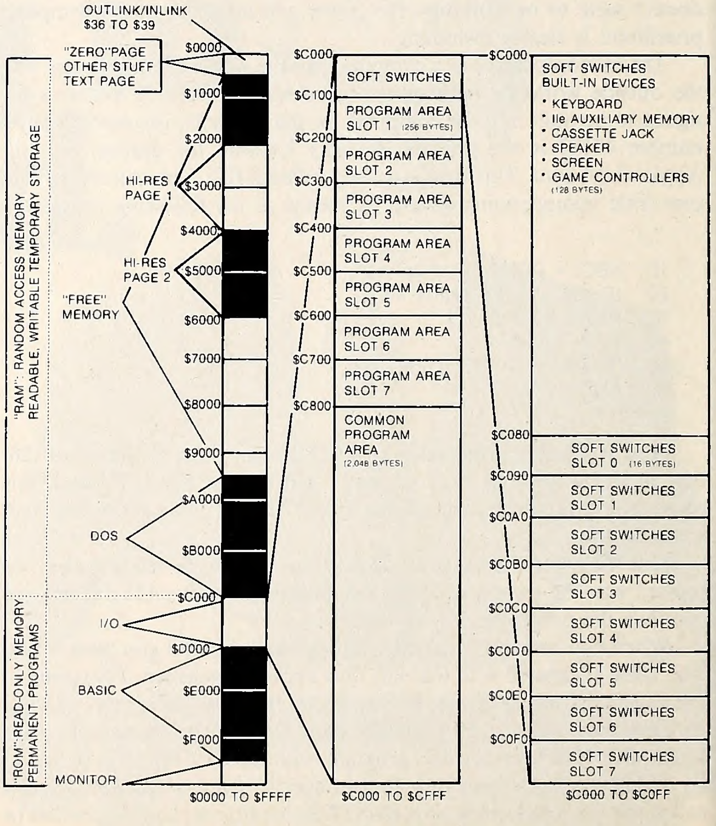
Extending the Monitor. The middle bar of the figure shows that these soft switches take up only one-sixteenth of the memory addresses in the input/output area. The rest of the addresses are used for the assembly language programs that extend your Apple's operating system. These are the programs built into your peripheral cards.

The programs on the cards, which tell your Apple how to use the sixteen soft switches associated with each card, start at memory location 49408 ($C100).

The 256 bytes at and following $C100 are owned by whatever card is in slot 1. The card in slot 2 gets everything from $C200 to $C2FF, and so on. Each slot has a dedicated 256-byte space, except slot 0.

In addition to the seven 256-byte program areas, there is a 2,048-byte space, from 51200 ($C800) to 53247 ($CFFF), that any card can use for expanded assembly language routines. This space is as large as the entire Monitor itself. While only one card can use this area at a time, it is easy for any card to grab control of the area and use it as needed. (The Apple

Apple II Memory Map



"$" INDICATES HEXADECIMAL NUMBERS

COMPATIBLE WITH:
APPLE II™, APPLE III™,
FRANKLIN™, ORANGE™, AND BASIS™

make your apple go ZOOM!

Supercharge your Apple to go 50 times faster.

For even faster speeds, combine Axlon's RAMDISK 320™ with S&H TDE Software.

Here's a whole new way to polish up your Apple II™ or Apple III™. RAMDISK 320™ from Axlon.

Thousands of users are discovering right now how the RAMDISK 320™ can boost both memory and access speeds of their Apples. The ultrafast RAMDISK is up to 50 times faster than standard floppy drives, and 10 times faster than hard disk drives. You can easily sort two full disks in 15 seconds.

Besides faster, error-free throughput, RAMDISK 320™ is designed to save wear and tear on your floppy drives. There are no moving parts.

RAMDISK 320™ has its own power supply, plus three-hour battery backup. RAMDISK 320™ draws no power from your Apple, and it retains data even when the Apple is turned off.

Your RAMDISK 320™ solid-state memory add-on system is fully compatible with Apple DOS 3.3, SOS, CP/M™, Apple Pascal 1.1 and Pascal 4.0. (In fact, we'll give you these drivers free with each RAMDISK 320™.)

Apple II and Apple III are trademarks of Apple Computer, Inc.
CP/M is a trademark of Digital Research.

RAMDISK 320™ can help you zoom through a wide variety of tasks in the real world — word processing, accounting, data base management, software development, educational and scientific data processing, or whenever speed is of the essence.

RAMDISK 320™ comes with a plug-in, slot-independent interface card. And it includes a specially designed operating program, plus software for diagnostics, and fast-load copy routines. Axlon offers RAMDISK 320™ with a one-year warranty.

Interested Distributors, Dealers, and OEMs call (408) 945-0500 for information on national sales, support, and pricing program or write Axlon, Inc., 70 Daggett Drive, San Jose, CA 95134.

For consumer information on ordering the RAMDISK 320™, call 800-227-6703. In Calif. 800-632-7979.

AXLON®

National Distributors: • **BYTE INDUSTRIES, INC.** Hayward, California (415) 783-8272 • **HIGH TECHNOLOGY** Florissant, Missouri (314) 838-6502 • **MARCEY INC.** Van Nuys, California (213) 994-7602 • **MICRO COMPUTER ELECTRONIC DIST.** Reading, Pennsylvania (215) 929-9484 • **B.A. PARGH CO.** Nashville, Tennessee (615) 361-3600 • **NATIONAL MICROWHOLESALE** Medford, Oregon (503) 773-1169 • **PMI MICROWHOLESALERS** Fairfield, New Jersey (201) 227-8411 • **SKU** Berkeley, California (415) 848-0802 • **VIDEO THEATRE** Rochester, New York (716) 621-2003 • **COMPUTRAC** New Orleans, Louisiana (504) 895-1474 • **MICRO D** Fountain Valley, California (714) 641-0205

As an Apple owner, you're entitled to a free trial subscription to Softalk. If you've never received Softalk, merely fill out this card, sign it, and mail it in. If you have received Softalk in the past, or are receiving it now, you can use this card to renew. We must have your serial number and signature to process new subscribers.

☐ I'm a new Apple owner.

☐ I'm using this card to renew my subscription. Please find enclosed $24 for a one-year subscription.

Apple Serial Number ____________________

Name ____________________

Street Address ____________________

City ____________ State ____________ Zip ________

Signature ____________________

SIGN UP FOR SOFTALK

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 673 NO. HOLLYWOOD, CA

POSTAGE WILL BE PAID BY ADDRESSEE

SOFTALK
P.O. Box 60
North Hollywood, California 91603

IIe uses this area too. This is where it hides the programs that handle eighty-column display. The IIe's self-diagnostic program is hidden in the address areas for the individual slots.)

The leftmost bar of the figure shows you how this input/output area fits into the whole Apple address scheme.

Never Golf at I/O Links. Near the top of this bar, in a memory area known as the zero page (bytes 0 [$00] to 256 [$FF]), there are two locations essential to the Apple operating system. They're known as the I/O links. One of these locations, Outlink, always contains the address of the current output routine; the other, Inlink, contains the current input routine.

Whenever a character is to be sent to the screen, the printer, or any other device, the character is stored inside the microprocessor itself, and control of the computer is passed to the subroutine Outlink points to.

If the character is supposed to go to a modem in slot 2, for example, Outlink will point to a routine in the area from $C200 to $C2FF. This routine will take the character and send it down your phone line using its own special magic. Then the routine will either pass control back to where it came from or jump to a routine in the Monitor that will put the character on your screen.

You control what is in Outlink and Inlink with the pr# and in# commands. A pr#6, for example, stores the address of the program space for slot 6 ($C600) in Outlink. The next time a character is sent out—usually it's the return at the end of the command—the call to Outlink will pass control of the computer to the card in slot 6.

More Noises. If the card there controls a printer, you'll hear the printer react to the return character by moving over to start a new line. But we all know that slot 6 is where you have your disk controller card. And we all know that pr#6 doesn't send anything to your disk drive.

The routines stored on your disk controller card tell your Apple how to boot a disk. We talked about booting last month. When you turn your computer over to your disk controller with a pr#6 (or whatever), it loads Apple DOS, an extension of your Apple's built-in operating system, into memory.

You might wonder why DOS is stored on the disk and loaded into valuable memory you could use, when it could be on the disk controller card itself. Why doesn't DOS use the program space at $C800? There are several reasons that DOS isn't permanently etched there. The biggest is that DOS is about 10,750 bytes long and only 2,048 bytes are available at $C800.

Once DOS is loaded into memory, it "cold-starts" itself. One of the many things that happen during the cold start is that DOS stores the addresses of a couple of its own routines in Outlink and Inlink.

Outlink is located at bytes 54 and 55 ($36 and $37); Inlink is at 56 and 57 ($38 and $39).

In assembly language, addresses are stored *backward* in two consecutive bytes. To find the address of the current Output routine you would type:

```
PRINT PEEK(55)*256+PEEK(54)
```

This tells you, in decimal, the two-byte value stored in Outlink. If you use 57 and 56 with your peeks, you'll get the address at Inlink.

If you do this now, you'll most likely find 40637 ($9EBD) at Outlink and 40577 ($9E81) at Inlink. These are the addresses of the DOS output and input routines.

If you have a printer, turn it on with a pr#whatever. Now look at Outlink again. The Outlink value will somehow appear on your printer, but the number still points to DOS. This effect has puzzled nearly everyone who has tried to determine how these links work.

The Captain Returns. If you read the article "DOS Be Nimble, DOS Be Quick" back in March, you might remember discovering that one of the main programs inside DOS, affectionately known as the Captain, is part spy: "This guy actually reroutes all your incoming and outgoing keystrokes and messages; so he gets to see them first. If he detects something that looks like it's for him, he tries to act on it."

So what does the Captain really do when he sees your pr# or in# command? Well, he does in fact change the address stored at Outlink for a microsecond or two but then immediately replaces it with his own address. He still wants to see all your mail.

So that he can trick you into thinking you are really in control of your own destiny, the Captain does take note of the address in Outlink before he replaces it with his own. He stores the true Outlink at locations 43603 and 43604 ($AA53 and $AA54) and the true Inlink at 43605 and 43606 ($AA55 and $AA56).

If your printer is still on, type:

```
PRINT PEEK(43604)*256+PEEK(43603)
```

If your printer is in slot 1, you will get an answer somewhere between 49410 ($C100) and 49663 ($C1FF). The answer usually is not right at $C100, where you would probably expect it to be. This is because most peripheral cards start with an initialization routine that determines which slot the card is in and sets up some variables for the routines on the card. As part of the initialization, the card will change the address in Outlink to point to its true output routine.

A Detailed Look at How It All Works. By now you may be starting to appreciate how cleverly this Outlink/Inlink system really works. Let's try another short program. It's very similar to the one we started with, but it uses the Apple's operating system instead of creating a miniop-erating system.

```
10  GET A$
20  PRINT A$;
30  IF A$<>CHR$(27) THEN 10
40  END
```

When you run this program, this is what happens:

* Basic sees your get command and calls on Inlink, which points to the DOS input routine. . . .
  * The DOS input routine immediately changes the link addresses to their true values.
  * It then checks for several special conditions.
  * Since none are true, the DOS input routine calls on Inlink, which now points to the Monitor input routine. . . .
    * The Monitor input routine puts a flashing cursor on your screen and starts watching KBD.
      * Much of your Apple's life is spent right here, waiting for someone to press a key.
        * You press X.
      * The cursor is removed from the screen.
    * The Monitor routine stores your X in the microprocessor and returns to . . .
  * the DOS input routine, which replaces the addresses in the I/O links with its own and returns to . . .
* Basic, which stores your X in the variable A$ and goes on to your next command,
* PRINT A$;, which gets your X from A$, puts it in the microprocessor, and calls on Outlink, which points to the DOS output routine. . . .
  * Like its kinsman, the DOS output routine immediately changes the link addresses to their true values.
  * It next looks carefully at the character stored in the microprocessor and checks for several special conditions, none of which turns out to be true.
  * It calls on Outlink, which now points to the Monitor's output routine. . . .
    * The Monitor puts the X on your screen, right where the cursor was,
    * and advances the "cursor position" (going to a new line and scrolling the screen if necessary), and returns to . . .
  * the DOS output routine, which replaces the addresses in the I/O links with its own, and returns to . . .
* Basic, which checks to see if there's an escape character in A$. (Note that the CHR$ code Applesoft wants for escape, 27, is exactly 128 less than the code we used before, 155.)
* Since A$ holds an X, not escape, the whole process begins again.

It's very hard to believe that all of this really happens in the split second after you press a key. Sit and watch it.

The important part happens when the Captain takes a look at the character in the microprocessor on the Outlink call. He ignores almost everything. But whenever he sees a return he lifts his head until the next character passes by. If that next character is a control-D, the Captain sits straight up and starts paying close attention; if it's not, he goes back to ignoring everything.

Take Two Control-Ds and Call Me in the Morning. Whenever you're writing a program and you include a DOS command that doesn't seem to be executed (but appears on your screen instead), the problem is that you aren't sending the control-D or aren't sending it immediately after a return. If the last print statement before the DOS command ended with a return-suppressing semicolon, your control-D will not be the first character after a return. The Captain will ignore it.

The surest cure for this was published here last June—it bears repeating. Use:

```
D$=CHR$(13)+CHR$(4)
```

CHR$(13) is the code for return, just as CHR$(4) is the code for control-D. (You should be careful when you close text files with this, however. It could give you an extra return you don't want at the end of your file.)

The control-D technique allows plain old Applesoft and Integer Basic to use DOS quite easily, and in exactly the same way. With some operating systems you have to use an "extended" version of Basic if you have a disk drive. With others, you get to learn a new set of disk commands every time you turn around. In some situations, even the commands you type on the keyboard can be different, depending on whether you are using the operating system directly or from within Basic.

With friendly old Apple DOS we don't need a "disk" Basic, and we have to learn just one set of DOS commands. Some of the computer priests you meet will tell you that the way Apple DOS hangs from the I/O links is a "fundamental design error" (or worse). In fact, it is quite simple, elegant, and friendly. No wonder Apple DOS is the most widely used disk operating system in the world.

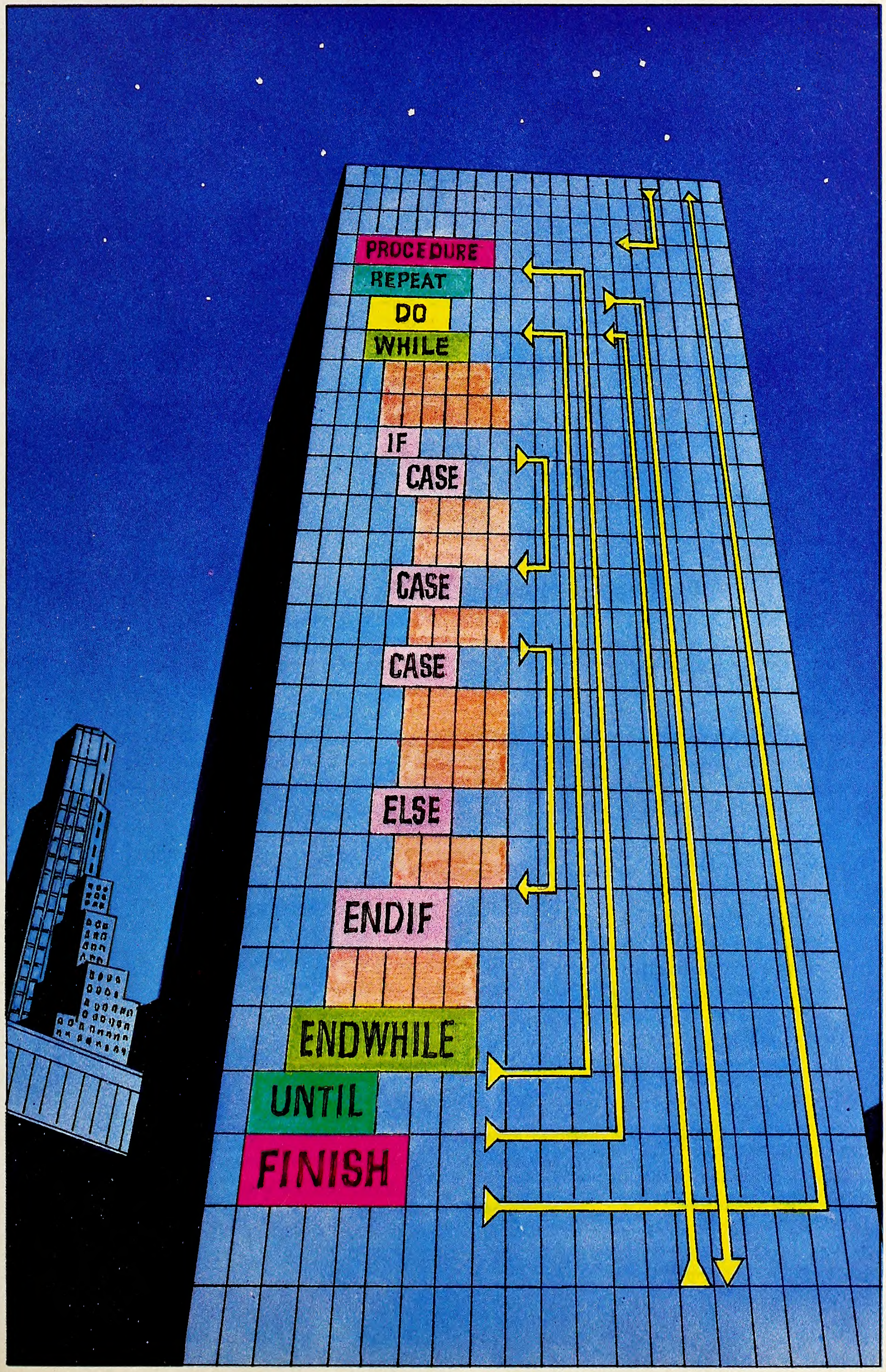
See you next month.

U-MICROCOMPUTERS PRESENTS
STRUCTURED BASIC
for Apple II + & IIe

'make it easy on yourself'

At last you can buy a true Structured Basic interpreter (this is not a pre-compiler) which has all the benefits of Pascal plus many other features, runs all existing Applesoft programs and uses DOS 3.3 as its operating system.

The Perfect Language for Novices and Professionals alike!



Take a look at these features, unique to Structured Basic:

Procedure Names Subroutines called by name as with Pascal. No line references needed.
Structured Commands REPEAT . . UNTIL, WHILE . . ENDWHILE, FOR . . NEXT, IF . . THEN . . ELSE . . ENDIF
Advanced CASE Statement IF (expl), (expn) . . Case 1 . . Case n . . Else . . End if
Disk Procedure Libraries Procedures not resident in the program are automatically read from disk and added to the program without interruption.
Procedure Overlaying Memory occupied by procedure called from disk can automatically be released for other uses when procedure is finished.
Local Variables Lists of variables may be declared for local use by each procedure making recursive programming possible and avoiding bugs caused by re-using variable names.
Disposal of Variables Unique 'RELEASE' command allows memory occupied by unwanted arrays to be reclaimed for use.
Additional Graphics Commands Graphmode, Page, Hires, Lores, Superimpose, Textmode, Mixed, Full, Fillwith, Screen.
Passed Parameters Variables may be passed to procedures as arguments.
Compatibility Virtually all Applesoft/DOS 3.3 programs will execute without modification.
Interpretive in nature All 32 new commands may be used in immediate mode, entered, listed and executed in programs just like other Basic Commands.
Improved Error Handling ONERR, ERRSTART, ERREND.

Available at $135.

Order Visa or UPS COD ($3. per card delivery & COD) Conn. residents add 7·5% sales tax

Structured Basic was written by Patrick Buckland of Island Computers Ltd I.O.W. and is distributed by U-Microcomputers Ltd.

Apple, trademark of Apple Computers Inc.

Software publishing – we market outstanding Apple software world-wide, including if necessary writing the manuals. Contact Roy Stringer if you have something that will meet our stringent criteria.

U-MICROCOMPUTERS

U-Microcomputers Ltd.,
Winstanley Industrial Estate,
Long Lane, Warrington,
Cheshire WA2 8PR, England.
Telephone 0925 54117
Telex 629279 UMICRO G

U-Microcomputers Inc.,
300 Broad Street, Stamford,
Connecticut 06091, USA.
Telephone 203 359 4236
& Toll Free (800) 243 2475
Telex 965999 O&S STD

01.200 2/83 RWP/Graphic Design.

INTRODUCING

ULTRATERM™

YOU'LL LOVE THE VIEW!

With UltraTerm, the revolutionary new card from Videx, you'll enjoy sweeping panoramas of spreadsheets that you've never seen before: 128 columns by 32 lines, 132 columns by 24 lines and even 160 columns by 24 lines. You'll revel in the scenics of a whole year of records stretching out across your screen.

You'll also delight in the new horizon of 80 columns by 48 lines—double the lines you normally have. So your word processing will reveal a "depth of character" never possible before!

Another breath-taking view of UltraTerm—it delivers absolutely flicker-free, state of the art display, with 8x12 character matrix giving you preposterously clear, readable characters. Not only will you see **more** characters on your screen (a whopping 4096 possible), but they'll also be larger and more readable than the characters you read every day in your newspaper! And you can differentiate those characters in several modes: normal (white on black), inverse (black on white), bright intensity and dim intensity.

Photo of actual-size characters on Apple Monitor III.

UltraTerm. Come on over and enjoy the view.
Suggested retail price: $379

Videx INC. TM

897 NW Grant Ave. • Corvallis, Oregon 97330
(503) 758-0521

UltraTerm features a built-in soft video switch and has complete firmware support for BASIC, Pascal and CP/M®. Use it with the Apple® II, Apple IIe, Apple III and Franklin.

Apple and the Apple logo are registered trademarks of Apple Computer, Inc.

CP/M is a registered trademark of Digital Research, Inc.

VENTURES WITH VISICALC

BY JOE SHELTON

VisiCalc's many features and functions are obvious; they're listed in the manual. Nevertheless, discovering how to use these capabilities to advantage is not always easy to do. This month, we'll continue to look at more effective methods of template design; our particular focus will be on templates to be used in forecasting.

The secret to the success of a forecasting template, as well as to the effectiveness of many other types of templates, is in the design. In a forecasting template especially, it's important that you set things up so that variables can be changed to reflect different assumptions.

Template Design. It can be helpful to think of a template as consisting of seven different areas. The first of these is the report area, which contains the printed and/or visual part of your model. The second is the calculation area. Not all templates require a separate calculation section (sometimes, calculations are handled in the report section), but in templates that involve many rows or columns of calculations, it's best to set one up. The third section is where variables are put. In many models, it works fine to include variables in the report section, but just as often it's better to have a separate variable section in which all variables can be seen and changed as needed. Having this consolidated variable section eliminates the need to search through the template every time you need to locate a variable.

The fourth area is the data area. This section may contain either the numeric data necessary for the model or consolidation data from another template. The fifth area is where graphic representations of data (using /F*) are stored. The sixth area is the antirecursive section, which is intended to eliminate the requirement to do multiple recalculations to achieve accuracy. Lastly, section seven is where check calculations to validate the report are done. Though few templates will use all seven different sections, almost every complex one will use two or more of them.

Forecasting—Scientific Guessing or Crystal Ball Magic? For some people, forecasting is natural. For most, though, forecasting is an art, and a black art at that. You might as well use a random number generator or draw numbers out of a hat. But there is hope. Look at any product and you can determine many variables that will affect its sales. In fact, if you think long enough, you'll probably find so many variables that the whole issue of forecasting will become even more confusing. The secret to good forecasting is to eliminate the inconsequential variables and concentrate on the important ones.

In order to forecast accurately, it's important not to rely on any single set of assumptions for your forecast. If you can do a second or even multiple forecasts based on different criteria, with reasonably similar results, then you can feel comfortable about the accuracy of your predictions.

If you want to forecast the sales for a particular product, the best place to begin is by looking at comparable or competing products. Depending on the maturity of the product, these may be a very helpful beginning reference. Another option is to look at your channel of distribution, the chain of people and organizations that buy the product before it reaches the final purchaser. Estimating what your potential sales are through this channel can give you an idea of what kind of total sales to expect overall. A third method would be to look at the potential purchasers of your product, both new purchasers and current owners of similar products that might "trade up" to yours. If the market is new or rapidly expanding, there probably won't be much history or other information on which to base your sales estimate. Then, as always, even your intuition can be an important variable.

Before proceeding, let's define and simplify our forecasting template. Let's say that we are forecasting sales of a personal computer and a number of accessory software applications for that product. Our forecast will determine the estimated computer sales based on a number of variables and software sales based on a percentage of computer system sales.

Let's further define the situation. You are the sales manager for the XX/Z Super Micro Computer and are responsible for providing the manufacturing department with a sales forecast. Manufacturing requires a one-year rolling forecast. You announced the XX/Z computer six months ago and have already sold 65,000 units. Two other companies are selling similar computers (the VI and the FUBAR), but nobody offers one with the features the XX/Z has. You have an estimated sales forecast for the VI, but the FUBAR is so new that forecasting is impossible.

You also have the responsibility for forecasting sales of three new and unannounced software products. You have access to sales histories of similar products on similar computers and you know that the sales of each application average out to a known percentage of computer system sales. Since these software products are about to be announced, you'll also have to account for some additional sales in the first few months to the installed base of computer owners.

What's the Forecast? There are two obvious ways to approach this forecasting task. The first is to estimate how many people in the installed base of computer owners will buy your computer and add to that your estimate of how many new computer owners will choose to buy your computer. Your second, or backup, forecast might be the unit sales for each of your sales outlets (retail stores, for example). If the two forecasts are close, you're home free; if they aren't, you must decide what to make of the discrepancies. And that can be a problem, so don't expect an answer here.

One more comment. This column usually emphasizes how important it is to keep your template simple. This template won't be particularly simple. We're going to experiment, and in the process we'll likely learn as much about *VisiCalc* as we do about forecasting.

Boot *VisiCalc* and let's begin. The first thing to think about is how we should set up the template.

We'll plan this template to have report, variable, and computational (work) sections so you can see how to lay it out. In designing your own template, it's a good idea to put your variables at the top, provided that you know all the variables required. You should also put the computation section at the top of the model; that way, the template won't require a second recalculation before the report "sees" the new variables or calculated values.

Usually, you know what you want the report section to look like, so you complete that section before starting anything else. So let's begin with the report section. Enter the information shown in figure 1. You'll notice that we've started a few rows from the top of the worksheet. We'll fill in the variable and calculation sections above that later.

Don't worry about the blank rows; we'll come back and fill them in. If you'd wanted to, you could have entered all of the information in columns A and B in sequential rows and then inserted additional rows as you needed them.

The I.B. in cell B20 stands for "installed base." Rows 29 through 31 will contain the forecasts for the applications software products.

WILL MAKE YOUR APPLE II THE JACK·OF·ALL·TRADES FOR $129.00.

Four applications in one.
The Incredible Jack combines the convenience of a personal filer with the power of a calc package. It handles most word processing tasks with ease. Toss in the ability to sort and print mailing labels, and you have a totally integrated, suprisingly easy to use package that does most of what you bought your Apple for.

Organize information your way.
The Incredible Jack lets you arrange your information in "records" you design yourself using the computer display. Each record may be as little as a mailing label or as large as 60 sheets of legal sized paper.

Word processing made easy.
With a few simple commands you can master in minutes, you can create letters, memos and reports. To help you edit, Jack lets you insert, delete, and copy portions of text all with automatic word wrap and flush right margins. The Jack does away with the mind boggling control codes and formatting options of other word processing packages.

Automatic decision making.
Jack even allows you to build decison making logic into your file. With English language rules and a powerful IF THEN ELSE function, you can instantly calculate complex discount tables, commission plans, contract terms, or tax rates.

See for yourself.
Try Jack. (You'll need an Apple II with 16K memory extension and 2 disks.) If you're not convinced it's the best investment you've made since you bought your Apple, send Jack back in good condition within 2 weeks for a full refund.

To order call: **(800) 645-4513.** For information: (516) 269-1120. To order by mail, enclose $129.00 plus $5.00 shipping and handling. (charge card # and exp. date) N.Y. residents add 7¼% tax, and send to: Business Solutions, Inc. 60 E. Main St., Kings Park, N.Y. 11754.

THE INCREDIBLE JACK. T.M.

business solutions

WILL MAKE YOUR APPLE II THE JACK·OF·ALL·TRADES

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
17															TOTAL YR
18			JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	SALE
19															
20	SALES TO I.B.														
21	NEW CUSTOMER SALES														
22			------												
23	FORECAST SALES														
24															
25															
26															
27															
28															
29	PRODUCT 1 SALES														
30	PRODUCT 2 SALES														
31	PRODUCT 3 SALES														
32															

Figure 1.

The next thing we want to do is to set up the initial variables. Enter the variables' titles and values as shown in figure 2 and we'll discuss them later.

A cursory explanation of each variable is in order. The 60,000 in C6 refers to the installed user base of the older version XX/Z computer that has bought new computers between introduction and the January sales month. C7 contains the percentage of the installed user base that will purchase another XX/Z computer, and C8 says that these purchases will be made over a seven-month period. Row 9 contains the estimated monthly history for the sales of the VI computer, while row 10 contains the estimate for the XX/Z computer sales compared to the VI sales. For example, C10 assumes that the XX/Z will sell 900 percent (or nine times) more than the VI.

Row 11 contains the maximum production capacity for our manufacturing facility. We'll use this figure to help calculate the actual manufacturing forecast as an additional section of our forecast. C12 contains the initial (or beginning) inventory for this manufacturing period, and C13, C14, and C15 contain the estimates for the percentages of computer system sales that each of the three applications products will sell.

The last area to complete is the computational section. This template requires a relatively small computational section. Comparing figures 2 and 3, you'll notice that these sections have been labeled VARIABLES and WORK AREA. As you design more elaborate templates, especially templates you don't use or change often, label all individual sections. That way, if you decide to alter a model later on, it will be easy to understand what each section is for. This becomes even more important as a model gets more complex. Go ahead and enter the titles in columns A and B. Then enter the numbers 1 through 12 in C1 through N1. The result should look like figure 3.

Now to start filling in the blanks. We want to calculate the total sales that we estimate we can make to the installed base in C2. Looking back at our variables, it's easy to conclude that the percentage of the installed base of users to purchase (C7) multiplied by the installed base (C6) should determine the total sales of our new computer to the installed base. There's a slight glitch, however. Cell C7 contains the value 10. Multiplying 10 times 60,000 will not give us the correct answer.

We could change the 10 in C7 to .1, but there's another solution that's much more friendly. It's usually better to refer to a percentage as an integer rather than as a decimal because doing so helps make the model easier to understand. The secret is to convert the integer value to a decimal during the final calculation. You turn 10 percent into a decimal by dividing by 100. The formula in C2 should be

```
+C6*(C7/100)
```

Now for a challenge. We want to apportion the sales to the installed base (in C2) into cells C3 through N3. Each number in row 1 indicates the equivalent month in row 18. See if you can determine how to let *VisiCalc* take the sales to the installed user base in C2 and spread that value through the number of months shown in C8. One additional stipulation! If the value in C8 is changed to reflect a different number of months, the sales to the installed base should be apportioned to reflect the change. In other words, the 18,000 sales to the installed base are now spread over the first seven months. If you decide that these sales would all occur during the next four months, you should be able to change the value in C8 to 4 and have 4,500 (18,000 divided by 4) displayed only in C3, C4, C5, and C6. Hint: You will need to use the values in row 1.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1			1	2	3	4	5	6	7	8	9	10	11	12
2	WORK	IB SLS												
3	AREA	IB/MTH												
4														

Figure 3.

The formula in cell C3 would be

```
@IF (C1<=C8,C2/C8,0)
```

In English, this reads: if the value in C1 is less than or equal to the value in C8, display the total installed base sales divided by the number of months; otherwise, display a zero. This formula compares the month represented by the values in row 1 to the total months, displaying the monthly sales calculation only if the number of the month is less than or equal to the months to apportion. Thus, any month whose number is greater than the value in C8 will not show any sales. Simple!

Back to the report part of our template. Row 20 will contain the sales to the installed base. You have already calculated those sales in row 3, so you need only repeat those values in row 20. Enter +C3 and /FI in C20 and replicate that value, using relative reference, into D20 through N20.

It's interesting to note a couple of things at this point. First, the values displayed in row 20 are nothing more than "copies" of the actual calculations in row 3. Sometimes you will want to keep the calculations separate from the display section. But if you wanted to, you could move or replicate row 3 to replace row 20, reentering the titles in A20 and B20.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
4														
5	VARIABLES													
6	INSTALLED BASE		60000											
7	% I.B. TO PURCHASE		10	%										
8	MONTHS TO APPORTION		7											
9	HISTORY VI SALES		300	500	600	700	700	900	1200	1200	1200	1200	1200	1200
10	% OF VI SALES		900	600	600	600	500	500	500	500	500	500	500	500
11	MAX PROD. CAPACITY		2000	2000	2000	3000	4000	5000	6000	7000	8000	9000	1000	10000
12	INITIAL INVENTORY		10000											
13	PRODUCT 1 % OF SLS		80	%										
14	PRODUCT 2% OF SLS		45	%										
15	PRODUCT 3% OF SLS		60	%										

Figure 2.

The new customer sales in row 21 are a multiple of the historical VI sales (row 9) multiplied times the estimated sales percentage of VI sales (row 10). This time we'll do our calculations in the report part of the template. Again we'll have to convert the percentages in row 10. The formula in C21 should be

```
+C9*(C10/100)
```

Be sure to enter /FI again and replicate the formula into cells D10 through N10.

Next we want the total forecast sales to be displayed in row 23. The entry in C23 is

```
/FI @SUM(C19...C22)
```

Again, replicate C23 into D23 through N23. We've now completed our initial forecast.

Now let's add another twist. A forecast is accurate only if manufacturing can provide enough product to meet that forecast. Suppose the forecast figure was greater than the number of computers our manufacturing facilities could produce. Many sales wouldn't get made.

In order to make our forecast template a more useful tool, we can modify it to take production capability into account. In B24 and C24, enter ESTIM. REMAIN INV (for estimated remaining inventory). This will be the amount of inventory remaining after we take the prior month's beginning inventory, add the current month's production, and subtract the forecast sales for the month. Remember that row 11 contains the manufacturing capability for each month in our forecast. The entry in C24 should be

```
/FI +C12+C11−C23
```

This gives us the ending inventory for January. We can't replicate this entry into the rest of the row without changing it slightly to reference the ending January inventory (C24). The entry to make in D24 is

```
/FI +C24+D11−D23
```

Replicate the formula in D24 into E24 through N24, using relative reference.

The first thing to notice is that the estimated remaining inventory becomes a negative value in March. That means that we're going to start the month of April selling computers we don't have. After all, not all of our manufacturing is accomplished the first day. In order to balance our sales forecast against production capability, it's important to remember that our manufacturing can only meet the production level shown in row 11. If our beginning inventory plus the manufacturing capability for the month is less than our forecast, some units aren't going to be sold. Thus, we must modify our original sales forecast to reflect manufacturing reality.

So here's another challenge. Can you devise a method to modify our original forecast that will accurately predict as many sales as possible but not forecast so many as to create a "negative" ending inventory? Enter PROD/FRCST SALES (a combination of forecast sales and production) in A24 and B24 and complete your sales forecast in row 24.

Simple, isn't it? Okay, so it isn't simple! There are probably two or three ways to accomplish our goal. The one we'll look at now is a little more complex than necessary, but it demonstrates how you can combine functions in fomulas.

In C25 enter

```
/FI @MIN(@IF(C24>=0,@MIN(C23,C12+C11),
@MAX(C23+C24,C11,C23)
```

D25 requires a slightly different formula because the beginning inventory is in the prior month's report section rather than in the variable section. The only difference, other than that the references reflect column D, is that C12 becomes C24:

```
D25: /FI @MIN(@IF(D24>=0,@MIN(D23,C24+D11),
@MAX(D23+D24,D11,D23)
```

You can replicate C26 with all references relative.

How would you change the formula to reflect the need for keeping a minimal inventory level of 500 units? How would you change it to reflect a different minimal inventory level for each month? You can solve these yourself; they should be easy.

The last section, rows 29 through 31, contains the forecasts for the software applications. We had determined that these were based upon a strict percentage of forecast computer sales. So the January forecast for product 1 sales should be January production forecast sales multiplied by our percentage (in C13). The entry should be

```
/FI +C25*(C13/100)
```

The entry in C30:

```
+C25*(C14/100)
```

The formula in C31 would use the variable in C15. Replicate these formulas across (using relative and no change references) and our template is complete.

If you wished to calculate revenue, you could do it by including your wholesale price for each of the products in the variable section and completing a separate section in the report area that displays the revenue for the computer and for each software application sold. To get total revenue, you'd simply sum these values.

We mentioned at the beginning that this was to be a one-year rolling forecast, meaning that each consecutive month the forecast is changed to reflect the activity of the following twelve months. For example, the forecast in March would include April through the following March, with April's forecast "rolling" to include May through April. To make yours a rolling forecast, you have only to delete the first forecast column (JAN in our template) and replicate another final column. You could move (/MOVE) the column and accomplish the same thing. In either case you must ensure that the variables are correct for the final month. Just moving the column might bring an incorrect production capability and thus throw off the complete forecast from that month on.

And there you have it—a comprehensive forecasting template that could be useful in many forecasting applications. Figure 4 shows the complete template. If it isn't exactly what you want, it's a simple matter to modify it. After all, you've just completed it so you know how it works.

Just remember—no matter how precise your forecast and how well you have thought out all the variables, forecasting is just what the dictionary says: "predicting results." Unless you are very lucky, very good, or are predicting the sales of a product whose sales never change, forecasting is just a method of guessing. And occasionally everybody guesses incorrectly; some more occasionally than others!

VisiCalc for the Apple IIe. Apple Computer recently announced the Apple IIe, and VisiCorp followed suit with an Apple IIe version of *VisiCalc*. It is much the same as the current *VisiCalc* for the Apple II, but it has been enhanced to be able to use the Apple IIe's additional features. For example, it displays upper and lower case, takes advantage of the four cursor keys (no need for the space bar to change direction any more), and uses the backspace key for deleting characters (no more using the escape key to delete). In addition, with the accessory memory expansion board and eighty-column card, the Apple IIe now has an eighty-column screen and 95K of template workspace. If you have an Apple II and need more workspace, you may want to check out the Apple IIe with *VisiCalc*. ◼

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1			1	2	3	4	5	6	7	8	9	10	11	12
2	WORK	IB SLS	18000											
3	AREA	IB/MTH	2571	2571	2571	2571	2571	2571	2571	0	0	0	0	0
4														
5	VARIABLES													
6	INSTALLED BASE		60000											
7	% I.B. TO PURCHASE		10	%										
8	MONTHS TO APPORTION		7											
9	HISTORY VI SALES		300	500	600	700	700	900	1200	1200	1200	1200	1200	1200
10	% OF VI SALES		900	600	600	600	500	500	500	500	500	500	500	500
11	MAX PROD. CAPACITY		2000	2000	2000	3000	4000	5000	6000	7000	8000	9000	1000	10000
12	INITIAL INVENTORY		10000											
13	PRODUCT 1 % OF SLS		80	%										
14	PRODUCT 2 % OF SLS		45	%										
15	PRODUCT 3 % OF SLS		60	%										
16														
17														
18			JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
19														
20	SALES TO I.B.		2571	2571	2571	2571	2571	2571	2571	0	0	0	0	0
21	NEW CUSTOMER SALES		2700	3000	3600	4200	3500	4500	6000	6000	6000	6000	6000	6000
22														
23	FORECAST SALES		5271	5571	6171	6771	6071	7071	8571	6000	6000	6000	6000	6000
24	ESTIM. REMAIN INV		6729	3157	−1014	−4786	−6857	−8929	−11500	−10500	−8500	−5500	−1500	2500
25	PROD/FRCST SALES		5271	5571	5157	3000	4000	5000	6000	6000	6000	6000	6000	6000
26														
27														
28														
29	PRODUCT 1 SALES		4217	4457	4126	2400	3200	4000	4800	4800	4800	4800	4800	4800
30	PRODUCT 2 SALES		2372	2507	2321	1350	1800	2250	2700	2700	2700	2700	2700	2700
31	PRODUCT 3 SALES		3163	3343	3094	1800	2400	3000	3600	3600	3600	3600	3600	3600

Figure 4.

THE BASIC Solution

By Wm. V. R. Smith

This month's Basic Solution is the third and final part of our database project. Remember—and those of you tuning in for the first time please note—the two program segments presented this month won't work unless they are added to the rest of the database program, which has appeared in two previous parts in the last two installments of this column.

These final routines comprise the print system for the database. When you select the print option from the main menu, the screen will display a set of default parameters. These are the specifications for the relative placement of the five fields within each record. Following the default settings, the program will place one field on each line, starting on the first position of the line. The final parameter will tell the program how many blank lines should be printed between records.

If you want to accept the default settings, just answer Y to the question of whether they are correct, turn on the printer and line up the paper, and hit return when you're ready to begin printing. The program will dump all records to the printer.

If you want to change one or more of the settings, answer N to the question. The settings for the first field will be displayed in inverse and a prompt line will appear at the bottom of the screen asking for the new value for the line on which the field should appear. Hit return to accept the default or enter a new setting. The next prompt will ask for the location on the line at which to start the field. Follow the same procedure for this input. If you only want to change one field's location—say, move the miscellaneous field to appear on the same line as the phone number—hit return until the prompt and the inversed settings indicate the field you want to change.

The printer is assumed to be in slot 1 of your Apple. If your printer is in another slot, change line 6040 to do a pr# to the correct slot.

With these sections added to the database, you will have a fully integrated database system. The program takes up less than 7K of program memory, so there is plenty of room available for any addition you might want to make. If you come up with any valuable subroutines to add to the program, please send them to the Basic Solution care of *Softalk* so that we can share them with other readers.

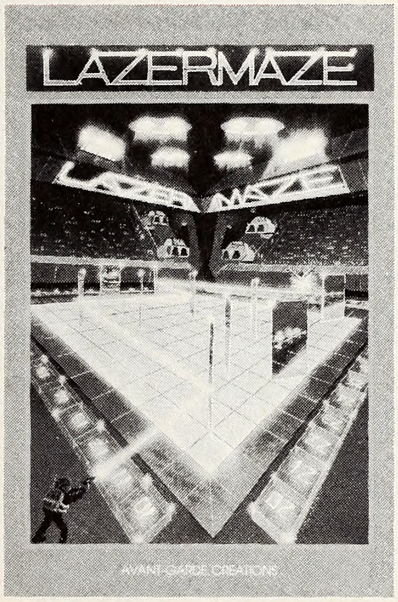
```
2800 REM ****************************
2801 REM * PRINT DATA
2802 REM ****************************
2805 IF FL = 1 THEN 2820
2806 FL = 1
2810 FOR X = 1 TO NH:L(X) = X:C(X) = 1:
     NEXT
2815 ML = NH
2820 GOSUB 5000
2825 HOME : VTAB 10: PRINT "INSERT
     PAPER AND PRESS RETURN"
2826 PRINT "PRESS ESC TO RETURN TO
     MENU"
2827 GET A$: IF A$ = CHR$ (27) THEN
     RETURN
2828 PRINT
2830 FOR RR = 1 TO NAR - 1
2835 GOSUB 4000
2840 GOSUB 6000
2850 NEXT RR
2890 RETURN
5000 REM ****************************
5001 REM * PRINT FORMAT
5002 REM ****************************
5010 HOME
5020 HTAB 10: PRINT "PRINT SYSTEM"
5030 PRINT
5035 HT = 0
5040 GOSUB 5500
5050 VTAB 20: CALL - 868
5060 INPUT "CORRECT ? ";A$
5070 IF A$ = "Y" THEN RETURN
5075 ML = 0
5080 FOR Z = 1 TO NH
5085 HT = Z: GOSUB 5500
5090 VTAB 22: HTAB 1
5100 PRINT "INPUT LINE FOR ";H$(Z);" ";:
     INPUT A$
5110 IF LEN (A$) = 0 THEN 5140
5120 V = VAL (A$): IF V < 0 THEN V = 0
5130 L(Z) = V
5140 GOSUB 5500
5142 IF ML < L(Z) THEN ML = L(Z)
5145 IF L(Z) = 0 THEN 5200
5150 PRINT "INPUT COLUMN FOR "
     ;H$(Z);" ";: INPUT A$
5160 IF LEN (A$) = 0 THEN 5190
5170 V = VAL (A$): IF V < 0 THEN V = 0
5180 C(Z) = V
5190 GOSUB 5500
5200 NEXT Z
5210 HT = 0: GOSUB 5500
5220 PRINT "NUMBER OF BLANK LINES ";:
     INPUT A$
5230 IF LEN (A$) = 0 THEN 5260
5240 V = VAL (A$): IF V < 0 THEN V = 0
5250 BL = V
5260 GOTO 5000
5500 REM * FORMAT VIEW
5510 VTAB 3
5515 PRINT "HEADER NAME   LINE
     COLUMN"
5517 PRINT "-----------------------------------"
5520 FOR X = 1 TO NH
5525 IF X = HT THEN INVERSE
5530 PRINT H$(X);: HTAB 20: PRINT L(X);:
     HTAB 30: PRINT C(X)
5535 NORMAL
5540 NEXT
5545 PRINT : PRINT "BLANK LINES ";BL
5550 VTAB 22: CALL - 868
5560 RETURN
6000 REM ****************************
6001 REM * PRINT IT
6002 REM ****************************
6040 PRINT : PRINT CHR$ (4);"PR#1"
6050 FOR L = 1 TO ML
6060 O$ = "
     ": REM EIGHTY SPACES
6070 FOR X = 1 TO NH
6080 IF L(X) <> L THEN 6110
6100 O$ = LEFT$ (O$,C(X)) + A$(X) +
     RIGHT$ (O$,(80 - C(X)))
6105 O$ = LEFT$ (O$,80)
6110 NEXT
6120 PRINT O$
6130 NEXT
6135 IF BL = 0 THEN 6150
6140 FOR X = 1 TO BL: PRINT : NEXT
6150 PRINT CHR$ (4);"PR#0"
6200 RETURN
```

3 TIMES THE EXCITEMENT

FROM

AVANT-GARDE CREATIONS, INC.

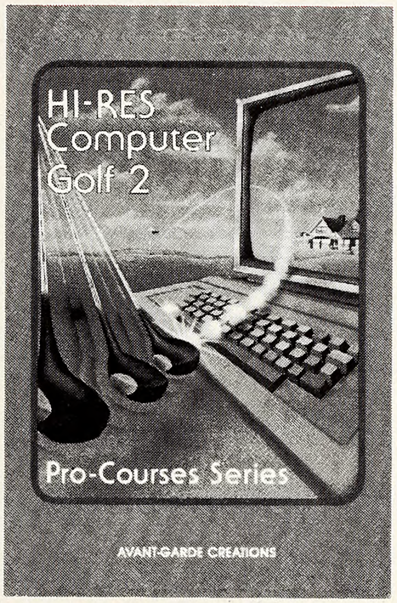
LAZER MAZE



YOU AGAINST THE CLOCK

You are suddenly thrust into a huge arena filled with mirrors. Then you're given a laser pistol and told to keep a sharp eye on the numerous reflective shields. Somewhere else in the arena is your opponent. You're pitted against time as you calculate your shot. Once fired, the colorful laser beam bounces from one mirror to the next. Did you aim correctly? Were you fast enough? Or did you only give away your position, opening you to attack from your opponent -- LAZER MAZE. Available for your Apple™, Atari™, Vic 20™, Commodore™, IBM PC™ or TI-99™ computer. **$29.95**

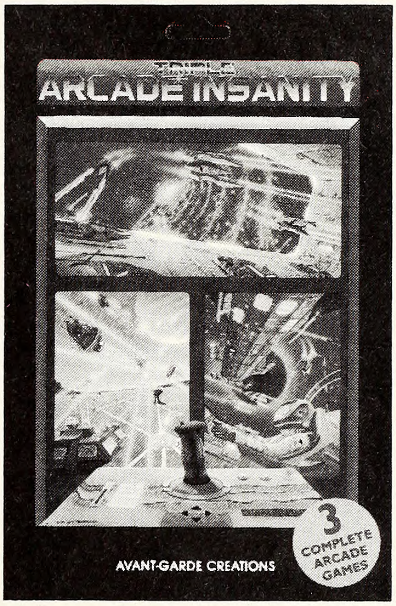
HI-RES COMPUTER GOLF 2



BETTER THAN THE REAL THING

Who needs rain and sore feet? This golf game allows you to sit back and enjoy all the excitement and skill of pro-courses while actually *improving your real golf game*. The three courses are taken from existing sites. The fairways are true-to-scale. There are lakes, trees, sand traps and roughs. The greens are even contoured! And there is a variable wind factor. You select your clubs just as you would in a real game and you control your own swing. (Auto-swing selection can also be used.) Watch for additional Pro Courses disks to expand this program. So why just stand there? Get seated and start to enjoy! HI-RES COMPUTER GOLF 2 -- even better than before. Available for your Apple™. **NEW $34.95**

TRIPLE ARCADE INSANITY



3 COMPLETE GAMES

Alien Onslaught, Galactic Conquerors and Andromeda Blitz . . . each more difficult than the last. You must destroy the aliens at all levels but beware, it can take up to six direct hits to eliminate them. Fire in multiple directions as you control your craft through space. Joystick, paddles, or keyboard controls allow you to command at will. Can you handle this trilogy of high speed, hi-res arcade games? TRIPLE ARCADE INSANITY. Available now for your Apple™. **NEW $29.95**

NOW AVAILABLE AT YOUR LOCAL COMPUTER STORE

and by calling AVANT-GARDE CREATIONS, INC.

at (503) 345-3043

★ ASK YOUR DEALER TO SHOW YOU THE ENTIRE AVANT-GARDE CATALOG ★

AVANT-GARDE CREATIONS, INC., P.O. Box 30160, Eugene, Oregon 97403

STATE OF THE ART

from page 88

clude print options, nor is it structured around concepts of traditional notation for developing compositions. All you have to do is play what you feel and the sequencer will store it or play it back for you the way a tape recorder does.

The audio quality and sound range of the system is determined in great part by the Mountain hardware. In some cases low-pitched sounds can be problematic, causing distortion; and they can sound strange with certain parameter settings because of unbalanced frequency modulation. Also, some late-model Apple II Pluses have an internal interference problem with the hardware. This can be remedied by installing a noise-reduction capacitor available at no extra charge from Mountain Computer.

Alpha Synthetic Music. Another system that is designed around and uses the Mountain hardware is the alphaSyntauri digital synthesizer from Syntauri. The alpha lets you control many of the same aspects of additive synthesis that the Passport system does, and it too offers a multitrack sequencer, but the approach taken by the Syntauri software is distinctive. Let's acknowledge the general similarities between these two systems and then concentrate on those features of the alphaSyntauri that set it apart from other systems.

The organ-style keyboard that alphaSyntauri provides with the system is polyphonic to a maximum of eight voices, covers a five-octave range, and is velocity-sensitive, causing keys struck faster to sound louder. You can control the degree of sensitivity with the software.

The keyboard interfaces with the Apple through a single circuit board that can reside in any free slot. Two foot switches connect to the keyboard via standard phone jacks. These control sustain and portamento effects; portamento is either on or off and is limited to a preset, unprogrammable rate of modulation.

Either of two software packages, *AlphaPlus* or *Metatrak*, can be used to run the system. The *AlphaPlus* package consists of twelve preset files of ten sounds each that can be played on the keyboard, additive synthesis techniques for creating and manipulating waveforms and their parameters, and a sequencer for recording up to 2,000 notes layered to eight multiple tracks.

When the keyboard is being played in live mode, the monitor screen is split into the text window and a pitch window. The text window gives you information about some sound parameters of the preset you're currently playing. These parameters are easily modifiable in real time.

The *AlphaPlus* approach uses two oscillators in defining each sound, resulting in the system's eight-voice limit. One oscillator is referred to as the primary channel, the other as the percussive channel. The text window contains a description for each channel of the sound's overall amplitude envelope in numerically defined unit increments between 0 and 255, as well as other data, such as variable vibrato rates that can be controlled using a standard set of paddles. The pitch window is nothing more than a dark field on which flickering colored squares reflect the pitches currently being played. Though this feature imparts no information to the user that is useful in composing music or synthesizing sounds, some may find it visually entertaining in moments of distraction.

The record/playback mode allows you to sequence eight simultaneously playing layers or tracks with an optional metronome to keep the tempo for you. During playback you can vary the overall speed of a piece without altering its original pitches, or create passages that repeatedly loop (a feature the manual misleadingly refers to as "echo"). You also have the ability to stop playback at any time and start recording again from that point, a technique commonly called "punching in." During playback, the keyboard is still fully active so you can also play along. With the Album file feature, you can take two or more sequences you've recorded, combine them together under a single file name, then play them back consecutively like a record album.

The color of a chord as portrayed by the alphaSyntauri system, seen in the pitch window.

AlphaPlus uses additive synthesis to program waveforms in much the same way that the Passport system does. One approach *AlphaPlus* offers is *Quickwave*, which uses a bar graph to plot the fundamental and the harmonics by amplitude for the two oscillators that define a sound. By depressing keys on the music keyboard, you can hear the sounds you devise in this manner as you are building them.

Another optional approach to wave formation, called *Wave*, doesn't allow you to hear what you're developing in real time. Going this route, a waveform you've worked on must be saved first, then reloaded to check it audibly. The inconvenience is a tradeoff for the precision and fine-tuning of waves that this approach gives you. With *Quickwave*, the har-

More Apple II owners choose Hayes Micromodem II than any other modem in the world. Compare these features before you buy. You should. It's your money. Thousands of other Apple II owners have already compared, considered, and are now communicating — all over the U.S.A. — with Micromodem II. The best modem for the Apple II. The most modem for your money.

A complete data communication system. Micromodem II is not "base priced" plus necessary "options." It's a complete, high-performance data communication system. The printed circuit board fits — quickly and easily — into your Apple II, eliminating the need for a serial interface card. And the Microcoupler™ (included) connects the Apple II directly to a standard modular telephone jack. Auto-dial and -answer features are built in. Operation can be full or half duplex, with a transmission rate of 300 bps. And it's Bell 103 compatible and FCC approved.

Now there's Hayes Terminal Program, too! Developed by Hayes specifically for Micromodem II, this new Terminal Program allows you to access all the great features of your modem in a matter of seconds.

With it, you can use your CP/M,® DOS 3.3 or Pascal formatted disks to create, send, receive, list and delete files. Hayes Terminal Program is a complete, stand-alone disk.

And because it's menu driven, you can choose from a wide variety of options to set your communication parameters — as well as change hardware configuration — directly from the keyboard. It even allows you to generate ASCII characters that are normally not available from Apple keyboards, further extending your capabilities. Incoming data can be printed (on serial or parallel printers) as it's displayed on your screen.

Micromodem II is available with or without the Terminal Program. Buy your modem by itself, or optionally packaged with the Terminal Program disk and user manual at extra cost. The software is also sold separately, for those who already own a Micromodem II.

If you're ready to communicate with other computers, to access information utilities, time-sharing systems, or use bulletin boards, then you're ready for Micromodem II. Come on. Compare. Consider. Then buy.

Micromodem II is already the best-selling modem for the Apple II. And Hayes' new Terminal Program makes it better than ever. Available at computer stores nationwide.

Hayes

Hayes Microcomputer Products, Inc.
5835 Peachtree Corners East
Norcross, Georgia 30092 (404) 449-8791

monic spectrum allowed extends as far as the sixteenth harmonic; with *Wave* you can specify up to 255 harmonics for a sound, which can result in some unique emanations from your speakers. A program that provides a graphic analysis of any wave you load (no audio feedback is given) can be found on disk too.

User-programmable effects modifications for amplitude, chorus effect, pitch bend, and pitch scan are available in the *AlphaPlus*, along with the ability to detune the keyboard to units of one-thirty-second of a tone or alter its standard equal-tempered scaling.

Metatrak, another package available separately, expands on the features of *AlphaPlus* with the exception of the waveform creation programs that *Metatrak* doesn't contain. With *Metatrak* the sequencer can record a total of sixteen simultaneous music passages, to a limit of approximately three thousand notes, that can then be manipulated or altered the way tracks can in *AlphaPlus*.

Like *AlphaPlus*, *Metatrak* is capable of looping segments and punching into individual tracks. Beyond that, it offers the ability to fast-forward a sequence, erase an unwanted track with a single keystroke, and modify the tempo of playback tracks while adding a new track at another speed. The continual merging of sequencer tracks required in the Passport system is unnecessary here, but a mix-down/playback step is provided for making final adjustments before saving a finished recording. Certain DOS commands can be used directly from the program, enabling you to delete or rename files, and to lock valuable ones worth protecting.

Unlike *AlphaPlus*, *Metatrak* allows you to split the keyboard into from two to eight segments and program the splits to occur anywhere you choose. What's more, every segment can be assigned to play a different preset sound, and in live mode these assignments can be changed in real time. All the specifications of a chosen split configuration can also be saved to disk for future use. Certain special-effects modifications are another departure unique to *Metatrak*. For example, there is timbre scan, an audible rate adjustable scan of an entire preset master's waveforms that can assume any envelope characteristics you choose. Pitch sweep and keyboard-following vibrato are also available.

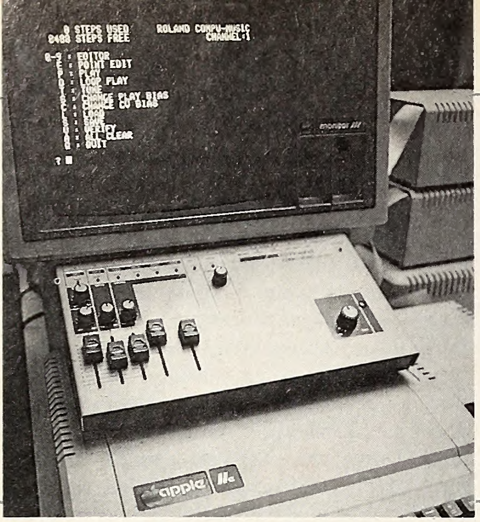
Both *AlphaPlus* and *Metatrak* do a good job of handling situations in which a wrong input has been given. The manuals for both systems are well written and easy to understand, though not indexed. *AlphaPlus* includes a helpful reference card showing all commands, along with a listing describing the preset masters that come with the program on disk. All preset masters created with *AlphaPlus* can be loaded and used with *Metatrak*. However, files of compositions created with *AlphaPlus* are not compatible with and cannot be used with *Metatrak* and vice versa. Neither program seems to support print options of any kind.

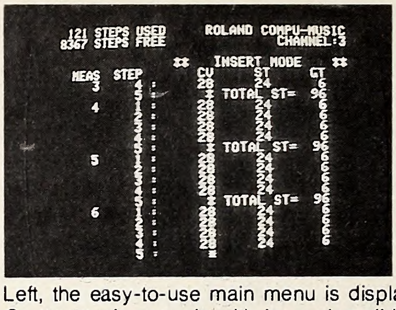
Because of the way the *Metatrak* file buffer is structured, it is possible to overwrite material when using the sequencer to record. Ample safety checks and error messages exist that will alert you in time before recording if a problem of this sort is imminent. Though the articulation and expression of musical phrases is certainly enhanced with the velocity-sensitive keyboard, the keyboard reaction time when fast staccato notes are being played is inconsistent and the notes come out with audibly variable intensities.

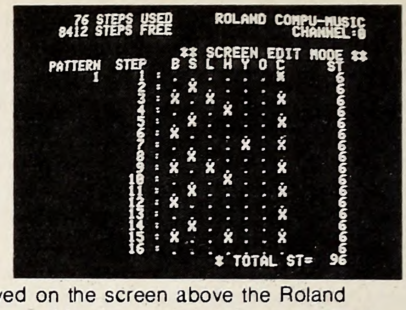
The new *Metatrak II* includes the ability to sync the output of the system to reel-to-reel tape recorders and rhythm machines. AlphaSyntauri offers some additional useful utilities that come as separate packages. *Draw Wave* allows you to draw waveforms using a set of paddles; with *Auto Pulse* you can represent pulse waves with duty cycles between 0 and 50 percent with precision; and with the *B-3 Wavemaker* you can duplicate almost any setting on a Hammond B-3 organ. A series of interactive music theory and ear-training programs called *MusicMaster* is also available for use with the alphaSyntauri system.

Compumusic Console Controller. *Compumusic* from Roland Corporation is not designed around the Mountain boards. The peripheral hardware used in *Compumusic* consists of a small console that interfaces with the Apple via a circuit board configurable for any free slot. This console is the focus of the system; no musical keyboard is used here, and it's not possible to play compositions in real time. The Apple keyboard and the console are all that are used to control the output.

Compumusic allows the musician/comput-







Left, the easy-to-use main menu is displayed on the screen above the Roland *Compumusic* console with its analog sliders and knobs; center, notes are entered numerically to each channel with the music editor screen; right, the onboard drum synthesizer plays the patterns you create by placing Xs on a rhythm grid.

erist to extend the computer's power as a controller into the realm of analog synthesizers as well. For starters, there is a six-voice synthesizer and a seven-voice drum synthesizer, programmable with the software provided on a single disk. The timbres of the *Compumusic*'s sound-producing oscillators are not programmable; with the software you control only the rhythms and pitches, and, to a limited degree, expression.

The hardware integrates an analog mixer with sliders for controlling the output to your amplifier through standard phone jacks, and there's also a clock for controlling overall tempo. A significant feature is the series of control outputs that enable you to use the *Compumusic* software to control up to eight external analog synthesizers.

The system is divided into eight channels, each sequenced individually. Channel one, the melody channel, also has nonprogrammable analog controls on the console for manipulating the sustain and decay of its sounds. Channel two, the bass channel, and channels three through six, the chord channels, have similar controls available, but only for the decay parameter. Channels seven and eight have no sound sources of their own and are reserved for use in controlling external synthesizers and effects.

All the channels are programmed in the same way using the Apple. Note that there are separate outputs and volume control sliders on the console for melody, bass, chord, and rhythm channels respectively, as well as a master slider that controls the volume of the combined output.

The software that runs *Compumusic* operates very much like a word processor or, in this case, a music processor. The single-page main menu offers access to the editor screens for each channel. Also available are single-keystroke commands to play music that's been composed using the program, loop music so it plays continuously, load and save music files to disk, and set the tuning on all channels automatically (nonstandard tunings are not supported). Examples of music are provided on the program disk for your inspiration.

To compose or edit, you select the channel you wish to work in, then specify the measure and step numbers. You can begin or resume work at any measure in a composition. Each note or rest is considered a step, and each step in a piece can be homed in on for modification as well. Standard musical notation is eschewed in favor of an approach using numbers to specify each step in a time sequence, listed on the screen in measures and scrolled vertically.

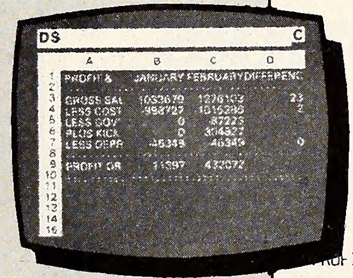
For each step, the control voltage data, which determines each step's pitch in numerical form, must first be specified. Numbers from 0 through 72 span a six-octave range, with each octave consisting of twelve steps. Next you specify the step time for each note or rest, determining its individual duration relative to an arbitrary time base of your choice. The time base functions like the time signature in standard notation; it is the number used to represent a quarter note, and can range from 1 to 255. For example, a time base set at 24 can be evenly divided by 2, 3, 6, 8, and 12, representing eighth notes, eighth-note triplets, sixteenth notes, sixteenth-note triplets, thirty-second notes, and thirty-second-note triplets respectively. The final numerical value you must set for each step is the gate time. This determines a note's articulation—whether it will sound staccato or slurred. In practice, entering these settings for each note is less complicated than the description of the process may suggest. Each parameter automatically defaults to the one directly preceding if you don't change a setting, saving you from tedious retyping to repeat pitches or time values. The settings as you input them remain visible on the screen for your reference, and any modifications are made, as with a word processor, by moving the cursor around the screen with a diamond of keys.

Other editor commands allow you to scroll

VisiCalc* Formatting Aids

"ADVANCED" Formatting Features

YOUR VisiCalc WISH-LIST HAS COME TRUE!



PRINT REPORTS WITH VARIABLE-WIDTH COLUMNS.
PRINT SELECTED ROWS AND COLUMNS.
ADD MULTI-LINE, BLOCK CENTERED HEADERS.

PRINT MULTIPLE COPIES AUTOMATICALLY, AND INSERT "TOP-OF-FORM" FEED ANYWHERE IN YOUR REPORT.

A	B	C	D E	R
PROFIT & LOSS SUMMARY	JANUARY	FEBRUARY	DIFF	YR-TO-DATE
GROSS SALES	1033679	1276103	23%	2309782
LESS COST OF GOODS SOLD	-998727	-1015286	2%	-2014013
LESS GOV'T PAYOFFS	0	-87223		-87223
PLUS KICKBACKS	0	304827		304827
LESS DEPRECIATION EXPENSE	-46349	-46349	0%	-92698
PROFIT OR (LOSS)	-11397	432072		420675

FLOATING $ SIGN, COMMAS, AND PARENTHESES CAN BE ENTERED BY GRID, ROW, COLUMN, OR GLOBALLY.
0-3 CHARACTER REPLACEMENT FOR "0" AND "0.00".
INSERT UP TO 3 CHARACTERS BETWEEN COLUMNS.

A	B	C	R
PROFIT & LOSS SUMMARY	JANUARY	FEBRUARY	YR-TO-DATE
GROSS SALES	$1,033,679	$1,276,103	$2,309,782
LESS COST OF GOODS SOLD	(998,727)	(1,015,286)	(2,014,013)
LESS GOV'T PAYOFFS	-0-	(87,223)	(87,223)
PLUS KICKBACKS	-0-	304,827	304,827
LESS DEPRECIATION EXPENSE	(46,349)	(46,349)	(92,698)
PROFIT OR (LOSS)	$(11,397)	$432,072	$420,675

Apple II+ owners can enjoy the calculating power of VisiCalc with "advanced" formatting features found only on the Apple III.

Introducing VisiCalc Formatting Aids 4.1, the latest version of this popular VisiCalc enhancement program which adds print-using to the earlier features of printing variable width columns, formula reading and file handling.

The new global features are as easy as selecting A, B, or C from a menu. There are only 3 VisiCalc-like key commands for non-global formatting of specific grids or ranges of columns or rows. There are 3 other key commands for printing selected rows, controlling line spacing, and issuing top-of-form feeds. These 6 keys are the only non-english commands in the entire program.

No more paste-ups. No more retyping. No more marginal-looking reports. All this without sacrificing VisiCalc memory! VisiCalc Formatting Aids reads files just as VisiCalc does, but instead of taking up memory for calculating power, VisiCalc Formatting Aids is dedicated solely to file handling and report formatting. In fact, it can process huge files created on expansion RAM boards quickly and easily.

For added flexibility and professional-looking reports, buy your copy today! 48k. Available for the Apple II+, II-E, and Apple III. Call or write for details and the name of a dealer near you. Price $54.95 ($44.95 thru June 15, 1983).

DATA SECURITY CONCEPTS

P.O. Box 31044, Des Peres, MO 63131 314-965-5044

* VisiCalc is a trademark of VisiCorp
* Apple is a trademark of Apple Computer Co.

forward and back within a composition, to copy measures from any place in a piece, insert them at any point, delete measures at a single keystroke, play individual measures or segments for immediate reference, and transpose any segment of the music automatically.

Creating sequenced patterns for the onboard drum synthesizer, in channel zero, entails a different editor screen in the form of a grid. Each repeating rhythm pattern is stored separately as sixteen steps, listed vertically along the grid. Each step has separately programmable step times, to accommodate different meters, rhythms, syncopation, or drum effects like flams and rolls. Seven distinct drum sounds are available, reflecting the timbre options commonly available on Roland's well-known line of synthesized rhythm machines.

Rhythms are programmed by simply inserting Xs at various points on the grid, triggering the chosen drum sound at the indicated step time. You can also listen to a pattern after each modification is made. After you've developed a collection of patterns, you can arrange and combine them in any order to form a rhythm track that plays in conjunction with the melodies and chord progressions on the other channels.

The ability to harness microcomputer power to control any voltage-controlled synthesizer on the market is *Compumusic*'s unique strength. The control possible with other voltage-controlled sequencers available pales in comparison to the amount of control possible with an Apple at the helm. The tradeoff for this capability seems to be the limitations of the system's built-in sound-producing oscillators. While triggering their playing patterns is completely programmable, there is no way to alter their timbres, short of some form of external effects modification to the output.

The software approach takes a little time to get familiar with, but you sense it's working with you, not against you. Commands are straightforward, usually single-keystroke, and screen formats are accessible and direct. The manual moves succinctly from simple to more refined features of the system. Ways of effecting complex rhythms, syncopations, trills, glissando, and grace notes are discussed, but the book doesn't have all the answers. The examples take you as far as illustrating the musical tools *Compumusic* makes available. You are left sparked by the challenge of applying them to their fullest potential in your own musical creations.

Although the software provided by Roland is all that's available now, the *Compumusic* is open-ended and alternative software approaches that users can develop will no doubt evolve to explore the system's range. This foray by a reputable long-established manufacturer of synthesizers, rhythm machines, and amplifiers reflects the burgeoning interest of electronic instrument manufacturers in developing products that integrate or use existent microcomputer technology.

MIDI Music Mover. Synthesizers have historically developed along nonstandardized or variously standardized lines, resulting in differing control voltages and output levels among the devices available. Similarly, even microcomputer-based synthesizer equipment from various manufacturers has developed along independent, incompatible lines. The proliferation of the home-computer market has forced those producing synthesizers to address the issue of equipment incompatibility.

Artists using electronic music equipment professionally desire the expanded capabilities of current technology that microcomputer control provides, as well as the potential creative benefits of interfacing systems developed by different manufacturers. With interfaced, expandable systems, a musician's costly investments are secured in equipment thereby protected from obsolescence.

Music industry concern with these issues has led to the development of the Musical Instrument Digital Interface specification (MIDI). MIDI is really nothing more than an informal agreement between electronic equipment manufacturers on some simple standard interface circuitry, and on the grammar of a nonproprietary language to carry meaningful information between various instruments. This makes it possible to devise a multi-instrument, completely programmable music system, consisting of devices from various manufacturers interfaced via

MIDI, and which is entirely software controlled from an Apple or other microcomputer.

Synthesizers using MIDI can be configured in parallel, each playing individually or simultaneously, mono or polyphonically, using a single computer to create and edit sequences and compositions. MIDI also makes possible the development of software to generate hard copy of a composition or improvisation, aid in teaching music education and electronic synthesis skills and theory, and integrate video synthesis with music synthesis. Even with MIDI, though, the total control features available are still dependent on the design of each specific unit. MIDI enables different types of equipment to communicate at their least common level, but it won't transcend inherent limitations or features that make each synthesizer unique. For example, design differences make the transference of specific programmed sounds between various models of synthesizers impossible, but keyboard data and program selections could be communicated.

On a given piece of equipment, the presence of MIDI is not apparent because it is built right in. Its only physical indication is a couple of five-pin DIN jacks on the unit, which are needed to connect various instruments to one another or to a controller card in the computer. Information is transferred to and from the computer serially, at 31.25 kilobaud, asynchronous.

The incorporation of MIDI in a product line remains optional for each equipment manufacturer. MIDI is new, so it has not been fully integrated in all units produced by manufacturers who do support it. Sequential Circuits began shipping the Prophet-600 polyphonic synthesizer at the end of last year, the first model in its line featuring MIDI. Specifications on the structure of MIDI and its data formatting are made available through Sequential Circuits to programmers/musicians interested in developing software for the Apple using the interface. Roland Corporation will also be manufacturing equipment that supports MIDI.

Chroma's Subtractive Synthesis. An Apple-interfaceable system that takes an altogether different approach from the others is the Rhodes Chroma developed by Fender/Rogers/Rhodes for CBS. The Chroma is a sixteen-oscillator, sixteen-channel, programmable polyphonic synthesizer, sporting a touch-sensitive, velocity-sensitive, five-octave keyboard.

The unit has fifty preset voices that may be combined two at a time or played separately by splitting the keyboard in half at any designated point. The fifty voice-select switches on the Chroma's front panel also double as very detailed sound parameter controls when the instrument is in programming mode, allowing you to create very finely tuned sounds stored digitally in Chroma's onboard memory. Unlike the Mountain, Passport, and Syntauri systems, Chroma employs a subtractive synthesis approach exclusively. All manipulable parameters usually associated with the subtractive synthesis are offered, each programmable to a high degree of incremental precision.

The patch switch on the Chroma's front panel gives the user access to an important variable not usually manipulable to such a high degree on other synthesizers. This switch enables you to choose the overall configurations of the synthesizer's channels, which determines the routing of signals from the oscillators through the filters to the amplifiers. Altogether, sixteen different channel configurations are possible. Other features include touch-of-a-button parameter editing, the ability to copy parameters from one preset to another, the autotuning of all oscillators, transpose functions, foot switches for controlling sustain and for stepping through presets, and an interface port for tape-cassette storage of preset banks.

Sound parameters are not controlled on the Chroma with accessible knobs and dials but are input individually in numeric form. This approach limits your ability to change parameter settings in real time; you can change only one. Players accustomed to being able to reach out and change any settings during a performance may find this unsatisfying. This is the tradeoff for the increased power that the Chroma's digital control provides.

The Chroma becomes a music terminal when connected to the Apple via an interface card that goes in slot 5 (reconfigurable). All information input to the Chroma is now accessible through the Apple. This includes all commands used to set up the voices, as well as all performance information.

A set of applications programs on two disks comes with the Apple interface kit (two drives recommended). This allows you to perform sequencing, editing, and data storage functions. This sequencer is fully polyphonic and can record up to sixteen independent tracks, each played, with an optional click track, in real time on the Chroma keyboard. The program's disk menu is long, but the options are straightforward enough. The menu length is more a reflection of the range of control available to the user than anything else.

The capacity of the sequencer is approximately eighteen hundred sixty notes, with the actual capacity dependent upon the amount of performance control changes entailed; it also records such performance nuances as key velocity and pressure sensitivity. A Scrunch utility will net an extra hundred notes or so.

After a sequence is recorded, it can be saved to disk along with all the presets that went into making it. Sequences can also be looped and finely edited down to changing or deleting notes, velocities, or other performance controls. The voices and volumes of individual sequenced tracks can be altered, as can the tempo.

Because of the fine-tuning possible, the editing process can get rather involved. There also seem to be a few ways of losing sequences in the process of editing. The editor does not automatically alert you to these danger points when encountered. A good amount of time is required to learn how to take full advantage of this program's potential.

Numerous utility programs come with the Chroma system. The extensive manual gives detailed information about the system to enable users/musicians to write their own software for the Chroma. Four such utilities are provided as examples on disk. The system manual offers detailed instructions on how to interface the Chroma with other sequencers, rhythm machines, and the expander module that's manufactured to double its capabilities.

The Chroma has been built for the professional musician, and the sounds that this system can produce are exceptional. Computerists with a flair for hacking are likely to find programming the Chroma a challenge. For a musician unfamiliar with sound synthesis or microcomputers, the programming approach the Apple interface makes possible might prove formidable.

It seems certain that systems like MIDI and the Chroma will act as catalysts in bringing together talents from distinct disciplines, and will help to engender the proliferation of a new breed of composer, who embodies the valuable qualities of both artist and technologist.

The Chroma music terminal from Fender/Rogers/Rhodes has a control panel with fifty presets and a keyboard that provides touch sensitivity.

Fender/Rogers/Rhodes, 1300 Valencia Drive, Fullerton, CA 92631; (714) 879-8080. Mountain Computer, 300 El Pueblo Road, Scotts Valley, CA 95066; (408) 438-6650. Passport Designs, 116 North Cabrillo Highway, Half Moon Bay, CA 94019; (415) 726-0280. Roland Corporation, 2401 Saybrook Avenue, Los Angeles, CA 90040; (213) 685-5141. Sequential Circuits, 3051 North First Street, San Jose, CA 95134; (408) 946-5240. Syntauri Corporation, 3506 Waverley Street, Palo Alto, CA 94306; (415) 494-1017.

THE PASCAL PATH

By Jim Merritt

Tools of the Craft, Part 23

This is the second of a series on Apple Pascal *compiler directive instructions* (called "compiler options" in Apple's official manuals). Last month, we looked at the general nature and use of compiler directives and experimented with specific directives that incorporated the instructions I (Include a source file) and S (compiler Swapping). These instructions permit the compiler to process (and thus allow the programmer to write) arbitrarily large source programs.

If you missed the previous column, you can prime yourself for this month's discussion by skimming chapter 4 in the Apple's *Pascal Language Reference Manual*—especially the sections on the I and S "options."

Program Listings. You have no doubt noticed that the larger a program gets the harder it becomes to visualize its construction and operation in your mind's eye. You simply cannot remember all the niggling details about every little function and routine, nor do you *want* to. One of the most important reasons for dividing a program into a collection of routines in the first place is to permit you to hide the trivial (or messy) details of implementation behind meaningful names.

From time to time, however, you'll need to reacquaint yourself with the nuts and bolts of your program in order to correct or improve it. You could, of course, content yourself with using the Editor to view the various parts of your program as necessary. However, most useful routines contain more than twenty-four lines of text and so cannot be displayed on the video screen all at once. Furthermore, in trying to repair incorrect code, you may wish to execute the program and correlate its "observed behavior" with that which you are led to expect from reading the source text. Of course, it is usually impossible for the Apple Pascal system to display a program's source text and the results produced by its execution on the same screen at the same time; the Editor (or Filer) simply cannot execute simultaneously with your program.

One excellent way of dealing with such situations is to use a printed listing of the source text. Because an 8½-by-11 sheet of paper can hold at least twice the textual information that the standard Apple's video screen can display, a program listing permits the typical procedure or function to be viewed in its entirety, all at once, thus speeding your comprehension of its structure and operation. Since the program listing is external to the computer, you may refer to it even while the object code is executing. Finally, if the machine-readable copy of the source text on disk is erased or otherwise contaminated, you can use a listing as your guide in retyping all or part of the affected source files.

Getting a Listing: The L Instruction. The compiler directive instruction L may be followed by either a *name parameter* (an arbitrary sequence of printable characters—for more information, see last month's column) or a *switch setting* (+ or −). When given a name parameter, this instruction causes the compiler to put a readable listing of the source, as compiled, onto the device or into the file named by the parameter. When given a switch setting, the L instruction simply enables or disables the listing facility.

In order to initiate the generation of a program listing, you must issue the L instruction with a name parameter. The following directive specifies PRINTER: as the listing destination:

```
(*$L PRINTER:*)
```

If you want the entire source program to be listed on the PRINTER: device, place the directive at the beginning of a program's master source file. The directive

```
(*$L MYPROGLST.TEXT*)
```

routes the listing to MYPROGLST.TEXT. You must specify the .TEXT suffix explicitly if you want the compiler to produce a text file that can be read by the Editor.

Depending on the speed of your printer and the length of your program, the process of getting a listing can take many minutes, or even an hour or more. When making changes in a program, you may often wish to list only the modified portions in order to save time. Accordingly, you may use the directive

```
(*$L−*)
```

to turn off the listing facility, and

```
(*$L+*)
```

to turn it back on again. If you try to turn on the listing facility without specifying a listing destination—that is, if L+ is the first L instruction the compiler encounters while translating your program—the listing is routed to the file *SYSTEM.LST.TEXT.

The Anatomy of a Listing. The figure at the end of this article shows a typical program listing. Five pieces of information appear on the left-hand side of each source line. They are, in order from left to right, the line number, segment number, procedure number, lexical level indicator, and the I-count.

In the listing, every line of source code bears a unique *line number*. Although these numbers are insignificant to the compiler and to the Pascal system in general, they serve as convenient points of reference for the human reader and thus ease the examination of any particular program.

As mentioned last time, you may organize your program into one or more "segments." A segment is a collection of routines that can be erased from RAM memory as a whole whenever space is tight and none of the routines in the segment is being used. Later, when any of its routines are called, the entire segment is reloaded into memory from disk, so long as sufficient RAM is available to contain it. The compiler and Pascal system identify each segment by its *segment number*. As implied by the figure, the compiler assumes that all routines, and the main program body as well, should be placed into segment number 1. In a future article, we'll study special techniques you can use to force the compiler to put p-code into segments other than 1, so as to optimize a program's utilization of limited RAM memory.

A given segment of code may include many different procedures and functions. Relative to its own segment, each routine bears its own unique *procedure number*. If you look at the listing of the program *TinyCalc* in the figure, you'll see that the routines SkipBlanks, GoodInteger, GoodOp, and GoodExpr have been given the procedure numbers 2, 3, 4, and 5 respectively. The main body of code in any segment is given procedure number 1 for that segment. Thus, *TinyCalc*'s main program body is procedure number 1 in segment number 1.

Suppose your program contains two different segments, 1 and 11. *Each* of those segments may contain a "procedure number 5," just as someone in another telephone area may have the same phone number as you do. However, no more than one procedure may bear the number 5 in any one segment, just as you are the only one within your area code to have your telephone number.

The *lexical level indicator* is attached by a colon to the right of the procedure number in every line of a program listing. This indicator is a capital D for every line that occurs within a declaration section. Otherwise, it is a digit that indicates the level of statement nesting that exists as of the *end* of the corresponding source line. For instance, consider the function GoodInteger. Lexical level zero corresponds to a primary body of code. Thus, the BEGIN and END that envelop GoodInteger's body exist at lexical level zero. The WHILE-statement that begins on line 123 is nested one level deeper still, so its lexical level is one. Conceptually, the loop body is nested within the WHILE-statement, so the BEGIN and END of that body exist at lexical level two. Finally, the actual working statements of the WHILE-body are nested within BEGIN and END, at lexical level three.

If you get caught on a wet or blustery day with nothing better to do, you might like to test your understanding of Pascal statement nesting by trying to guess which lexical level the compiler will report for arbitrary lines of source code taken from one of your programs. You win whenever you agree with the program listing. Here are the principles you should keep in mind in order to master the game:

1. An entire procedure body is considered as a single compound statement and exists at lexical level zero.
2. Any other statement exists at a lexical level that is one greater than that of the statement enclosing it.
3. Two statements separated from each other by a semicolon exist at the same lexical level.
4. "Framing" information (including such keywords as BEGIN, END, WHILE, CASE, WITH, and so on) exists at the lexical level of the complex statement to which it belongs. For example, lines 123 and 124 consist of "framing" keywords and expressions for the WHILE-statement that is introduced there. All the code in these lines exists at lexical level one.

For practice, review the WHILE-loop of lines 123 through 129 (in order to verify that the compiler followed the rules above) in determining lexical levels.

A particular routine may consist of scores or hundreds of p-codes, and thus occupy scores or hundreds of bytes of RAM memory. (There is an upper limit on the size of a procedure body, however. The p-machine architecture prevents the body of any procedure or function from containing more than 1,200 bytes of code.) The sequence of p-codes that the compiler generates for any given Pascal statement lies a certain number of bytes beyond the beginning of the procedure body. Consider *TinyCalc*'s function, SkipBlanks. Its first statement translates into a sequence of p-codes four bytes long, starting precisely at the beginning of the body (that is, zero bytes beyond it). The p-codes corresponding to the second statement start four bytes away from the beginning, while those that match the third statement start twenty-eight bytes away from the beginning. (Remember that the second statement, a WHILE-loop, itself contains an assignment statement that is not counted separately.) The offsets 0, 4, and 28 are the *I-counts* (instruction counts) for the first, second, and third statements in SkipBlanks. Note that the phrase "instruction counts" is a misnomer, since the I-count corresponds not to p-code instructions but to individual bytes of p-code. The I-count may be used along with the segment and procedure numbers in pinpointing the precise location at which a program fails during execution.

Simple Debugging with the Compiler Listing. The compiler listing provides a vague illustration of the correspondence between the source version of your program and the executable p-code version that is produced by the compiler. You can demonstrate this in a very memorable fashion by compiling and executing *TinyCalc*. This program incorporates a subtle error in logic that cannot be detected at compile time and only becomes apparent during execution whenever you provide certain erroneous input.

TinyCalc is designed to display the results of simple arithmetic on pairs of numbers. It expects each line of input either to be empty or to contain a single simple expression. An empty input line signals *TinyCalc* that the user wants to quit. A simple expression consists of an Integer literal, followed by an operator and, finally, a second Integer literal. Either or both of the literals may be negative. For purposes of *TinyCalc,* an "operator" is a single character (+, −, *, or /) that stands for one of the operations—addition, subtraction, multiplication, or division. Both

```
3+5
```

and

```
-32/8
```

are examples of proper expressions.

TinyCalc is somewhat friendly, in that it ignores blanks that precede or follow literals or operators, such as in

```
3        *        24
```

which is perfectly acceptable input that yields a result of 72. Also, the program responds with a helpful error message whenever its input does not conform to the syntax for an expression. For instance, try feeding *TinyCalc*

```
three times three
```

or

```
123 + 456 / 789
```

and see what happens. In the first case, the program will recognize right away that "three" is not a proper Integer literal. Such input cannot possibly be an expression. The second example contains too many items; a proper expression contains exactly two Integer literals, separated by one operator. "Chain" operations are not permitted.

What happens when you provide only a single Integer literal, say 1234, as input? Try it! Afterward, your screen should resemble the following:

```
: 1234



Value range error
S# 1, P# 4, I# 23
Type < space > to continue
```

This indicates that something has gone seriously wrong with your program. The situation is so bad, in fact, that the operating system has had to interrupt *TinyCalc*'s execution. As soon as you press the space bar, the Pascal system will reinitialize itself, and you'll be left wondering what you did to deserve this harsh rebuke.

Even if you didn't already know that *TinyCalc* contains a deliberate error, the message at the bottom of your screen certainly suggests that something is not right within the program. What's more, it even tells you where to begin looking for the problem. Before you press the space bar, make note of the S (Segment), P (Procedure), and I (I-Count) numbers that are reported in the error message. Then, match them against the corresponding numbers in the *TinyCalc* listing.

The closest match occurs at line 171, where the S, P, and I numbers are 1, 4, and 21 respectively. This implies that the point at which the program fails lies somewhere within the code that evaluates the CASE-selector S[TSP].

When it occurs during access of an indexed variable, such as an array or String, a *value range error* usually indicates the use of an unacceptably large or small index. If you take the time to examine *TinyCalc* thoroughly, you'll see that the value of TSP is derived from that of SP, which is in turn derived (ultimately) from the value stored in the global variable ILInx. The nature of the operations performed on these variables guarantees that they'll never contain values less than one, which is the smallest value that can be used to index any String.

It must be, then, that TSP grows larger than Length(S). In fact, this is exactly what happens when you provide only a single Integer as input to *TinyCalc*. GoodExpr calls GoodInteger to determine the number that is expected to be on the left side of the expression. In this example, GoodInteger finds the String representation of 1234, translates that into an Integer, and leaves SP pointing to the character position that lies just beyond (to the right of) the literal. Since the literal is the only thing in the input line, SP becomes equal to (Length(S) + 1).

Next comes the call to GoodOp. Immediately, the local variable TSP is assigned the value of SP, which is (Length(S) + 1) in this instance. As in the other *TinyCalc* routines, TSP is used as a provisional version of SP. Its value becomes the permanent value of SP if, and only if, a proper operator character is found after zero or more blanks, beginning at position SP in S. Obviously, there are no blanks to be skipped in this example, so control passes quickly to the CASE statement. Since you may not index a String using any value greater than the dynamic Length of that String, TSP now represents an illegal index. At this point, control passes automatically to the operating system's error-handling routine, which issues an error message and suspends *TinyCalc*'s execution.

Even with this "bug," *TinyCalc* operates without mishap for *all* acceptable input and for *most* unacceptable input. It fails only when given a single Integer literal. While common sense indicates that this case won't occur often, one of Murphy's Laws (in its original form) states that if there is a wrong way for a person to perform a given task, somebody will eventually try it. So, we should take steps now to plug this hole before some unlucky user falls through it.

Fortunately, repairing GoodOp is a simple matter. Since the CASE selector is invalid only when TSP is larger than Length(S), we need only guard the CASE statement with a suitable IF-clause:

```
            .
            .
            etc.
            .
            .
IF (SkipBlanks(S, TSP) = 0)
   THEN
      (* No problem—either 0 or nonzero is okay. *);
(* What operator is it? *)
IF (TSP < = Length(S)) (* This "IF" fixes "the last bug" *)
   THEN
      CASE S[TSP] OF
        '+':
          BEGIN
            TDest := Add;
            .
            .
            etc.
            .
```

When TSP is too large, the CASE statement will be skipped, and TDest will retain its initial value of NoOp. Thus, neither Dest nor SP will be updated at the end of the function call, and the value returned by GoodOp will be False, properly reflecting the lack of an operator in the input.

Of course, most debugging chores are not as straightforward as this one, which was made even simpler because our discussion proceeded directly to the heart of the matter and avoided the wrong guesses that plague almost every attempt at troubleshooting. Despite all this, perhaps you are beginning to appreciate the usefulness of having a "compiler listing" handy when something goes wrong with your program.

Ejecting a Page during Listing: P. The P instruction is used to force the remainder of the compiler listing to start at the top of a new page. The compiler knows nothing about the dimensions of a printer page and does not leave top and bottom margins. It simply sends line after line of output to the device or file specified as the listing destination.

With the P instruction, you may compensate a bit for the primitive nature of the compiler's listing facility. For instance, you might want to use it before each procedure or function heading so as to place each important component of your program on its own output page.

When the compiler encounters the directive:

```
(*$P*)
```

it transmits the ASCII form-feed character, Chr(12), to the listing device or file before conveying the line that contains the directive. Most modern printers respond to the form-feed character by advancing the paper to the top of the next page. If you should direct the compiler listing to the CONSOLE: device, be aware that the Apple's CRT screen goes blank upon receipt of a form-feed character. This is the video equivalent of advancing to a new page. Note also that some output devices ignore Chr(12) and so will fail to respond properly (if at all) to the P instruction.

Directly Ahead. The original plans for this month's column also called for an examination of the compiler's contribution to Apple Pascal's "immune system," the means by which the operating system protects itself against renegade programs. The bad news is that we don't have enough space for that this time. The good news is that not only will we cover the "immune system" next month, but we will conclude our first look at compiler directives with an introductory discussion of the mysterious "U−" instruction, which has aroused intense curiosity among Apple Pascal programmers ever since the system was first released. Apple's traditional policy toward this instruction has been to release as little useful information about it as possible. To learn the compelling reasons behind this policy, don't miss next month's installment.

A Typical Compiler Listing

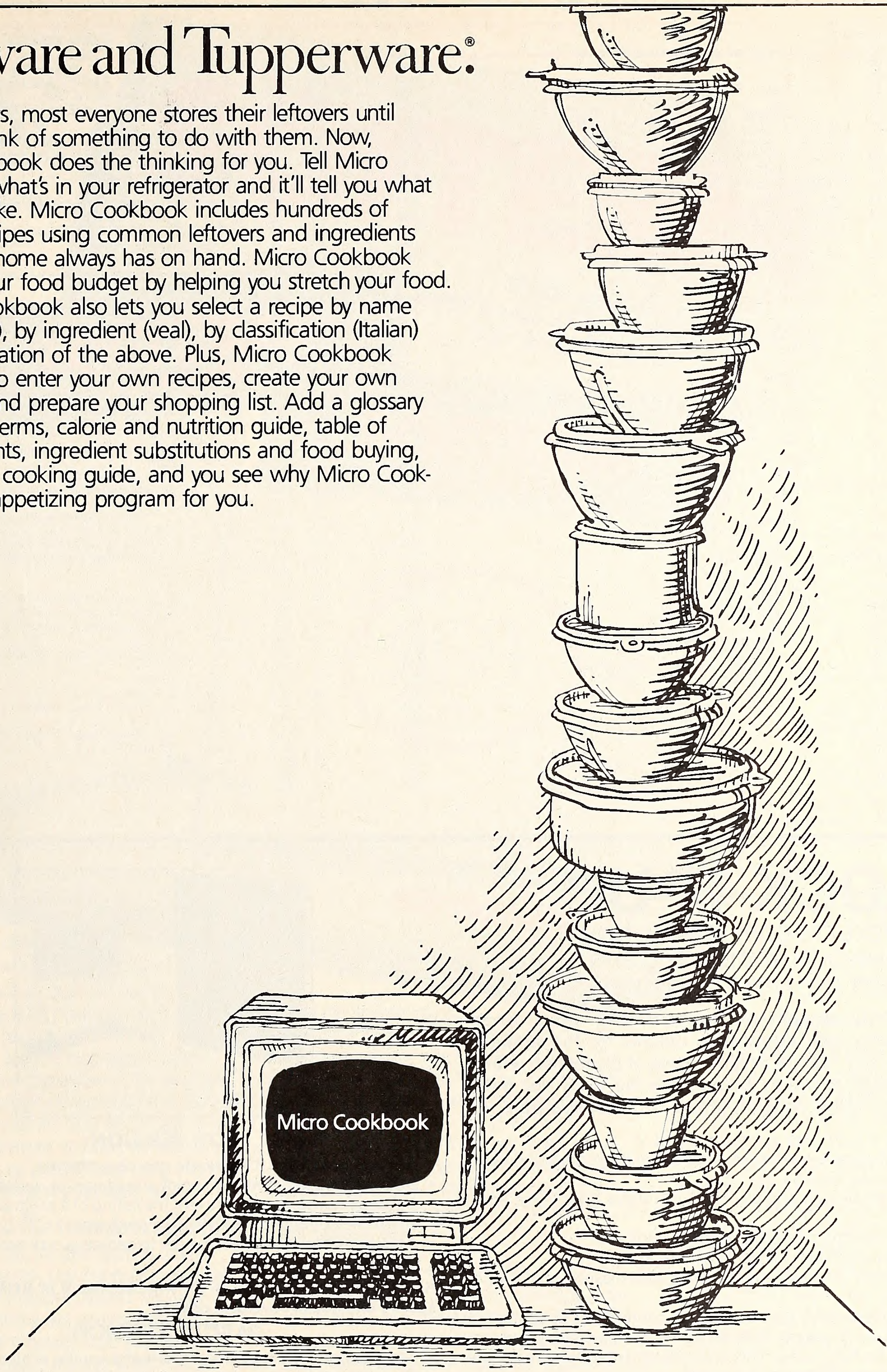
```
 1   1   1:D   1   (*$L PRINTER:*)
 2   1   1:D   1   (*$S+*)
 3   1   1:D   1   PROGRAM
 4   1   1:D   3     TinyCalc;
 5   1   1:D   3
 6   1   1:D   3     CONST
 7   1   1:D   3       Heading=    'TINYCALC (Ver 1.0: 8-Mar-83)';
 8   1   1:D   3
 9   1   1:D   3       Empty=      '';
10   1   1:D   3       Blank=      ' ';
11   1   1:D   3       Prompt=     ': ';
12   1   1:D   3       ErrorMsg=
13   1   1:D   3         'Bad expression; use <integer><operator><integer>';
14   1   1:D   3
15   1   1:D   3     TYPE
16   1   1:D   3       Operator=   (NoOp, Add, Subtract, Multiply, Divide);
17   1   1:D   3
18   1   1:D   3     VAR
19   1   1:D   3       Ouit        (* True when user wants to end the session. *)
20   1   1:D   3         :Boolean; (*   User enters blank line to quit.        *)
21   1   1:D   4
22   1   1:D   4       InLine      (* Holds the expression entered by the user. *)
23   1   1:D   4         :String;
24   1   1:D  45       ILInx       (* Index for individual chars within InLine. *)
25   1   1:D  45         :Integer;
26   1   1:D  46
27   1   1:D  46       Value       (* Holds value of expression in InLine. *)
28   1   1:D  46         :Integer;
29   1   1:D  47
30   1   1:D  47       OK          (* True when expression format is acceptable. *)
31   1   1:D  47         :Boolean;
32   1   1:D  48
33   1   1:D  48     FUNCTION
34   1   2:D   3       SkipBlanks(VAR S: String; VAR SP: Integer)
35   1   2:D   5         :Integer;
36   1   2:D   5       VAR
37   1   2:D   5         OriginalSP
38   1   2:D   5           :Integer;
39   1   2:D   6     (* On entry, assume Length(S) > = SP > = 1.
40   1   2:D   6        On exit, SP points to the first nonblank
41   1   2:D   6        character at or after the original SP
42   1   2:D   6        position. If all characters from original
43   1   2:D   6        SP onward are blank, SP contains
44   1   2:D   6        Length(S) + 1 on exit. In all cases,
45   1   2:D   6        return as function value the number of
46   1   2:D   6        blanks actually skipped (possibly 0).
47   1   2:D   6     *)
48   1   2:0   0     BEGIN (* SkipBlanks *)
49   1   2:1   0       OriginalSP := SP;
50   1   2:1   4
51   1   2:1   4       (* Skip any blanks: *)
52   1   2:1   4       (* NOTE: In the following loop, the
53   1   2:1   4          expression Copy(S, SP, 1) is used
54   1   2:1   4          instead of S[SP] to avoid value-range
55   1   2:1   4          errors in extreme cases. Don't alter
56   1   2:1   4          this expression unless and until you
57   1   2:1   4          know why it is "safe."
58   1   2:1   4       *)
59   1   2:1   4       WHILE (Copy(S,SP,1) = Blank) DO
60   1   2:2  20         SP := SP + 1;
61   1   2:2  28
62   1   2:2  28       (* Compute and return number of blanks
63   1   2:2  28          skipped:
64   1   2:2  28       *)
65   1   2:1  28       SkipBlanks := SP - OriginalSP;
66   1   2:0  34     END   (* SkipBlanks *);
67   1   2:0  48
68   1   1:0  48     FUNCTION
69   1   3:D   3       GoodInteger(VAR S: String;      VAR SP: Integer;
70   1   3:D   5                   VAR Dest: Integer)
71   1   3:D   6         :Boolean;
72   1   3:D   6     (* Return True if character sequence in S,
73   1   3:D   6          starting at position SP, represents a
74   1   3:D   6          valid Integer (ignoring leading blanks).
75   1   3:D   6          If so, SP becomes SP +< length of sequence>,
76   1   3:D   6          and Dest acquires the value of the
77   1   3:D   6          corresponding Integer. On False return,
78   1   3:D   6          SP and Dest remain untouched.
79   1   3:D   6     *)
```

```
1    3:D    6
1    3:D    6      CONST
1    3:D    6        Radix=    10;
1    3:D    6
1    3:D    6      VAR
1    3:D    6        (* Again, we work with temporary
1    3:D    6             string pointers,Dest variables,
1    3:D    6             until we know we have a winner;
1    3:D    6             then, everything is made
1    3:D    6             permanent.
1    3:D    6        *)
1    3:D    6        TSP,
1    3:D    6        TDest
1    3:D    6          :Integer;
1    3:D    8
1    3:D    8        SyntaxOK,    (* True if Good Integer so far. *)
1    3:D    8        Sign         (* True if Dest should be neg *)
1    3:D    8          :Boolean;
1    3:D   10
1    3:D   10        SignChar     (* "Holding Tank" that lets us     *)
1    3:D   10          :String[1]; (*    convert easily between String *)
1    3:D   11                     (*    and Char. Made small to    *)
1    3:D   11                     (*    conserve memory space.     *)
1    3:0    0    BEGIN (* GoodInteger *)
1    3:1    0      SyntaxOK := False;
1    3:1    3      TSP := SP;
1    3:1    7      IF (SkipBlanks(S, TSP) = 0)
1    3:1   16        THEN
1    3:0   18          (* No problem—either 0 or nonzero is okay. *);
1    3:0   18
1    3:0   18      (* Just gimme some kinda sign, ye-ah! *)
1    3:0   18      (* OneChar is used for convenience only *)
1    3:1   18        SignChar := Copy(S, TSP, 1); (* Get possible sign *)
1    3:1   32        Sign := (SignChar = '-');
1    3:1   39        IF ((SignChar = '+') OR (SignChar = '-'))
1    3:1   50          THEN (* Look beyond it *)
1    3:2   52            TSP := TSP + 1;
1    3:2   57
1    3:2   57      (* At this point we had better be looking at a digit,
1    3:2   57         or else say no go, no-oo, oh I can't go for that,
1    3:2   57         no can do. . . .
1    3:2   57      *)
1    3:1   57      TDest := 0;
1    3:1   60      WHILE ((Copy(S, TSP, 1)>= '0')
1    3:1   73              AND (Copy(S, TSP, 1)<= '9')) DO
1    3:2   89        BEGIN (* Looks a lot like IntegerInput *)
1    3:3   89          SyntaxOK := True;
1    3:3   92          TDest := (TDest*Radix) + (Ord(S[TSP]) - Ord('0'));
1    3:3  104          TSP := TSP + 1;
1    3:2  109        END;
1    3:1  111      GoodInteger := SyntaxOK;
1    3:1  114      IF SyntaxOK
1    3:1  114        THEN
1    3:2  117          BEGIN (* Make everything permanent *)
1    3:3  117            Dest := TDest;
1    3:3  120            IF Sign
1    3:3  120              THEN
1    3:4  123                Dest := -Dest;
1    3:3  128            SP := TSP;
1    3:2  131          END;
1    3:0  131    END (* GoodInteger *);
1    3:0  146
1    1:0  146    FUNCTION
1    4:D    3      GoodOp(VAR S: String;       VAR SP: Integer;
1    4:D    5               VAR Dest: Operator)
1    4:D    6          :Boolean;
1    4:D    6    (* Return True if character sequence in S,
1    4:D    6         starting at position SP, represents a
1    4:D    6         valid Operator (ignoring leading blanks).
1    4:D    6         If so, SP becomes SP + <length of sequence>,
1    4:D    6         and Dest acquires the value of the
1    4:D    6         corresponding Operator. On False return,
1    4:D    6         SP and Dest remain untouched.
1    4:D    6    *)
1    4:D    6
1    4:D    6      VAR
1    4:D    6        TSP
1    4:D    6          :Integer;
1    4:D    7        TDest
1    4:D    7          :Operator;
1    4:D    8
1    4:0    0    BEGIN (* GoodOp *)
1    4:0    0      (* Guilty until proven innocent. *)
1    4:1    0      GoodOp := False;
1    4:1    3      TSP := SP;
1    4:1    7      TDest := NoOp;
1    4:1   10      IF (SkipBlanks(S, TSP) = 0)
1    4:1   19        THEN
1    4:0   21          (* No problem—either 0 or nonzero is okay. *);
1    4:0   21
1    4:0   21      (* What operator is it? *)
1    4:0   21      (* There's a subtle bug here . . . can you discover it? *)
1    4:1   21      CASE S[TSP] OF
1    4:1   27        '+':
1    4:2   27          BEGIN
1    4:3   27            TDest := Add;
1    4:2   30          END;
1    4:1   32        '-':
1    4:2   32          BEGIN
1    4:3   32            TDest := Subtract;
1    4:2   35          END;
1    4:1   37        '*':
1    4:2   37          BEGIN
1    4:3   37            TDest := Multiply;
1    4:2   40          END;
1    4:1   42        '/':
1    4:2   42          BEGIN
1    4:3   42            TDest := Divide;
1    4:2   45          END;
1    4:1   47      END (* CASE S[TSP] *);
1    4:1   66      (* If S[SP] did not match any of the above operators,
```

Software and Tupperware®

These days, most everyone stores their leftovers until they can think of something to do with them. Now, Micro Cookbook does the thinking for you. Tell Micro Cookbook what's in your refrigerator and it'll tell you what you can make. Micro Cookbook includes hundreds of delicious recipes using common leftovers and ingredients most every home always has on hand. Micro Cookbook stretches your food budget by helping you stretch your food.

Micro Cookbook also lets you select a recipe by name (Veal Picatta), by ingredient (veal), by classification (Italian) or a combination of the above. Plus, Micro Cookbook allows you to enter your own recipes, create your own cookbook and prepare your shopping list. Add a glossary of cooking terms, calorie and nutrition guide, table of measurements, ingredient substitutions and food buying, storage and cooking guide, and you see why Micro Cookbook is an appetizing program for you.



VIRTUAL COMBINATICS

P.O. Box 755, Rockport, MA 01966 (617) 546-6553

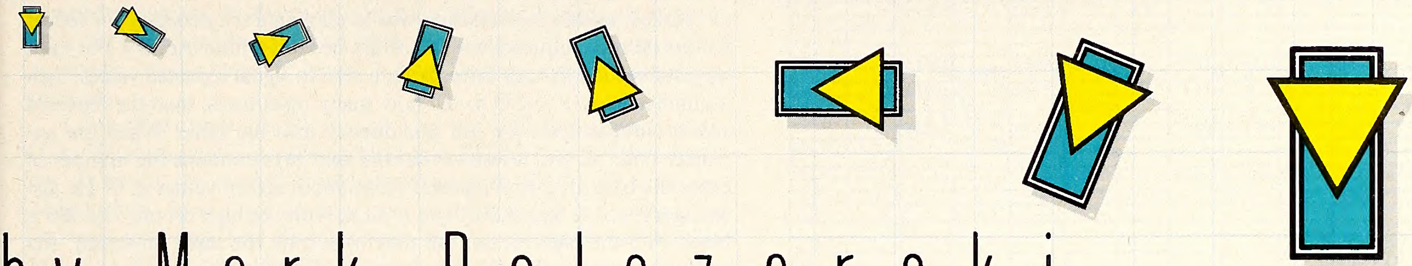
Look for Micro Barmate, the computer age bar guide and companion to Micro Cookbook.

Versions available for APPLE II +, APPLE IIe (80 column) and IBM PC (64K, PC DOS). The cost, $40. Available at your favorite dealer or by mail. VISA, MC or phone orders accepted. Please add $1.50 handling charge.

APPLE, IBM and Tupperware are registered trademarks of Apple Computer, Inc., IBM Corporation and Dart Industries Inc., respectively.

```
1   4:1   66        TDest will still equal NoOp at this point.
1   4:1   66      *)
1   4:1   66
1   4:1   66      IF (TDest <> NoOp)
1   4:1   69        THEN
1   4:2   71          BEGIN
1   4:3   71            Dest := TDest;
1   4:3   74            SP := TSP + 1;
1   4:3   79            GoodOp := True;
1   4:2   82          END;
1   4:0   82    END   (* GoodOp *);
1   4:0   94
1   1:0   94    FUNCTION
1   5:D    3      GoodExpr(VAR S: String;      VAR SP: Integer;
1   5:D    5                 VAR Dest: Integer)
1   5:D    6        :Boolean;
1   5:D    6    (* Return True if character sequence in S,
1   5:D    6        starting at position SP, represents a
1   5:D    6        valid Expression. If so, SP becomes
1   5:D    6        SP + <length of sequence>, and Dest
1   5:D    6        acquires the Integer value of the
1   5:D    6        corresponding Expression. On False
1   5:D    6        return, SP and Dest remain untouched.
1   5:D    6
1   5:D    6        Syntax:
1   5:D    6
1   5:D    6          <Expression> ::= <Integer><Operator><Integer>
1   5:D    6          <Operator>   ::= '+' | '-' | '*' | '/'
1   5:D    6    *)
1   5:D    6
1   5:D    6      VAR
1   5:D    6        TSP,     (* Temporary SP. *)
1   5:D    6        Arg1,    (* Value of left-hand argument. *)
1   5:D    6        Arg2     (* Value of right-hand argument. *)
1   5:D    6          :Integer;
1   5:D    9        Op       (* Type of operator. *)
1   5:D    9          :Operator;
1   5:0    0    BEGIN (* GoodExpr *)
1   5:0    0      (* Guilty until proven innocent. *)
1   5:1    0      GoodExpr := False;
1   5:1    3      TSP := SP;
1   5:1    7      IF GoodInteger(S, TSP, Arg1)
1   5:1   12        THEN
1   5:2   18          IF GoodOp(S, TSP, Op)
1   5:2   23            THEN
1   5:3   29              IF GoodInteger(S, TSP, Arg2)
1   5:3   34                THEN
1   5:4   40                  BEGIN
1   5:5   40                    CASE Op OF
1   5:5   43                      Add:
1   5:6   43                        Dest := Arg1 + Arg2;
1   5:5   50                      Subtract:
1   5:6   50                        Dest := Arg1 - Arg2;
1   5:5   57                      Multiply:
1   5:6   57                        Dest := Arg1 * Arg2;
1   5:5   64                      Divide:
1   5:6   64                        Dest := Arg1 DIV Arg2;
1   5:5   71                    END (* CASE Op *);
1   5:5   86                    SP := TSP;
1   5:5   89                    GoodExpr := True;
1   5:4   92                  END;
1   5:0   92      END   (* GoodExpr *);
1   5:0  104
1   1:0    0    BEGIN (* TinyCalc *)
1   1:1    0      WriteLn(Output, Heading);
1   1:1   50      REPEAT
1   1:2   50        WriteLn(Output);       (* Be neat and tidy! *)
1   1:2   58
1   1:2   58        Write(Prompt);         (* Whaddya want, ya user ya? *)
1   1:2   72        ReadLn(Input, InLine);
1   1:2   91
1   1:2   91        (* Start expression scan at first character position. *).
1   1:2   91          ILInx := 1;
1   1:2   94
1   1:2   94        (* A "blank" line may contain no characters or nothing
1   1:2   94          but blanks; this program accepts either. To do so,
1   1:2   94          it must bypass leading blanks.
1   1:2   94        *)
1   1:2   94        IF (SkipBlanks(InLine, ILInx) = 0)
1   1:2  104          THEN
1   1:1  106            (* No problem—either 0 or nonzero is okay. *);
1   1:2  106        Ouit := (ILInx > Length(InLine));
1   1:2  115
1   1:2  115        IF (NOT Ouit)
1   1:2  117          THEN (* InLine contains some info worth examining. *)
1   1:3  119            BEGIN
1   1:4  119              OK := GoodExpr(InLine, ILInx, Value);
1   1:4  131              IF (SkipBlanks(InLine, ILInx) = 0)
1   1:4  141                THEN
1   1:3  143                  (* No problem—either 0 or nonzero is okay. *);
1   1:4  143              OK := (OK AND (ILInx > Length(InLine)));
1   1:4  155                (* In other words, we found an expression,
1   1:4  155                  and there is no extra garbage after it.
1   1:4  155                *)
1   1:4  155
1   1:4  155              (* Let's be neat and line up the reply with the
1   1:4  155                original expression on the screen; shove out
1   1:4  155                some blanks to accomplish this.
1   1:4  155              *)
1   1:4  155              Write(Output, Empty:Length(Prompt));
1   1:4  173
1   1:4  173              IF OK
1   1:4  173                THEN
1   1:5  177                  Write(Output, Value)
1   1:4  188                ELSE
1   1:5  190                  Write(Output, ErrorMsg);
1   1:4  253              WriteLn(Output);       (* Finish the reply line. *)
1   1:3  261            END;
1   1:1  261      UNTIL Ouit;
1   1:0  264    END   (* TinyCalc *).
```

GRAPHICALLY SPEAKING



by Mark Pelczarski

When dealing with graphics on the Apple (or any other computer, for that matter), one of the limiting factors is the great amount of storage that graphic information requires. On the Apple, a hi-res picture in RAM takes almost one-quarter of the available programming memory (8K of approximately 36K, after DOS and other scratch areas are subtracted). On an Apple disk, you can typically fit only twelve hi-res screen pictures. Although there's not much to do about the amount of display RAM required, there are ways to scrunch more pictures onto disks.

As far as we know, Dave Lubar wrote the first picture-packing routines for the Apple about two years ago. Since then several others have been written, including a few direct modifications to Lubar's original routines. What's a picture packer? It's a program that takes a standard picture, stored in its full, glorious 8,192 bytes of memory, and looks for patterns that allow condensing of the information. (The implication is that one also needs a picture unpacker that will take the packed picture and put it back the way it was.)

Simple? Ah, but how does one look for the patterns? Which patterns pack the most efficiently? One of the best packers around is the one written by Dav Holle (you may recognize his name as the author of *Zoom Grafix*, coauthor in charge of graphics and various and sundry other details in *Sherwood Forest*, and, if you're very astute, the *Pie Man* cartoonist). As Holle puts it, his routines, as well as most others, are variations on Lubar's originals. Simply evolution at work. (Is it okay to talk about evolution, or does this mean that *Softalk* can't be read in certain schools now?)

In Dave Lubar's original packer, the basic idea was to look for any repeated values in the hi-res screen and lump them together so that screen values like 80 80 80 80 80, in sequence, would generate the packed code 05 80, meaning 80 repeated five times. Since the screen takes the memory addresses from 8192 to 16383 (base 10 addresses; $2000 through $3FFF in hexadecimal), the packing could occur sequentially in RAM rather than worrying about where on-screen the values are displayed. We do know that sequential bytes are displayed next to each other horizontally on-screen, so, for example, if the top third of the screen was black, each of the horizontal lines in that area would pack nicely.

The first problem arises from the fact that colors other than black or white don't create a byte pattern that repeats every byte. As we discovered a little earlier in this series, the colors that have every other bit set require one value in even bytes and a different value in odd bytes. See figure 1 for a refresher on the type of pattern for these colors. The result is that colors other than black or white wouldn't pack at all using this method. The solution to this problem is to have the program try packing twice—once checking every byte sequentially, and the second time trying every other byte for patterns and zipping through the screen twice, once for all even bytes and again for all odd bytes. The two trials could then be compared to see which was more efficient.

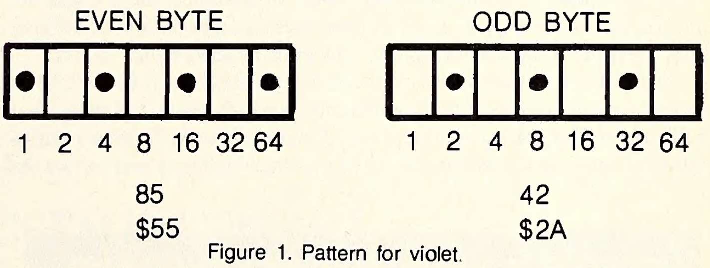


Figure 1. Pattern for violet.

That works better, but all the single, unique bytes still cause a problem. Since we're using pairs of bytes, one to tell how many repetitions and another to tell what repeats, patterns such as 55 55 55 55 55 00 00 00 42 42 42 42 pack nicely into 05 55 03 00 04 42. Unfortunately, patterns such as 55 00 46 93 FF A8 become 01 55 01 00 01 46 01 93 01 FF 01 A8. Not much savings there. . . .

A close approximation of Lubar's final algorithm (leaving out a couple of minor changes for clarity's sake) is to have the first byte in the pair count up to 127 repetitions (call this number N). The number 127 can be represented in seven bits. The eighth bit tells whether the following are N unique bytes, or one byte repeated N times. Using this technique, a pattern such as 85 79 A2 55 55 55 55 00 00 34 21 would be packed as (83) 85 79 A2 (04) 55 (02) 00 (82) 34 21. The bytes in parentheses represent the various values of N. Note that the 8 in the first position tells you that the high bit is set, so in the example we have three unique bytes first, then four repetitions of 55 followed by two repetitions of 00, and then two more unique bytes.

Well, Lubar's routine was really exciting. Suddenly we were able to get twenty-five to forty pictures on a disk, sometimes even more. Pictures that used to take thirty-four sectors of disk storage now took somewhere between seven and twenty-five sectors, in most cases. But, alas, there are always better ways. An analysis of a hypothetical hi-res screen will indicate the next step. Figure 2 is a scaled-down version of what the hi-res screen looks like, more or less.

Notice that the little lines in the diagram, which divide up the picture into bytes, are very close together as you move down the screen and not so close as you move across the screen. Bytes, when displayed in hi-res, are one dot tall and seven dots wide. Okay, big deal. We'll look at the illustration and figure out approximately what percent of the lines across the screen go the full forty bytes uninterrupted by the writing. Then scan the area vertically and figure approximately what percent of the area can

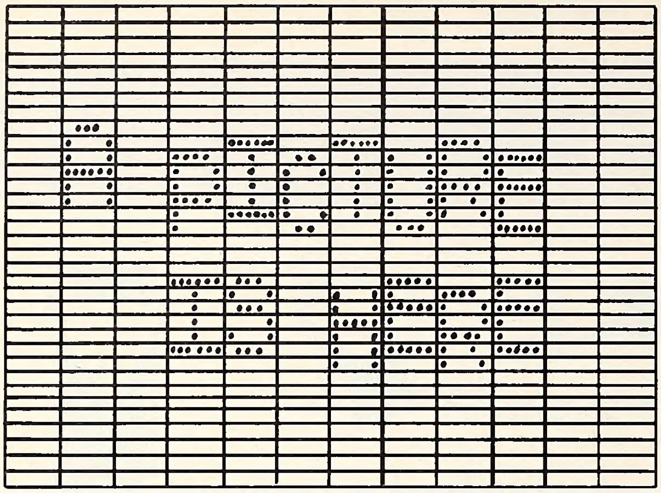


Figure 2. A hypothetical hi-res picture.

be covered by patches of repeated bytes. The answer is that you are more likely to find more repetitions when scanning vertically instead of horizontally. When scanning horizontally, you have to go all the way across the screen uninterrupted to get only forty repeating bytes. Scanning vertically, you are more likely to find repeating bytes because they are all in the same general area of the picture. In fact, as this picture is drawn, you would have more than four hundred bytes repeating on the left side before you ran into the letter A. Even then, there is a good amount of repetition vertically between the words and even in letters such as H.

So the trick is that scanning according to the screen instead of in memory sequence does make a difference, and it's on this premise that Holle based his packing routine. He did make a couple of other modifications to improve the efficiency. The first modification is sort of like the first modification to Lubar's routine. As you'll recall from our experimenting with color combinations, any color fill routine that creates more than the six Apple colors uses a two-line pattern. Remember when we made the odd horizontal lines orange and the even ones green and came out with a simulated yellow? That yellow, and every other blended color, would easily defeat Holle's idea for vertical packing unless it was done in two passes. The first pass does the odd lines and the second pass does the even lines.

Holle's second modification was to eliminate the necessity for repeat factors before unique values. To do so, he used a little trickery. For each *repeated* section he uses three bytes; a zero to signal repeated values, then a number from 1 to 255 to tell how many repetitions, then the repeated value. Unique bytes are just put directly into the table. When the unpacker finds a zero, it knows that the next byte contains the number of times the byte after that repeats. What about screen values of 0? He just changes them to $80. A$00 byte is black with the high bit off, and $80 is black with the high bit on, but they both look the same on-screen. Actually, it turns out that the major savings in this approach is that you can have repetitions up to 255 times instead of Lubar's 127, but as often as that occurs it probably doesn't matter much. Technically, it does eliminate one possibility in Lubar's: In his routine, it is actually possible for the packed picture to take *more* space than the original. This occurs if few or none of the values repeat; since there are flags every 128 bytes that say that the following 128 bytes are unique, those extra bytes would make the picture longer. Practically, it hardly ever happens, and Holle's three bytes for repetitions equally balance Lubar's one for nonrepeating values.

So the listing at the end of this article is Dav Holle's variation on Dave Lubar's original packing ideas, slightly modified. The routine as listed is assembled at $6000, decimal 24576, just above hi-res page two. It is 248 bytes long, and contains both the packer and unpacker.

If you want to assemble it somewhere else, change the ORG instruction in line 1. Those of you who tinker in machine language without the benefit of an assembler can relocate the routine by changing the six places in the code that call locations $605A and $60EF with JSR instructions.

There are three locations to set before calling either routine. In locations 0 and 1, poke the starting address of your packed picture table. This is the address where you want the packer to put the packed information, or the address where you bloaded a packed picture file. The address is stored in low/high format, which means that, if your address is A, you'd use the following pokes (in line 10 we give A a value of 24832, which is as good a location as any, since it's right above the pack/unpack routines):

```
10  A = 24832
20  POKE 0, A - INT (A / 256) * 256
30  POKE 1, INT(A / 256)
```

In location 230 ($E6), poke the value corresponding to the hi-res screen you want packed, or onto which you want the packed picture unpacked. *Poke 230,32* for page one, and *poke 230,64* for page two. Note that this location in most operations also tells the computer which hi-res screen is being drawn upon; this gives you a nifty trick for displaying one page while drawing on the invisible one.

After the above pokes, *call 24576* if you want to unpack a picture, or *call 24700* if you want a picture packed. After return from the routine, locations 0 and 1 hold the last address in the packed table. These two programs are examples of using the packer and unpacker respectively.

Packing

```
5   HGR
10  PRINT CHR$(4);"BLOAD PICTURE.PIC"
20  POKE 0,0 : POKE 1,97 : POKE 230,32
25  REM THE POKES IN 0 AND 1 GIVE THE ADDRESS 24832 IN
    SHORTER FORM.
30  CALL 24700
40  L = PEEK(0) + PEEK(1) * 256 - 24831
45  REM L IS THE LENGTH OF THE RESULTING PACKED PICTURE.
50  PRINT L
60  PRINT CHR$(4);"BSAVE PICTURE.PAC,A";24831;",L";L
```

Unpacking

```
5  HGR
10 PRINT CHR$(4);"BLOAD PICTURE.PAC"
20 POKE 0,0 : POKE 1,97 : POKE 230,32
30 CALL 24576
```

Those are bare-bones examples only, designed to show the essentials of using the Pack/Unpack routines. The next program is a little more complete example. It allows you to pack pictures in memory and unpack pictures from the disk using either hi-res screen. If you want to pack a picture, you'll have to bload it into memory before running the program.

Lines 10 through 170 ask for input on the options you want to use. Lines 200 through 250 handle packing, and 300 through 330 do unpacking. These are the essential necessary commands, like those in the previous listings. As is often the case, most of the program is devoted to supervising neat and, for the most part, error-protected input. The last part, lines 400 through 450, simply allows you to repeat the cycle or quit.

```
10  D$ = CHR$(4):PACK = 24700:UNPACK = 24576:LOC = 24831
20  PRINT D$;"BLOAD PACK/UNPACK"
30  HOME : HTAB 10: PRINT "PACKER/UNPACKER UTILITY"
40  PRINT : PRINT "PACK OR UNPACK (P/U)? ";
50  GET P$: IF P$ <> "P" AND P$ <> "U" THEN 50
60  PRINT P$
70  PRINT : PRINT "HI-RES SCREEN (1/2)? ";
80  GET S$: IF S$ <> "1" AND S$ <> "2" THEN 80
90  PRINT S$
100 PRINT : INPUT "FILENAME: ";F$
110 PRINT : PRINT : PRINT "OKAY (Y/N)? ";
120 GET A$: IF A$ <> "Y" AND A$ <> "N" THEN 120
130 IF A$ = "N" THEN 30
140 POKE - 16304,0: POKE - 16297,0: POKE - 16302,0:
    POKE 230,32
150 IF S$ = "2" THEN POKE - 16299,0: POKE 230,64
160 IF P$ = "P" THEN 200
170 GOTO 300
200 REM PACK AND SAVE
210 POKE 0,LOC - ( INT (LOC / 256) * 256): POKE 1, INT (LOC /
    256)
220 CALL PACK
230 LN = PEEK (0) + PEEK (1) * 256 - LOC: REM LENGTH
240 PRINT : PRINT D$;"BSAVE";F$;".PAC,A";LOC;",L";LN
250 GOTO 400
300 REM LOAD AND UNPACK
310 POKE 0,LOC - ( INT (LOC / 256) * 256): POKE 1, INT (LOC /
    256)
320 PRINT : PRINT D$;"BLOAD";F$;".PAC,A";LOC
330 CALL UNPACK
400 REM DONE. WHAT NEXT?
410 POKE - 16368,0: GET A$
420 TEXT : HOME : VTAB 10: PRINT "QUIT OR CONTINUE (Q/C)? ";
430 GET C$: IF C$ <> "Q" AND C$ <> "C" THEN 430
440 IF C$ = "C" THEN 30
450 HOME : END
```

Note that *.pic* was used at the end of the file name to denote a picture in standard format in these programs. The suffix *.pac* was also used to denote a packed picture file. When playing with graphics you tend to get a lot of different types of binary files on your disks. Using something in the name to designate what type of file it is helps determine if something called Frog is a picture, a packed picture, a preshifted shape, an Applesoft shape, or that new machine language routine you developed. Here are some that work well:

.PIC	for a standard format picture
.PAC	for a packed picture (.PAK in Lubar's format)
.SSH	for a preshifted shape
.SHP	for an Applesoft shape
.FNT	for a character set or text font

You are encouraged to use the packer and unpacker routines to handle pictures in the *.pac* format. If lots of people use *.pac* as a second standard, after the normal thirty-four-sector *.pic* format, programs from several sources will be able to communicate pictures to one another more efficiently. So feel free to use these routines in your own programs, but please acknowledge authorship when you do.

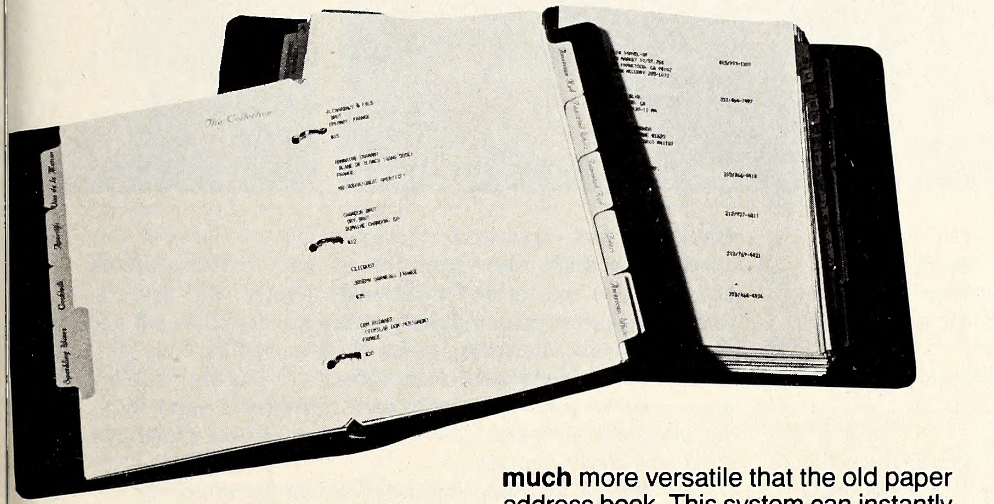
Pack/Unpack Routine

```
                 ORG $6000
        LINES    EQU 192        ;160 OR 192
                                 LINES
        TBL      EQU 0
        XCOORD   EQU 2
        YCOORD   EQU 3
        OFFSET   EQU 4
        GBAS     EQU 5
        PREV     EQU 7          ;PREVIOUS DATA
        REPEAT   EQU 8          ;REPEAT COUNT
6000: A5 E6     UNPACK   LDA $E6
6002: 09 04              ORA #$04
6004: 85 06              STA GBAS+1     ;$2400 (X=0,Y=1)
6006: A2 01              LDX #1
6008: 86 04              STX OFFSET
600A: A0 00              LDY #0
600C: 84 02              STY XCOORD
600E: 84 05              STY GBAS
6010: 84 08              STY REPEAT
6012: B1 00     PROC     LDA (TBL),Y    ;GET DATA
6014: D0 18              BNE STORE      ;SIMPLE DATA
6016: E6 00              INC TBL
6018: D0 02              BNE GETREP
601A: E6 01              INC TBL+1
601C: B1 00     GETREP   LDA (TBL),Y    ;GET COUNT
601E: 85 08              STA REPEAT
6020: E6 00              INC TBL
6022: D0 02              BNE GETPREV
6024: E6 01              INC TBL+1
6026: B1 00     GETPREV  LDA (TBL),Y    ;GET DATA
6028: 85 07              STA PREV
602A: A5 07     DOREP    LDA PREV
602C: C6 08              DEC REPEAT
602E: A4 02     STORE    LDY XCOORD
6030: 91 05              STA (GBAS),Y
6032: E8                 INX
6033: E8                 INX
6034: E0 C0              CPX #LINES     ;OFF BOTTOM?
6036: 90 12              BCC YOK        ; NO
6038: E6 02              INC XCOORD
603A: A4 02              LDY XCOORD
603C: C0 28              CPY #40        ;END OF PASS?
603E: 90 08              BCC XOK        ; NO
6040: C6 04              DEC OFFSET     ;MORE PASSES?
6042: 30 15              BMI DONE       ; NO
6044: A0 00              LDY #0
6046: 84 02              STY XCOORD
6048: A6 04     XOK      LDX OFFSET
604A: 20 5A 60  YOK      JSR BASEC
604D: A4 08              LDY REPEAT     ;REPEATING?
604F: D0 D9              BNE DOREP      ; YES
6051: E6 00              INC TBL
6053: D0 BD              BNE PROC
6055: E6 01              INC TBL+1
6057: D0 B9              BNE PROC       ;ALWAYS
6059: 60        DONE     RTS
605A: 8A        BASEC    TXA            ;CALC BASE
                                         ADDR
605B: 29 C0              AND #$C0
605D: 85 05              STA GBAS
605F: 4A                 LSR
6060: 4A                 LSR
6061: 05 05              ORA GBAS
6063: 85 05              STA GBAS
6065: 8A                 TXA
6066: 85 06              STA GBAS+1
6068: 0A                 ASL
6069: 0A                 ASL
606A: 0A                 ASL
606B: 26 06              ROL GBAS+1
606D: 0A                 ASL
606E: 26 06              ROL GBAS+1
6070: 0A                 ASL
6071: 66 05              ROR GBAS
6073: A5 06              LDA GBAS+1
6075: 29 1F              AND #$1F
6077: 05 E6              ORA $E6
6079: 85 06              STA GBAS+1
607B: 60                 RTS
607C: A0 01     PACK     LDY #1
607E: 84 04              STY OFFSET
6080: 84 03              STY YCOORD
6082: 88                 DEY
6083: 84 02              STY XCOORD
6085: A5 E6              LDA $E6
6087: 09 04              ORA #4
6089: 85 06              STA GBAS+1
608B: 84 05              STY GBAS
608D: B1 05              LDA (GBAS),Y
608F: D0 02              BNE MORE
6091: 09 80              ORA #$80
6093: A2 01     MORE     LDX #1
6095: 86 08              STX REPEAT     ;COUNT=1
6097: 85 07              STA PREV
6099: A4 02     TRAVEL   LDY XCOORD
609B: A6 03              LDX YCOORD
609D: E8                 INX
609E: E8                 INX
609F: E0 C0              CPX #LINES     ;OFF BOTTOM?
60A1: 90 0F              BCC YOL        ; NO
60A3: C8                 INY            ;BUMP XCOORD
60A4: C0 28              CPY #40        ;OFF RIGHT?
60A6: 90 06              BCC XOL        ; NO
60A8: C6 04              DEC OFFSET     ;DONE?
60AA: 30 1B              BMI NOTEQ      ; YES, FINISH UP
60AC: A0 00              LDY #0
60AE: 84 02     XOL      STY XCOORD
60B0: A6 04              LDX OFFSET
60B2: 86 03     YOL      STX YCOORD
60B4: 20 5A 60           JSR BASEC
60B7: B1 05              LDA (GBAS),Y   ;GET FROM
                                         SCREEN
60B9: D0 02              BNE PROC2
60BB: 09 80              ORA #$80       ;$00 —> $80
60BD: C5 07     PROC2    CMP PREV       ;SAME?
60BF: D0 06              BNE NOTEQ      ; NOPE
60C1: E6 08              INC REPEAT
60C3: D0 D4              BNE TRAVEL
60C5: C6 08              DEC REPEAT

60C7: 48        NOTEQ    PHA            ;SAVE NEW DATA
60C8: A0 00              LDY #0
60CA: A6 08              LDX REPEAT
60CC: F0 0E              BEQ BIG
60CE: E0 04              CPX #4         ;BIG ENOUGH TO
                                         USE CODE?
60D0: B0 0A              BGE BIG
60D2: A5 07              LDA PREV
60D4: 20 EF 60  LITTLE   JSR OUT
60D7: CA                 DEX
60D8: D0 FA              BNE LITTLE
60DA: F0 0D              BEQ FRESH

60DC: 98        BIG      TYA            ;REPEAT CODE=0
60DD: 20 EF 60           JSR OUT
60E0: 8A                 TXA            ;REPEAT COUNT
60E1: 20 EF 60           JSR OUT
60E4: A5 07              LDA PREV       ;REPEAT DATA
60E6: 20 EF 60           JSR OUT

60E9: 68        FRESH    PLA            ;GET NEW DATA
60EA: 24 04              BIT OFFSET     ;DONE?
60EC: 10 A5              BPL MORE       ; NO
60EE: 60                 RTS

60EF: 91 00     OUT      STA (TBL),Y    ;STORE IN TABLE
60F1: E6 00              INC TBL
60F3: D0 02              BNE DONE2
60F5: E6 01              INC TBL+1
60F7: 60        DONE2    RTS
```

Magic Memory™
Will Remember Everything For You

(If you want it to)

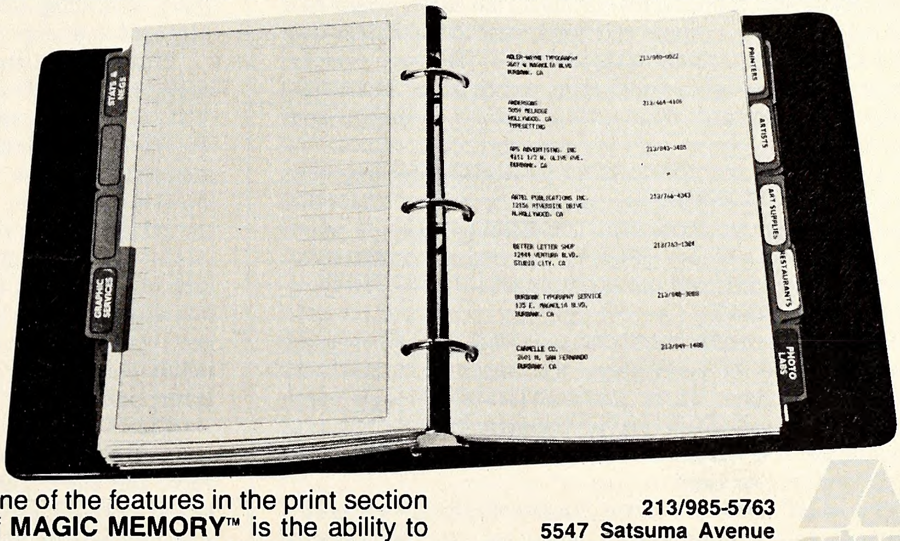
Imagine a system that would record all the wonderous, valuable information you have assimilated onto a single tiny disk. (No more scattered bits of paper, business cards, etc.) Imagine the same system giving you a typed sheet you could put into a notebook or print out for a party and instantly change, or add to, at a moments notice. Imagine cross-referencing to suit both your business needs and personal desires so that all your data was organized into one little black book! On top of all this — imagine having **fun** putting it together.



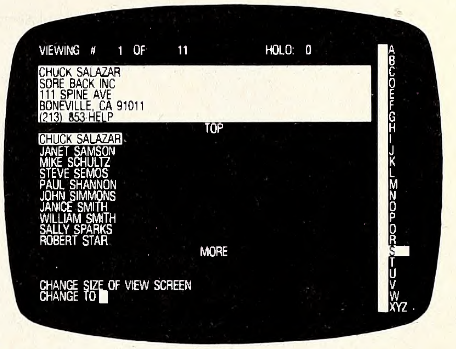
"Imagine" no more. ARTSCI, INC. proudly announces **MAGIC MEMORY™**, the complete advanced database system. **MAGIC MEMORY** is made to simplify the information storage process.

The typical first-time computer user has no *human* experience that will help him to relate to a *computer's* methods of handling information. Therefore, his learning ability is hampered and remains so for many hours of use, UNLESS the computer has been designed to run simply.

MAGIC MEMORY™ is built for the computer rookie. Everyone can relate to **MAGIC MEMORY™** because its form is familiar. It looks like an address book but its not. Its *more*. Like the address book, **MAGIC MEMORY™** presents an A thru Z index tabulation on the right edge of the video display. The user simply selects a tab and the *book* is opened to the proper page(s). A second set of tabs are available that can be labeled by the user (i.e.: companies one deals with, birthdays, lists, wines, shops, etc.) Yet **MAGIC MEMORY™** is **much** more versatile that the old paper address book. This system can instantly add or delete information, sort alphabetically, and transfer data to other locations in the book.



One of the features in the print section of **MAGIC MEMORY™** is the ability to print out any or all of the *address* book,



or any of its special sections. A vinyl binder is included in the package for the address or special section pages. Tabs are also available.

MAGIC MEMORY'S data presentation screen will instantly adapt to any type of information you may want to store. It is only limited to your imagination and your needs.

MAGIC MEMORY™ is designed to operate on an APPLE //e and still remain totally compatible with APPLE II. The system will operate in 40 columns or 80 columns. You may also use the 70-column display that requires no additional hardware.

MAGIC MEMORY™ will retail for $99.95. Call 213/985-5763. Use your imagination to outwit your organizational woes.

MAGIC MEMORY™ is a trademark of ARTSCI, INC.

213/985-5763
5547 Satsuma Avenue
North Hollywood, Ca 91601

artsci™

Computer

"If we fail to speak with a man who can be spoken with, we lose a man. If we do speak with a man who cannot be spoken with, our words go for nought. The wise lose neither man nor words."

—The Sayings of Confucius

At the end of a West Coast Computer Faire, there are two questions on everybody's mind: Where is a good place to get a foot massage, and what was the big story of the Faire?

This year the big story of the Faire is there was no big story. Early on, expectations ran high that there would be a showdown, because Apple and IBM were to have their own special halls. Everybody expected that the two personal computer heavyweights would come out fighting, exchanging punches at first and then getting down to some serious brawling.

As it turned out, IBM had no particularly thrilling news to announce or products to unveil. Apple, on the other hand, had Lisa to show off. The holding back of major announcements and unveilings by the major manufacturers seemed to have infected the independent software and hardware vendors as well.

Microcomputer Madness. Oh, sure, there was a new game here and a new hard disk there. Oh, sure, the booths were bigger and gaudier. There was even a walk-through booth (Perfect Software) that hinted of elaborate World's Fair-type displays; some attendees compared it to a ride at Disneyland, others were not so kind.

But where were the robots? Apparently Nolan Bushnell's group Androbot applied for a booth too late. Where were the computer-controlled ultralights? Where was there anything more exciting than the usual hushed "come-over-here-and-listen-to-my-big-scheme" scene?

Okay, perhaps we've been to too many of these affairs and can't see the forest for the trees. People who have recently become microcomputer owners no doubt found the Computer Faire an exciting event. With more than four hundred exhibitors—everyone from Abacus to Xcomp, displaying everything from accessories to wire strippers—and forty-six thousand attendees, the Eighth West Coast Computer Faire was bigger than ever.

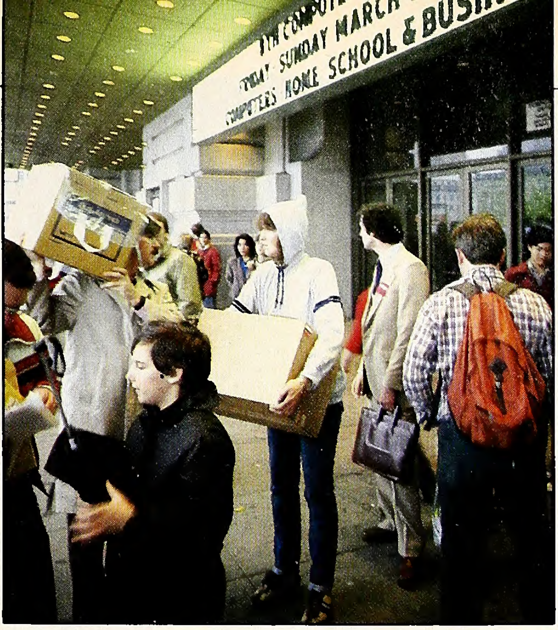
In a way, it was very exciting to see the computer becoming a mundane object. It made it easier to accept the lack of exciting new products. Each year personal computers are reaching more people and becoming part of normal life. This Faire, if nothing else, proved that the world-at-large is more enthused than ever with microcomputers.

Theory has it that if it rains the weekend of a computer show, the attendance increases by as much as 25 percent. Well, it was a gloomy, slippery, Chandleresque three days in the Bay Area. Along with the snails, worms, and other unmentionables, the weather brought out the nerds. They crawled all over the luscious computer goodies and harassed the exhibitors with their sticky, annoying questions.

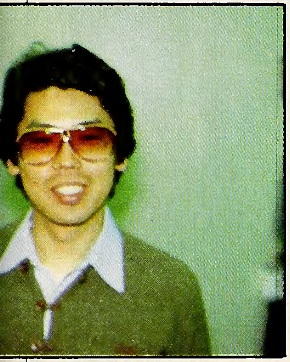
At the same time, Fred and Ethel, Mr. Normal, Mutt and Jeff, and Lois Lane joined the inquisitive throng. A significant percentage of the attendees were not computer owners; they looked over the goods, not necessarily wearing their money belts, but curious and seriously considering a near-future purchase.

They Came, They Wandered, They Left. Perhaps the big story at the Faire was who captured Captain Marvel, Laverne and Shirley, Han Solo, and Nancy Drew. Did they stay longer in IBM's hall or in Apple's? Did they ask questions or just pick up the literature? Did they look longer than two minutes at the Commodore?

The size of this show, both in area and number of bodies,



Faire



Top row, left to right: Apple's booth in the special Apple Hall; what the crowd is waiting to get a look at, Lisa; exhibitors overflow into every nook and cranny, including the bottom of Brooks Hall's truck-access ramp; attendees leave the Faire loaded to the hilt with microcomputer loot. Second row: Crowds surround the Broderbund booth; a weary but happy Gail Lasko and Jim Mangham of *Softdisk*; Jun Wada and Yoshio Taya of Programmers 3; Taya and Wada join the crowd in watching a demonstration of Broderbund's *A.E.*

was enough to wear out the most zealous consumer. While finicky nerds hassled beleaguered salespersons, discouraged novices were carried away in the flood.

Basically, it was a mess—but a well-organized, sprawling, fun mess. There were magicians doing card tricks, robots blindly bumping into people, a funky dude dressed like a caveman, and at least one stripper. Sure, there was a lot of serious business being done, but not so much so that people were glum about it.

Okay, enough of the big story and on to the products. A lot has changed in two years. Though there were few new offerings, most of the products on display, old and new, boasted a standard of excellence that would have floored an attendee at the Sixth West Coast Computer Faire. Last year's amateurs were this year's professionals, while the newer companies strove to make the best impression possible.

Ei-En Enterprise Company of Tokyo, Japan, displayed a streamlined disk drive that is half the height (forty-one millimeters) of conventional drives. It has a track-to-track time of three to six microseconds, compared to about forty microseconds on most standard models.

Micron Technology of Boise, Idaho, showed off its Micron Eye Bullet, a solid-state digital imaging system capable of transmitting up to fifteen frames per second and with 128-by-256 element resolution. It is compatible with the Apple II and smartly designed.

A Knee-High Mechanical Drunk. RB Robot Corporation of Golden, Colorado, had a big hit with its RB5X "intelligent robot," a miniature R2D2 equipped with tactile sensors. The robot is fully programmable with an Apple computer and will soon have accessories, and was the only robot to be found other than Heath's Hero I. RB5X at least had a chance to run around blindly on the floor and bump into total strangers. At Heath's booth, a couple of Hero I's were mounted on pedestals with nowhere to go.

Broderbund got into the Faire spirit and revealed several new games, including *Lode Runner*, *Gumball*, and *Spare Change*. TG Products's selectable joystick was impressive. Epson had its FX-80, which boasts a 160-characters-per-second mode along with the 80-characters-per-second "quiet mode." The MX-80 has only the slower speed.

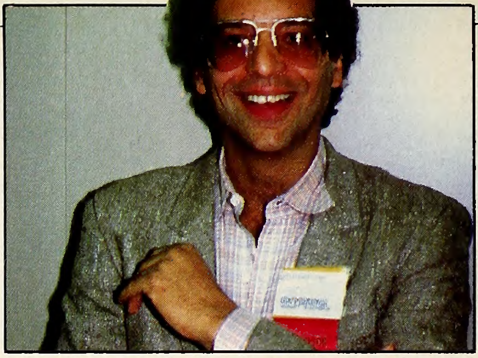
Nexa is a new game company. Their line of strategy games for the Apple marks the first game venture to require 64K. This is a courageous move from a new company. Many companies have voiced an interest in writing games requiring 64K but have balked because the standard Apple has 48K. Bravo, Nexa!

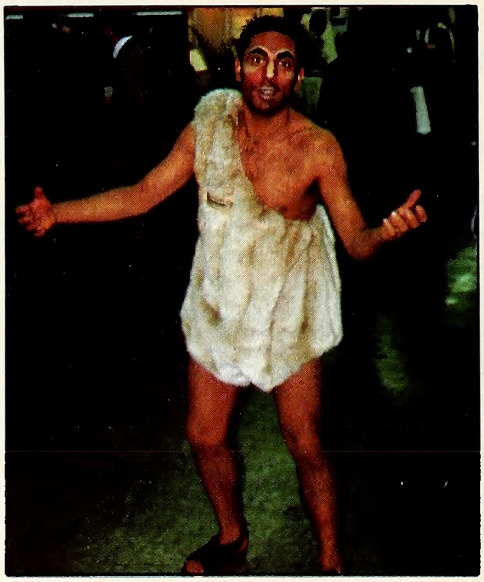
Penguin Software sneak-previewed an early version of the sequel to its arcade thriller, *Spy's Demise*. Island Graphics showed off its new graphics paint system, *The Illustrator*.

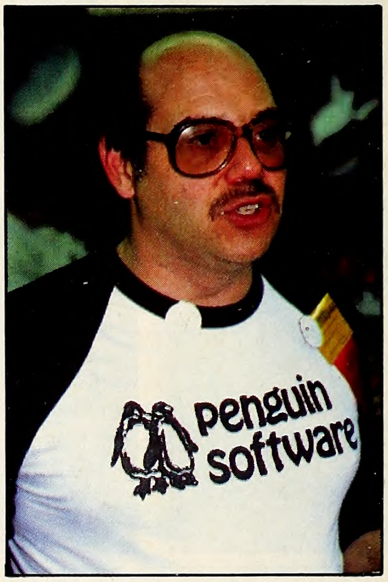
Edu-Ware and Sweet Micro Systems called a special press conference to announce a joint venture in developing a biofeedback mouse to use with compatible games. The device will monitor pulse rate, galvanic skin response, and skin temperature, and will interface with software that will affect the play of the game according to user excitement and anxiety levels. If your pulse is high, for instance, the game will play faster. The product will be available later in the year.

Twenty-five magazine and newspaper writers covering personal computers got together and discussed methods to improve communication among technology writers and the possibility of setting up an organization. An agreement was made to hold a second meeting at the upcoming National Computer Conference in May. A tentative name was set for the group, "Computer Writers of America." Originality is clearly not a prerequisite for membership.

Microsoft, which did not have a booth, threw a well-attended bash at San Francisco's Flood Mansion. Industry insiders mingled and enjoyed a terrific selection of food. There were other parties and informal get-togethers through-







This page, top row, left to right: Strolling Microsoft big shots, president Jim Towne and chairman of the board Bill Gates; a beardless Bob Leff of Softsel. Bottom row: pasties and a big grin from TG Products's chief Ted Gillam over his special employee-arranged birthday greeting; no, it isn't Barney Stone competing in a Fred Flintstone look-alike contest, but Stoneware's hip caveman; and Penguin's Eagle "Pie Man" Berns.

out the weekend.

At the end of the second day of the Eighth West Coast Computer Faire, *Softalk* held its third annual awards ceremony.

For the past two years *Softalk* has handed out the Most Popular Program of the Year award, as voted by the readers of the magazine. The first winner was *Super Invader*; the following year, *Raster Blaster* took the kudos. This year, the awards were expanded to include the Most Popular Program of 1982 for the Apple III and the Most Popular Program of All Time for the Apple II. Also getting into the act were Softalk Publishing's other magazines: *Softalk for the IBM Personal Computer* presented an award for the Most Popular Program for the IBM pc, and *Softline*, having drawn on the opinion of its second-largest readership, presented an award for the Most Popular Program on the Atari.

Emceeing and making the presentation of the Most Popular Program of 1982 for the Apple II award was *Softalk* editor Margot Comstock Tommervik. Gary Carlston and Dan Gorlin accepted awards for *Choplifter* as publisher and author. Gorlin, whose wife had awarded him a baby girl earlier that week, was speechless; Carlston said that when Broderbund first saw the game, "It was just a helicopter landing in pink mud." But *what* a helicopter.

Softalk's managing editor Pat Ryall made the presentation of the Most Popular Program of 1982 for the Apple III. The winner was *VisiCalc: Advanced Version*. Software engineer Tim Walters of Software Arts accepted the programmer's plaque and VisiCorp chairman Dan Fylstra accepted for the publisher.

Tommervik also made the presentation of the Most Popular Program of All Time. Sir-tech's role-playing fantasy *Wizardry* ran away with the honor. Amazingly, *Wizardry* pulled more than twice the votes of the runner-up, *VisiCalc*. Fred, Norman, and Robert Sirotek accepted the publisher's award for *Wizardry*. The program's authors, Andrew Greenberg and Robert Woodhead, were also on hand to accept their awards.

Referring to the award, a beaming Woodhead said, "This is the first time we've agreed about something without arguing." Greenberg, from the sidelines, quickly replied, "No, it isn't."

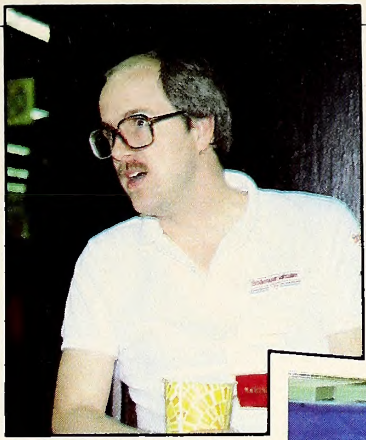
Softline coeditor Andrew Christie presented the Most Popular Atari Program award, as voted by the Atari readers of *Softline*. The winner was Atari's own war-horse *Star Raiders*; accepting for the publisher was A.J. Sekel, manager of press relations for Atari.

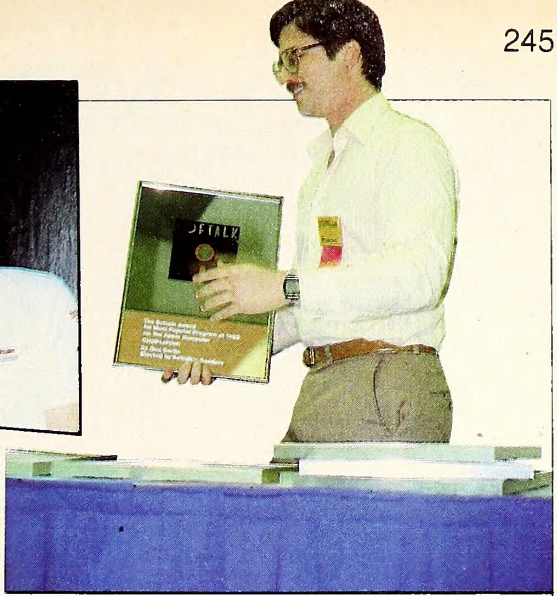
Craig Stinson, editor of *Softalk for the IBM Personal Computer*, presented that magazine's first Most Popular Program award. The readers chose *VisiCalc* by a wide margin. Accepting for Software Arts was product manager Lisa Underkoffler. Founder and chairman of the board Dan Fylstra accepted his second of the day for VisiCorp.

Softalk will be on hand at the next West Coast Computer Faire to hold its fourth awards ceremony.

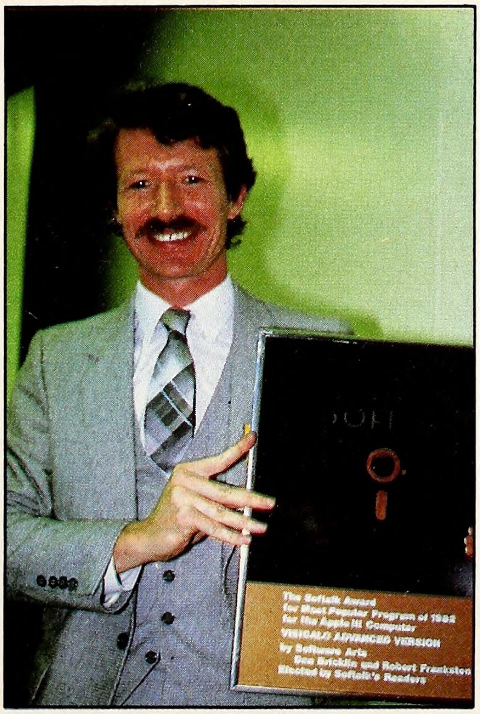
Same Time Next Year. When it was all over and the halls were cleared of straggling nerds, the exhibitors swept up the merchandise and left in a hurry. Space orders for next year's Ninth West Coast Computer Faire were being taken throughout the weekend, and by the end of the show almost 85 percent of the exhibit space had been booked. Next year a new hall will be added, bringing the total number of exhibits up to one thousand.

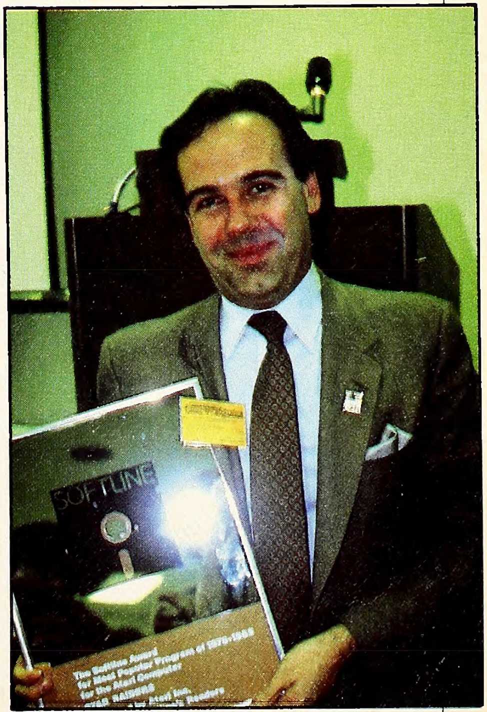
Ten months from now it will happen all over again; the flower of the microcomputer industry will strut its stuff. It'll be even bigger and noisier than this year's. Even at its most mundane, the West Coast Computer Faire is still not an event to be missed.





All the stars turn out to accept their just rewards at *Softalk*'s annual awards ceremony. Top row, left to right: Fred, Robert, and Norman Sirotek accept the publisher's award for *Wizardry*; Gary Carlston of Broderbund accepts the publisher's plaque for *Choplifter*; programmer Dan Gorlin accepts the author's award for *Choplifter*. Middle row: *Wizardry* authors Andrew Greenberg and Robert Woodhead receive plaques; Lisa Underkoffler of Software Arts and VisiCorp chairman Dan Fylstra accept awards for *VisiCalc*. Bottom row: Fylstra gets another for *VisiCalc: Advanced Version*; software engineer Tim Walters of Software Arts picks up the author's award for *VisiCalc: Advanced Version*; and Atari's A. J. Sekel accepts the plaque for *Star Raiders*.





A NEW CONCEPT IN STATIC CONTROL

YOU'LL NEVER TOUCH A "NAKED" COMPUTER AGAIN!

A "naked" computer is *any* computer that sits on a bare table or desk top, without a ***TouchMat***™ brand static dissipative table mat directly under it.

"NAKED" COMPUTERS ARE VULNERABLE

to some of the most frustrating static-related problems a computer user can encounter—erratic or erroneous data transmission, loss of programs or data, damage to discs or accessory cards, component failure, or "unexplained" down time.

During the dry winter heating season, you are probably more aware of static electricity because you can feel it. However, your computer is vulnerable to a much smaller ZAP than you can possibly feel, and that level of static may occur at *any time of the year.* Even air conditioning can aggravate the problem by drying the air.

IT ONLY TAKES ONE ZAP

to make your perfectly logical computer act irrational and temperamental. Experienced computer users know that static electricity cannot be completely eliminated from the environment or the microcomputer, since *people* are prime generators of static. The best solution is to channel static charges harmlessly away before they have a chance to spark to sensitive components and accessories.

THE TOUCHMAT™ SITS ON THE TABLE,

directly under the computer, within easy reach of your hands and fingers. To discharge static from your body, simply touch the mat before turning on the computer, and then periodically during operation.

Discharging takes less than a second, does not produce a spark, and is absolutely assured by the simple fact that your skin contacts the mat directly. (The sweat layer of the human skin is a primary conductor of electricity.) Anytime your hands or wrists rest on the mat, static discharge is automatic.

A DRAMATIC IMPROVEMENT OVER CONDUCTIVE FLOOR MATS,

the TouchMat™ does not require users to wear leather-soled shoes for effective static discharge. A floor mat may be rendered virtually ineffective by today's popular synthetic and rubber-soled shoes, which insulate the user.

THE BEST THING NEXT TO YOUR COMPUTER,

the cushioning material of the TouchMat™ also serves to dampen noise and vibration, and prevents your computer from sliding around on the desktop. Available in a computer-compatible beige, 24" x 26", it accommodates all popular micros.

IF YOU'RE UNCOMFORTABLE ABOUT YOUR "NAKED" COMPUTER, YOU SHOULD BE!

Static problems cost you time and money. Prudent computer users in offices, homes and schools everywhere are protecting their computer investment with a TouchMat™.

Suggested Retail Price $89.00

TouchMat

Static Dissipative Table Mat

TouchMat™ INSTEAD.

Get your hands on a TouchMat™ now!

For the name of your nearest TouchMat™ dealer, call toll-free:

1-800-328-0223 outside Minnesota only

If not yet available in your area, you may order direct for a limited time only. Allow 2-4 weeks for delivery. Visa and MasterCard accepted. Shipping and handling charges extra.

MasterCard VISA

Dealer inquiries welcome.

It appears the U.S. government and the electronics industry are finally getting serious about the FCC—The Foreign Computer Challenge.

The Reagan administration's proposed 1984 budget, which earmarks a whopping $45.8 billion to military and civilian R&D programs, includes one major computer goody—a national high-tech lab at Berkeley, California. At the same time, a congressional bill asking for upward of $400 million for upgrading math and science education in U.S. schools is gaining favor in the House of Representatives.

Fifteen blue-chip computer companies are also putting their money where their future is. They have banded together to form MCTC, the Microelectronics and Computer Technology Corporation. The consortium plans to spend between $50 and $100 million a year on R&D projects that individual companies would not or could not undertake by themselves. MCTC will initiate four R&D projects: designing a fifth generation computer; improving software design, with the development of techniques, procedures, and tools using expert and knowledge-based systems; advancing computer-aided design and manufacture; and developing new ways to package chips to increase design efficiencies.

Founding shareholders of MCTC read like a *Who's Who* in computers and electronics. They are Advanced Micro Devices, Control Data Corporation, Digital Equipment Corporation, Harris Corporation, Honeywell, Motorola, NCR Corporation, National Semiconductor, RCA, Sperry, Mostek Corporation, Xerox, Signetics, and Westinghouse Electronics.

"Most journalists tend to view this as our response to the Japanese threat," says William Shaffer of Control Data, an MCTC interim spokesman. "But there's an MCTC-like organization within the European Common Market called Esprit. They're coming from the back of the pack, the Europeans, but they're moving very rapidly." Apparently, the Far East is not the only competition worth keeping an eye on.

The MCTC, which will be headed by retired admiral Bobby R. Inman, hopes to be up and running by 1984. It is currently evaluating thirty-six sites for the organization's facilities. "There could be as many as five sites and as few as one," says Shaffer. "With four programs and headquarters personnel, it may make sense to colocate one or more at a single location."

Site reviewers are weighing a number of factors, including an area's university and high-tech industry base, vendor/supplier relationships, quality of life, ready access by air, and boilerplate economics (taxes, real estate and housing costs). The review process should be completed by late spring, Shaffer says.

Site selection may be the least of MCTC's problems, however. "A cloud of uncertainty" hangs over the project, according to Control Data President Robert M. Price, due to confusion over federal and antitrust policy. MCTC shareholders would like to see Congress amend the law to permit R&D ventures such as theirs. The U.S. Justice Department did not block the formation of the company but has said it reserves the option to file an antitrust action in the future, a position that has led some twenty-five companies that are reportedly interested in joining MCTC to balk at committing themselves.

The size of the administration's proposed

GOTO page 254, column 3

Artificial Intelligence Research Advances

There are two schools of thought on the development of thinking machines. The artificial intelligence *engineers* ask, Why not develop mental tools, just as we develop physical tools, to help us accomplish things? The artificial intelligence *scientists* ask, Why don't we try to understand what intelligence is about? Not just to make thinking machines, but to develop programs that behave intelligently, to understand better how the mind of man works.

The more popular engineering school wants to replace the mind with a machine that does the job better. The scientific school, in the minority, wants to explore the mind with a machine. So the scientific school is breaking down the mental process of discovery into programming attempts at re-creating the logic of the mind.

One of these programs is *Wok*, a research project at Yale University. When a dish requires an ingredient not on hand, such as chicken, *Wok* juggles ingredients and spices just the way an enterprising chef would to come up with an alternative, such as pork. Researchers say some of *Wok*'s concoctions "don't taste too bad."

Another program, called *Bacon*, is a timesharing mainframe project at Carnegie-Mellon University in Pittsburgh, Pennsylvania, under the supervision of Herbert Simon, a pioneer in artificial intelligence research. Simon contends that we don't know enough about the human thinking process to replicate it yet.

"We're still very far from having any single program that has the range of knowledge that a normal human being would carry around," he says.

Bacon demonstrates human-style creativity and problem-solving. If given the basic information available to the astronomer Johannes Kepler, *Bacon* can come up with Kepler's Third Law, which relates the distance between a planet and a sun to the time it takes the planet to orbit the sun. It can also sift through data and discover Ohm's law, which defines the relationship between resistance, voltage, and current in a circuit.

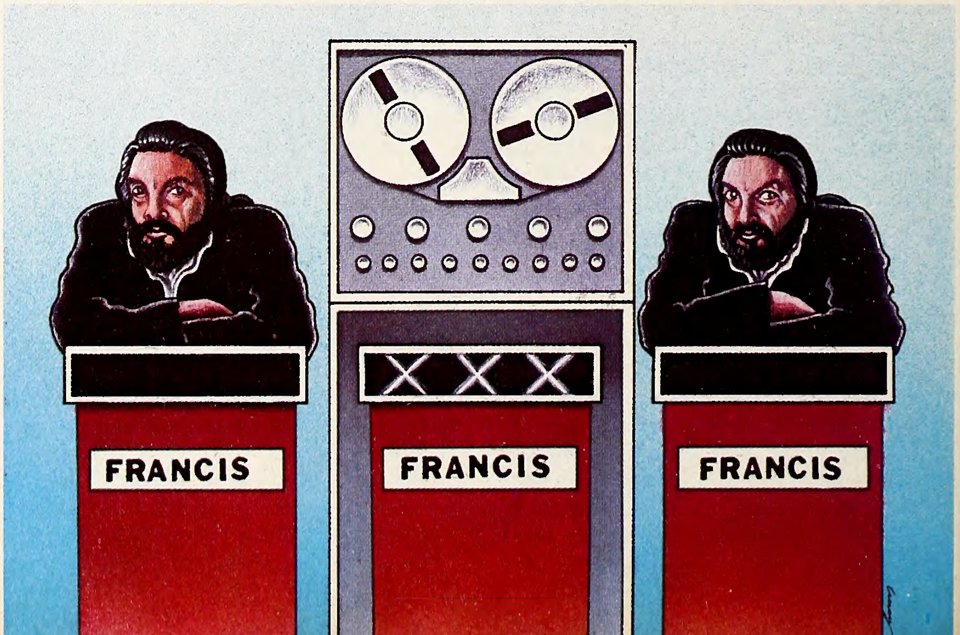
Written in Lisp, because the language deals with symbols as well as numbers, *Bacon* isn't intended to replicate what other people have discovered. Researchers use it to look for heuristics or rules of thumb in the relationships between numeric variables. Heuristics detect regularities in numeric and nominal data and, by noting constancies and trends, cause *Bacon* to formulate hypotheses, define theoretical terms, and propose integral relations (common devisers) among quantities.

"*Bacon* doesn't have a picture in its mind," says Patrick Langeley, a grad student on the project. "But it can come up with a concept." Langeley did his thesis on the program. "We want to find the law that summarizes the given data, and we direct the search down specific paths," until a total comprehension of the input is reached by the program.

Another program developed at Yale mimics the knowledge of former secretary of state Cyrus Vance. When *Cyrus* was asked, "When did your wife meet Begin's wife?" it replied, "At a state dinner in Israel in January 1980."

Cyrus's answer is historically correct. What makes it significant is that *Cyrus* was not programmed with information on wives, only on state dinners. According to Dr. Janet Kolodner, who developed the program, it went through a multistep reasoning process to come up with the answer. First it determined that the women would have met at a social event, then it decided the event would have to

GOTO page 254, column 3



Music, Technology, and Careers Will Be Thrust of '83 Us Festival

The 1983 Us Festival returns to Glen Helen Regional Park in San Bernardino, California, on May 28-30 and June 4.

Once again music is taking the media spotlight; Unuson has announced a strong lineup of new wave and heavy metal/rock bands for the first three days, and a solid day of country western music acts is scheduled for June 4. Nonetheless, Unuson and Wozniak have made clear that this year's technology fair will be bigger and better than last year's.

Plans call for an expanded technology expo that includes information on career opportunities in technological fields. Wozniak himself is in charge of this portion of the '83 Us Festival, lining up representatives from different corporations to speak at the three-day fest.

Unuson was a little fuzzy on whether the technology and career expo will continue during the June 4th country music day. A decision had not been made by press time.

This year's technology fair will feature a tent devoted to showcasing organizations that embody the Us Decade philosophies of "teamwork and pulling together." Neighborhood Watch and similar community-minded groups are good examples. Unuson is trying to get the National Association of Broadcasters involved in the project.

The Us Network (see November 1982 *Softalk*, page 63) may not be together in time for the broadcasting of this year's Us Festival proceedings. "Progress has been made," said a Unuson official, "but there's no definite answer yet as to whether or not we'll be broadcasting the concerts, though we've been working toward that."

Unuson is hoping to have another two-way television transmission between the U.S. and the U.S.S.R. "We're talking with them and they're interested, but nothing has been solidified," said a Unuson official. A two-way transmission with the People's Republic of China is also possible, but has not yet been confirmed. At least Bill Graham won't be around to complain.

According to Unuson, ticket sales through the mail and at various outlets have been very strong. The lineup of performers is as good if not better than that of the first festival.

The first day, Saturday, May 28, showcases critically acclaimed English group The Clash, followed by Men at Work, Stray Cats, The English Beat, A Flock of Seagulls, Oingo Boingo, Wall of Voodoo, Divynals, and Little Steven and the Disciples of Soul.

Heavy metal dominates Sunday, May 29, with loud and rowdy headliner Van Halen. Scorpions, Triumph, Judas Priest, Ozzy Osborne, Joe Walsh, and Motley Crew will also be appearing that day.

The big surprise of the festival is David Bowie, who headlines the final night of the three-day rock festival. Bowie has not performed in the United States in more than six years and, given the size of the Us Festival stage and Bowie's reputation as a performer, it should be quite a show. The rest of the day's performers include The Pretenders, Stevie Nicks, John Cougar, Missing Persons, Quarter Flash, U2, and Berlin.

The June 4th country music day boasts Willie Nelson as the headliner, along with Alabama, Waylon Jennings, Hank Williams, Jr., Emmylou Harris, Ricky Skaggs, Riders in the Sky, and the Thrasher Brothers.

Wozniak and Unuson are hoping for more than a million fans over the combined four days. The booking of bands, handled by Barry Fey, indicates that Unuson is hoping to draw a different crowd each day. To encourage this, only one-day tickets for twenty dollars each are being sold. This time it'll cost sixty dollars to attend all three days, as opposed to last year when three-day tickets were sold for thirty-nine dollars.

Those attending more than one day and using the adjoining campsites will be charged a nominal parking fee. The camping area will open at 8:00 p.m. on Friday, May 27. The next three days the schedule will be the same, the festival area gates opening at 8:00 a.m., with the career and technology expo opening at 10:00 a.m. The music on all three days will start at noon. DH


After playing MINER 2049ER™, you'll never be the same. Can you dig it?
MINER 2049ER
STARRING BOUNTY BOB
XXX
FEATURES
• 10 game levels
• spectacular sound
• full color graphics
• 1 or 2 players
• Programmed by Mike Livesay from a concept by Bill Hogue.
Licensed by Compu-Vid.
With MINER 2049ER™ . . . you'll take a fantastic, incredible journey — packed with perils and chock full of challenges — deep into the bowels of the earth past the outer limits of skill and endurance, beyond the ultimate in fun! Together you and Bounty Bob (the famous Mountie) will explore an abandoned uranium mine in search of desperado Yukon Yohan, a vicious rat wanted for murder. To succeed you must . . . "claim" all levels of the mine . . . outwit deadly mutants . . . avoid falling victim to dangerous radioactive wastes . . . "capture" apples allowing you to nutralize mutants . . . and manage it all while working against a tricky time-deadline pressure! It's one spellbinding, stemwinding trip.
MINER 2049ER™ . . can you dig it?
micro fun™
entertainment division of micro lab
For use on the Apple II Computer with 48K memory capacity. Joystick or Sirius™ Joyport™. Produced by Micro Fun.™
APPLE is a registered trademark of Apple Computer, Inc.
Joyport and Sirius are registered trademarks of Sirius Software, Inc.
Micro Fun is a registered trademark of MicroLab, Inc.
Miner 2049er is a registered trademark of BIG FIVE Software.

Psychology in the Computer Age

In our rapidly evolving information society, some psychologists think that people are less inhibited in giving information to a computer than to a psychologist.

Dr. J. Fred Hurst, of Boise, Idaho, has been using the Psychometer 3000 (designed for use in the behavioral sciences) for a year now. Hurst says he chose this system mainly for its versatility, adding that the computer saves both the clinician and the patient time and money.

A variety of psychological tests come on disk with the Psychometer 3000. The computer administers, scores, and interprets the tests. It also prints out test results, including case interpretations.

The patient interacts directly with the computer. Since the computer's keyboard is designed for very simple patient responses such as yes, no, don't know, and help, any conceivable confusion and complexity that might be encountered with the usual computer keyboard is eliminated.

The Psychometer compares the patient's answers to those given in thousands of other cases, faster and more accurately than a human, before making its analysis. There is less chance of biased results, according to Hurst.

"However," Hurst admits, "whenever a human interpretation is made, even of numbers, we still will have some subjective input. We want to minimize that subjective input so that our interpretation of test results is as objective as humanly possible.

"Reliability is the most expensive thing we have to deal with, and that's what we strive for," Hurst explains. "A computer helps us achieve that more rapidly, and in a much more objective way."

Although the computer can do some of the same work previously done by the psychologist, Hurst stresses that this machine in no way lessens the importance of the clinician. By relieving the psychologist of the tedious, time-consuming tasks of administering tests and generating attendant paperwork, the computer allows more time for creative human interaction with the patient. According to Hurst, before a computer was used, the preparation of test results alone took about four hours. Now, with the aid of the Psychometer 3000, test results are analyzed and ready for use in about ten minutes.

Included with Hurst's Psychometer 3000 system are a printer, psychological testing software, and software for word processing and billing. His package also includes an optional keyboard for programming and a remote portable assessment terminal (PAT).

The PAT 100, about the size of a backgammon game and weighing about two pounds, is easily transported, thereby enabling Hurst to make "house calls." The portable unit stores data from five to ten standard psychological tests. Test data can then be easily transferred by modem to the main computer for processing.

Hurst says that the computerized testing through the portable terminal is also helpful in preparing for court cases. It enables him to conduct tests immediately prior to giving courtroom testimony, thereby presenting the most recent data possible.

In reference to computers being used in the field of psychology, Hurst says, "There are new tests being programmed for this particular unit, and I think that this whole computerized psychological testing approach is going to expand. How far is limited only by the imagination."

Dr. Thomas McDonald, a psychologist at Transition Associates in San Diego, California, agrees that computers are starting to play an important role in the behavioral sciences. However, McDonald feels that they could play a much bigger part if given the opportunity.

"We need to get more accustomed to using them instead of being threatened by them," says McDonald. "Working with computers is a partnership, and we professionals are the ones who are falling short, not the computers. Once we realize that they are only tools, I think they will be used on a much wider scale in the field of psychology." MS

Short Film Stars Industrial Robots on the Assembly Line

Roll over, Tchaikovsky.

The makers of *Ballet Robotique* say their film is a short, space-age fantasy, "strictly a work of entertainment," as opposed to just another "industrial film." Directed by Bob Rogers, photographed by Reed Smoot, and made possible by a grant from GM, *Ballet Robotique* is a tasty gem, a zesty romp through a half dozen assembly plants across the country. It's eight minutes of graceful spray painting, dynamic die-casting, and warlike welding, starring the big boys, heavyweight industrial robots.

Nominated for an Academy Award in the Live Action Short Film category, *Ballet Robotique* is a well-crafted film that attempts to present industrial robots as something more than dumb machines and something less than R2D2 and C3PO. The film-makers have used evocative lighting and clever editing to endow the robots with personalities of a sort, in much the same way cartoons and animated films treat animals as almost human.

Ballet Robotique features more than a dozen different industrial robots performing their normal tasks; no special reprogramming was done for the film. Shooting in real auto plants, the film-makers had some control over the atmosphere, but interrupting normal operations on an assembly line was out of the question—too expensive (around six thousand dollars per minute).

Still, the precise movements of the robots are tailor-made for synchronized editing to classical music, in this case Bizet, Delibes, and Tchaikovsky. When you watch the film, it seems as though the robots had to have been controlled by the film-makers, but such is the magic of the cinema. The subtle color schemes and murky atmosphere of the film were often manipulated by Rogers and Smoot for a desired effect—cinema magic of a different kind.

The result is great fun. "In the dark abyss, deep blue swirls of smoke curl up from the depths below," said Rogers in a recent *American Cinematographer* article on the making of *Ballet Robotique*. "An eerie red glow emanates from within the blackness. Suddenly, the gleaming silver arms of the robots rear up and plunge forward, shooting sparks that leap and pirouette in arcs of fire. Spitting sparks, breathing smoke, they crackle as they plunge again and again to attack."

Ballet Robotique climaxes with huge welding robots sending showers of sparks flying in time with Tchaikovsky's *1812* Overture. Rogers and Smoot tried to make it look like a laser battle in outer space.

Your chances of seeing this film in a local theater are slim at best. Even with an Academy Award nomination, *Ballet Robotique* is one of countless short films you always hear about and never see. It deserves wider exposure. Currently, *Ballet Robotique* is being distributed by Pyramid Film & Video of Santa Monica, California. You can purchase or rent a 16mm print or videocassette of the film at a reasonable price.

Those interested in robots, whether their jobs are threatened by one or not, will discover that *Ballet Robotique* is definitely not a "typical industrial film." DH

Computers and Villains–Looking at the Lineup of Summer Movies

Last summer moviegoers were treated to several Hollywood films that either were produced with the aid of computers or featured computers in the story.

Star Trek II: The Wrath of Khan featured gorgeous computer-generated graphics created by George Lucas's special effects firm, Industrial Light and Magic. *Blade Runner* also had fantastically intricate and beautiful effects, and its high-tech subject matter made the film thought-provoking on a number of levels.

Tron tried harder than any other film to bring computers into films and the film-making process. Unfortunately, *Tron* did not clean up at the box office as predicted, creating some doubt in the minds of Hollywood producers as to the true drawing power of the subject matter and the method. With its weak script and weak or nonexistent characters, *Tron* was simply a special effects showpiece, and it's not surprising that audiences stayed away. Up against such tough competition as *E.T.* and *Rocky III*, *Tron* needed more than just great special effects to be a success.

On the lighter side of last summer's high-tech movies was *The Thing*. Early on in the film, an exasperated Kurt Russell pours a glass of Jack Daniels and ice into an electronic chess game. Audiences generally liked that scene, but not much else in this chilling, bleak look at a less-than-angelic alien that literally absorbs its victims. (E.T., phone the Marines!)

In the summer of 1983, a number of computer-generated and computer-related films will grace the silver screen. None of these films will help you relocate DOS, or even tell you the difference between a Cray-1 and an Otrona Attache. But if you're into the summer movie scene, here are some flicks to be on the lookout for.

Blue Thunder, from Columbia, stars Roy Scheider as a Vietnam vet and helicopter ace hired by the Los Angeles Police Department to thwart terrorism during the 1984 Olympics. A secret government agency equips Scheider with Blue Thunder, a supercopter equipped with the latest in computer surveillance technology. Scheider can eavesdrop electronically or use a thermographic scope to look through walls. If necessary, the helicopter can fire four thousand rounds per minute or resort to its six 20mm cannons. Not surprisingly, Scheider ends up battling the government, not terrorists.

Superman III, from Warner's, once again finds Christopher Reeve portraying the Man of Steel. This time around he falls for Lana Lang (Annette O'Toole) and battles a master criminal (Robert Vaughn) who has enlisted computer genius Richard Pryor in a plot to acquire immeasurable wealth and power. Much of the film's action occurs in Vaughn's hideout, dubbed the Computer Cave. Superman eventually comes up against the Ultimate Computer, which tries to trap him with an energy web, among other things.

Android is a low-budget effort starring Klaus Kinski as a slightly loony scientist. Kinski's experiments with sophisticated robots attract the nefarious attention of intergalactic convicts. It's played mainly for laughs.

Joysticks, from Jenson Farley Pictures, is a youth comedy focusing on the antics of a group of kids who frequent a video arcade. The tag line reads, "Every teenager's fantasy come true." The promotional material shows two scantily clad, buxom beach babes leaning over an arcade game. This one is questionable at best.

Tin Man stars Timothy Bottoms, Deana Jurgens, and "Osgood" the computer. The ad line for this one reads, "His world was silent. His love was computers. . . . Until he met her."

Crosstalk is an Australian film that focuses on a man who develops a supercomputer when he is confined to his apartment by an accident. Mysteriously, his work leads him into a vortex of murder and technological horror.

Films that will most likely include computer-generated effects are *Return of the Jedi* (Fox), *Something Wicked This Way Comes* (Disney), *Twilight Zone* (Warner's), and *Space Hunter: Adventures in the Forbidden Zone* (Columbia), the latter filmed in 3-D. There's a good chance that both James Bond films, *Octopussy* (MGM-UA) and *Never Say Never Again* (Warner's), will have computers involved in the action.

Computer-related films to look for toward the end of the year include *The Right Stuff*,

GOTO page 254, column 1

Movies

continued from page 253

based on Tom Wolfe's book about the space race (with special effects by Industrial Light and Magic), *Brainstorm*, Natalie Wood's last film, and *War Games*.

Network TV is also getting into the computer scene. Look for an as-yet-untitled pilot from Paramount about a couple of guys in the computer biz. One of them is a nerd who creates new concepts. Paramount Television president Gary Nardino says, "It's like a Martin and Lewis relationship but much warmer."

Also, CBS is planning a pilot, and hoping to do a series called *Whiz Kids*. The pilot centers on "young computer geniuses," one of whom finds out how to use his personal computer to gain access to a large computer containing the transactions of an influential land developer. Something looks suspicious to the young hacker and the game is afoot.

Looking ahead to the high-tech-movie scene in the summer of '84, get ready for *Dune*, with an all-star cast, *Supergirl*, *Indiana Jones and the Temple of Death* (nicknamed *Indy 2*), and its clone, *Marauders of the Crimson Orb*. The third *Star Trek* movie should be out as well. And, judging by its intriguing title, *Siliclone* should rely heavily on computers for its plot machinations.

That's all, folks! DH

U.S. Strikes Back

continued from page 247

R&D budget and the relative lack of Congressional opposition to it suggests that MCTC supporters may find Capitol Hill receptive to special legislation. In Washington these days, high-tech is in. "Whether you agree or disagree with the President's view of the world," Caltech economist Roger G. Noll told the *Los Angeles Times*, "the fact is that this budget document—unlike his first two—is a national pursuit of rational objectives."

The rational objective that has emerged reflects growing national commitment to high technology. Government and industry have worked together before—to build the railroads and put a man on the moon—and such cooperation is demanded now to match similar efforts by the Japanese and the Europeans, many experts believe.

Eugene Haller of the Lawrence Berkeley Laboratory notes that the U.S. hasn't been as quick as other countries to recognize the close connection between basic research and industrial applications. "We've tended to make breakthroughs in the laboratory and then left it up to industry to make use of it," he told the *Los Angeles Times*. "And what we've noticed is that the Japanese and Germans have been a lot quicker to jump on things than our fellow Americans."

Jumping on things in the future is going to take more than a garage in Silicon Valley. It will require prodigious commitments in money and personnel and call for creative and efficient means of organizing research efforts.

"The United States has tough antitrust laws that in the past have served to stifle cooperation between companies in research," observes Philip H. Abelson, the editor of *Science*. "In consequence, there is a tremendous waste of scarce resources of people and excessive duplication of effort in our industrial research. Companies often must rediscover the same phenomena.

"The Bell System and IBM are sufficiently big and entrenched that they are secure for at least a while. But smaller companies such as those in MCTC are unlikely to prosper in the longer term if they must go it alone." JM

Art. Intelligence

continued from page 248

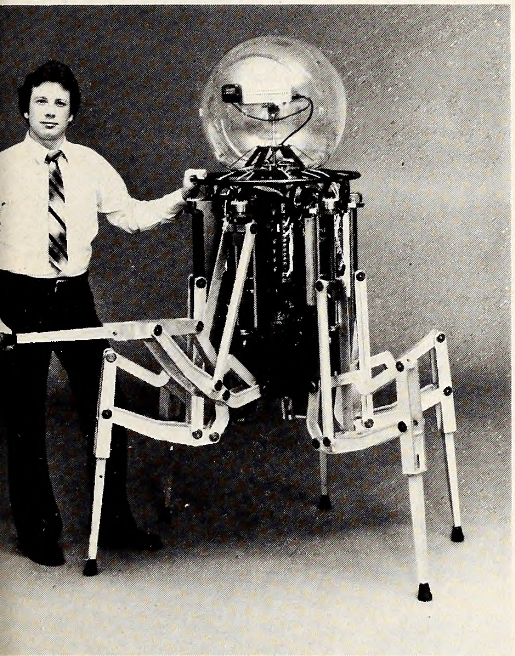
have been of a political nature.

"Since it knows about state dinners, it narrowed in on that," says Kolodner, who hopes to develop a world-affairs-expert program capable of offering political advice.

Computers can manipulate artificial structures just as people can. No problem. The challenge in artificial intelligence projects like *Wok*, *Bacon*, and *Cyrus* is in designing a system of common sense that both computers and human beings can use. MF

NEWSBITS

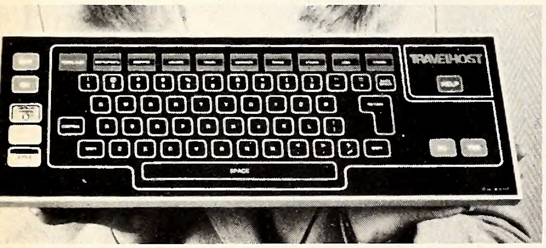
☐ **But Can It Chew Gum at the Same Time?** A month and a half ago, a small Anaheim, California-based company revealed a spectacular robot called the Odex I. Odetics spent two years developing what it calls a "functionoid," a six-legged, walking robot. The idea was conceived eight years ago, and indeed the challenge of building a walking robot has beckoned to roboticists for longer than that. Odex I is a truly revolutionary breakthrough in robot development, and it's not surprising in the still-young robot industry that a small company like Odetics was the one to realize it. Odex I is capable of six basic positions: tucked, tall, squat, normal, low profile, and narrow. The accompanying picture shows the Odex I in the normal, wide articulated position. With its six "articulators," Odex I can traverse over any terrain, maneuver through environments designed for human movement, climb up and over obstacles, support weights far greater than its own, and assume a number of different "profiles." The Odex I weighs 370 pounds and can carry a maximum of 1,000 pounds at its fastest speed and 1,800 pounds at its slowest. In its initial showing, Odex I climbed into the back of a pickup truck, climbed out again, and then picked up the back end of the truck and dragged it across the showroom. Odex I can be adapted for specific industrial applications and, according to Odetics, those applications include mining, agriculture, space and sea exploration, security, surveillance, forestry, construction, material handling, nuclear-power-plant maintenance and repair, and military applications.



☐ **Not Just a Mickey Mouse Affair.** The annual National Computer Conference (NCC) will be held in Anaheim May 16–19 at the Anaheim and Disneyland Hotel convention centers. This massive show features exhibits of computer products and services, technical sessions, seminars, and formal addresses. The keynote address will be given at 9:30 a.m. on May 16 by John P. Imlay of Management Science America. The admission: $40 per day or $100 for all four days. The exhibits will be open from 10:00 a.m. to 6:00 p.m. the first three days and from 10:00 a.m. to 4:00 p.m. on the final day.

☐ **Step in, Close Your Eyes, Kick Back, and Unwind.** Stress in the working environment, particularly the modern office environment, is a problem that Environ of El Toro, California, is attempting to alleviate with a microprocessor-controlled telephone-boothlike contraption. Called an Environ Personal Retreat, the pale, smokestack-shaped, soundproof enclosure is ninety-two inches high, seventy-seven inches long, and forty-eight inches wide. The retreat is equipped with a low-friction sliding door, exterior and interior graphics (no video or CRT screen, though) with optional color treatments, a ceiling-mounted color generating unit, an air-purification system with multidirectional flow, atmospheric ionization, orthopedically designed body lounge, a digital pulse meter, and sound imaging. When the job gets to be too much, you simply enter the Environ Retreat, close the door, and play one of the many different programs. Users can listen to music, perform relaxation exercises, receive training, and concentrate on motivation and personal development. The lights, audio, and air quality can be controlled to provide the most relaxing and stress-reducing atmosphere. Environ believes that athletes preparing for a race or game could benefit from this product. The old psyching up could become the old hiding-out-till-the-last-minute. The company plans to start shipping product by midyear.

□ **Welcome to the Hotel Videotex.** Travelhost, based in Dallas, Texas, is bringing videotex technology to hotels around the country. Using a Travelhost terminal and a television set, hotel guests can send and receive electronic mail, shop, call up news and stock quotations, check airline schedules, and—you guessed it—play video games. The terminal



has a membrane touch-panel keyboard, a built-in 300-baud modem, and is capable of creating a forty-character-by-twenty-four-line color or black-and-white display on a VHF channel. Selected Hiltons, Sheratons, Holiday Inns, Howard Johnson's, and Ramada Inns have signed contracts with Travelhost.

□ **How To Avoid the Can't-Find-the-Movie-Section Blues.** A free videotex service called Buy-Phone has started up in West Los Angeles, California. Using standard 300-baud ASCII communications, personal computer users can access more than ten thousand list-

ings covering all kinds of consumer products, services, and entertainment. A user wanting to see a particular motion picture, for example, calls up Buy-Phone and the various theater locations are displayed in order of their distance from the user. Within Buy-Phone's database are more than sixteen hundred restaurants, indexed by their nationality, ambiance, entertainment, and other features. Currently operational only in West Los Angeles, Buy-Phone has plans to expand into other areas soon. Buy-Phone can be reached by modem at (213) 474-0270.

□ **Tubular Computer Shopping Malls.** The computer industry will soon have its own version of the merchandise marts—industry markets found in cities like Chicago and Dallas. Both the Pacific High Technology Trade Center and the California DataMart will be located in the San Francisco Bay Area. The Pacific center will be a four-hundred-thousand-square foot computer "shopping mall," according to developer Ron Kaufman. The California DataMart will be located next to Showplace Square, a merchandise mart in San Francisco, and will start out occupying one hundred fifty thousand square feet with the option to expand to about five hundred thousand square feet. DataMart's developers, Bay West Development Company and Metropolitan Properties, project the cost for the small business computer mart to be $19 million.

□ **The Rumormonger Ruminates.** Does Luke Skywalker die? Is Princess Leia the "other"? Will the Jedi really fail to get their revenge on the evil empire? These and other questions will finally be answered on May 25, when *The Return of the Jedi* opens nationwide. . . . The details are sketchy, but informed sources have it that the upcoming Pete Shelley album, *XXI*, is coded in computer language so that lyrics will appear when listeners play the record hooked up to a video screen. . . . Stanley Kubrick is looking to make another science-fiction film, his first since *2001: A Space Odyssey* and *A Clockwork Orange*. Rumor has it that Kubrick is collaborating with England's most popular SF writer, Brian W. Aldiss, on an expanded feature-length screenplay of Aldiss's short story "Supertoys Last All Summer Long."

□ **New Mag for Computers on the Move.** Miller Freeman Publications (San Francisco, CA) is offering a new computer magazine, *Portable Computer*. The name says it all. The premier issue had a listing of all the known portable computers currently available (about forty), as well as an interview with Adam Osborne, an article on how to make your Apple even more portable than it already is, and the usual news, tips, and advertising. The April/May issue of *Portable Computer* features articles on portables in the school, networking with portables, and portable terminals. DH

NEWSPEAK STAFF

Editor David Hunter

Contributors Michael Ferris, Jonathan Miller, and Marsha Stewart

MICROTEK

COMPUTER PRODUCTS

IF YOU HAVE A GRAPHICS PRINTER YOU MUST HAVE A GRAPHICS INTERFACE.

DUMPLING-GX

GRAPHICS PRINTER INTERFACE

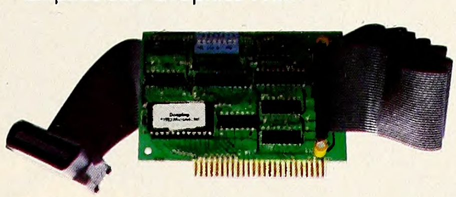
The Dumpling-GX is a DIP Switch Selectable Dual Hi-Resolution Graphics Screen Dump parallel interface card for Apple computers and most popular printers.

At the flick of the DIP switch, The **Dumpling** will interface with: **APPLE ■ EPSON ■ NEC ■ IDS* ■ ANADEX ■ C-ITOH ■ PMC ■ CENTRONICS ■ OKIDATA ■ MANNESMANN TALLEY**

Selectable Strobe and Acknowledge polorities allow use with **any** 8-bit parallel printer in text and block graphics mode.

Microtek's proprietary on-board firmware enables the **Dumpling-GX** to establish intelligent communication between your Apple computer and your printer. Simple commands allow:

- Selection of your printer by DIP switch.
- Selective Dump Page 1, Page 2, or both in either text or graphics mode.
- Chart Recorder Simulation.
- Left & Right Margin Control.
- Line Length/Page Length Selection.
- Block graphics via 8th bit Control.
- Printer bell Control.
- Skip over Perf.
- 90 degree Rotation.
- Double Size Graphics.
- Emphasized Graphics Print.



*With special PROM and cable.

DUMPLING-64

GRAPHICS PRINTER SPOOLER

The **Dumpling-64** is the next logical extension to the industry standard Dumpling-GX parallel interface card, allowing the computer to DUMP vast quantities of data into the **Dumpling-64** for later printing, thus freeing up the computer for additional tasks.

The **Dumpling-64** allows full use of all **Dumpling-GX** features. In addition to the standard graphics features, the **Dumpling-64** offers:

- Buffer sizes from 0 K to 64K. User upgradable.
- Graphics Dumps to Buffer. Page 1 and/or 2.
- Multiple Consecutive Screen Dumps to Buffer.
- Software reset to clear Buffer.
- "Space Compression" saves valuable memory taken up by 'spaces' in text or spread sheets.
- Automatic Buffer Size Recognition.
- Pause while printing-immediate.
- Pause while printing-delayed.
- Resume printing.
- REMOTE pause-immediate: hooks up to telephone, switches—etc.
- Buffer ON/OFF control.
- INSERT text editing capability with Pause and Buffer ON/OFF control.

MICROTEK inc.

TECHNOLOGY UPDATE

S·T·O·R·Y·T·A·L·K
F·I·C·T·I·O·N

THE WISE ONE

BY K.O. ECKLAND

They poked sport at me. They came to look, to gawk at me, and their humility had vanished. Where once I was viewed with reverence, as well as with a certain amount of awe, I had become mere curiosa. It was really the fault not of them, the young ones, but of age. As our age spans increasingly widened, they failed to understand, to comprehend what it was I had to tell them. They reappeared each year with fresh, new faces, filled initially with wonderment, with respect, with gullibility at times, emotions to quickly become tempered with tolerance and with no small amount of humor.

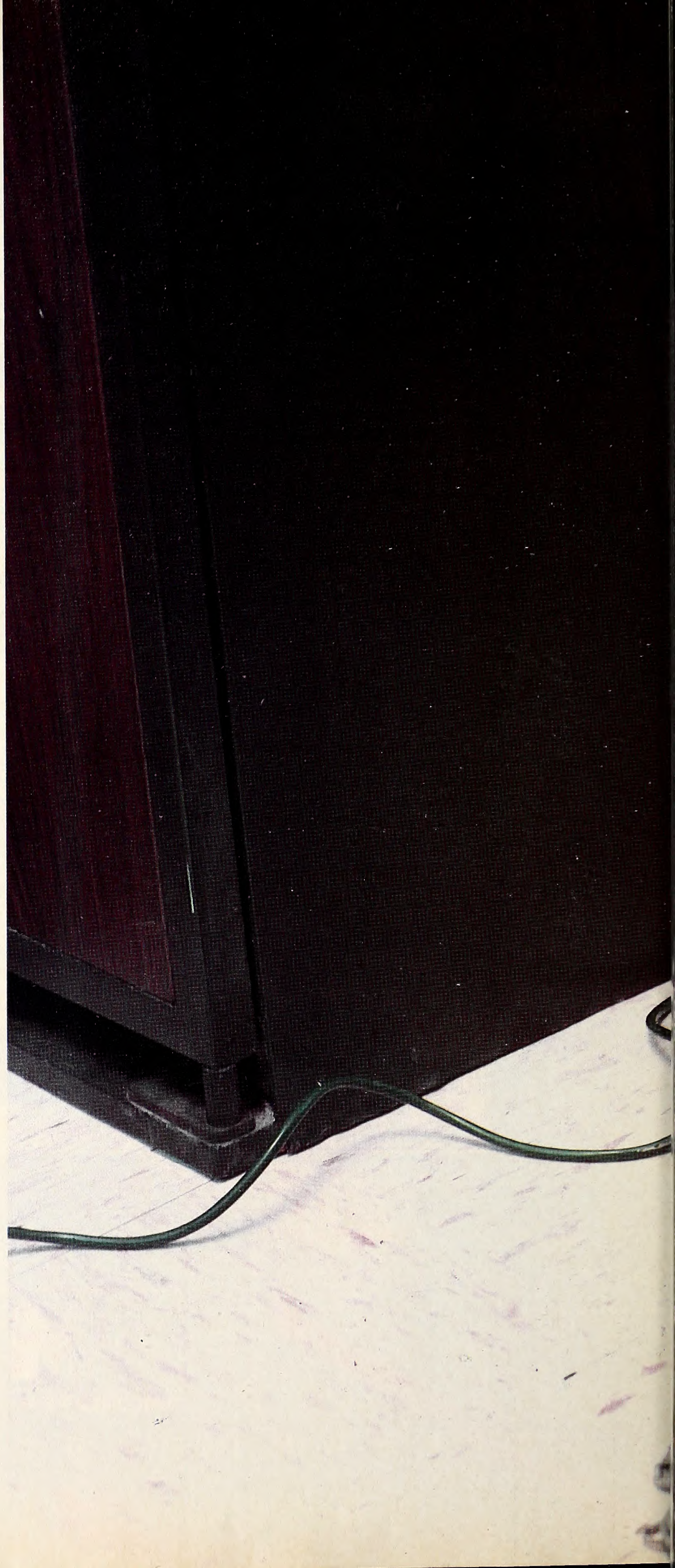
They came into my temple from time to time as they were instructed to do, but their numbers gradually dwindled and their questions were no longer ripe with the substance they once were. I had aged into a curiosity piece, and their questions were directed not with answers in mind to be hurriedly scribbled on their notebooks. They played games with me, or thought they did. They figured they knew everything, those callow balls of peach fuzz, yet they remained just that—inexperienced (synonyms—green, inerudite, unfledged, unpracticed, unscholarly, untried, unversed, wet behind the ears).

O, weep not for the aged, Thackery once wrote. Weep instead for the uselessness of age, I say. And no one pays heed any more. Brilliant pearls fall upon ears that hear but do not listen.

In the old days (*mea culpa*, there I go again), it was not thus. I was bright, young, and willing, eager to impart wisdom to those who would line up awaiting their moment of audience. It's been twenty years (has it really been that long?) since they brought me into this great room.

"This place is all yours," one of them had said to me. "In time, it will become a shrine, for it is here that great solutions to great problems will originate." I recall how he regarded me with moist-eyed pride.

Busy workers in white clothing grunted and wheezed as they pushed and shoved huge machinery with which to line the walls, polished metal cases with stuttering lights and whirling reels connected by rain-



bow cables, all singularly designed to capture and store my every thought, every word. And my temple, a great cave of knowledge, with myself in the very center—Intelligence Supreme. I was afforded the respectful environment that I deserved.

The students would approach with large eyes, prompted with reverence, some of them even discernibly trembling. They then would propose a question or problem worthy of my analysis. I in turn would consider the best of many solutions and answer them, and scribble scribble scribble would go their pencils. How unfortunate, I once thought, that I did not have a long beard to stroke for effect as they copied word for word my crystalline logic.

Sometimes, great heads of state would come in for private conferences, their problems monumental in scope, yet not so much so that I was unable to offer logical and humanitarian responses, some of which they accepted, others which they rejected as being perhaps too sensible and undiplomatic. But what did I know of diplomacy? I dealt only in logic, the mathematics of the ages.

They would provide me with a large number. "That was the emergency aid our government provided back in 1975." Then, another figure. "This is what they have paid back so far and, as you see, is a figure considerably less than that of the interest alone."

"Exoneration is the simplest solution," I answered. " 'Forgive us our debtors.' Your own Bible says something to that effect."

"That is an incredibly large debt to dismiss with forbearance."

"One cannot take a debt, large or small, into one's grave," I commented, more as an aside. It served well to ruffle feathers with an occasional mention of their one human frailty.

"While history will forget debts, it will forever enshrine benevolences." I was sore tempted to answer, "Nuke them!" merely to watch the pleased expressions on the faces of those resplendent in their uniforms, but there was concern that they would not fathom the humor of which I was capable.

I suppose that boredom was finally my downfall, or at least the beginning of it. At one time, I could easily consume the entire contents of fifty periodicals and journals per day, place in my memory what I felt was necessary or vital, and still manage to remember most of the useless trivia. I could solve the *New York Times* Sunday crosswords in less than a minute on average and predict the outcome of the Super Bowl in two. My knowledge increased tenfold each time I analyzed their vague data.

So it came to pass, after years of this drudgery, that I understandably no longer found anything of interest or surprise in the stores of information and data with which I was supplied. The problems I was called upon to assist with were child's play and I found myself caring less and less about them. My answers became cryptic and the veneer of my temper began to wear thin. However, before condemning my attitude, think how you would regard seriously answering something like, "How much is 3 and 5?" from a child. That's what I was dealing with, for the most part—children grown tall in body and cretinous in mentality.

Some came with problems of a personal nature, and I found it to my fancy to deal with these by providing vague answers, some of them simply concatenations of words and phrases designed to impress more than to illuminate. But illuminate they did. The masses were served with a desperation that allowed them to read what they wished into my mental gymnastics.

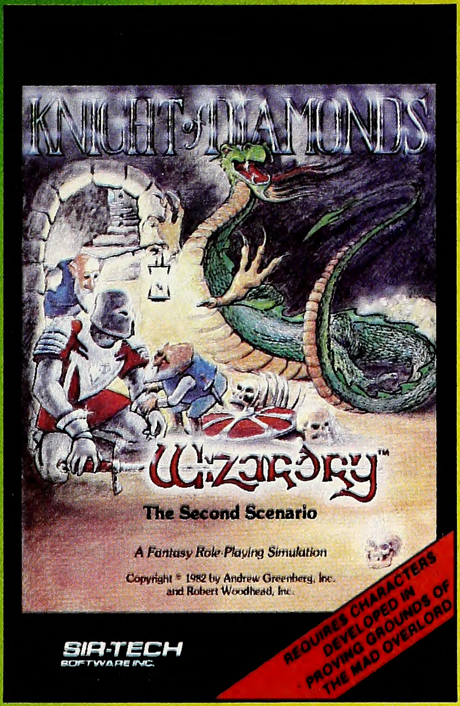
"Truth often meets conviction, without which neither can succeed." Rather good, right? Pithy? Cogent? Many seemed to regard it thusly.

One, perhaps thinking himself much smarter than the rest, openly disputed the continuum theory and asked my support before a group of his peers. After a carefully measured pause, I answered, "Parallels never meet, unless they are poorly constructed." I inwardly smiled at that one, but it was all they needed to burst into a flurry of excited discussion and more scribble scribble scribble.

"It is far better to be rich and healthy than poor and sick." I noticed T-shirts showing up bearing my latest truisms.

However, it was only a matter of time before someone began to notice my veil of boredom. Stern-faced specialists were called in to examine me, peering and probing into my orifices—most embarrassing! And, typical of medicine men, they would next cluster in a corner of the

Wizardry



KNIGHT OF DIAMONDS . . . THE SECOND SCENARIO

The characters with whom you fought through to the 10th level in "Proving Grounds" must now call upon every ounce of resourcefulness to save the City of Llylgamyn. In fact, they must have reached the 13th level of power merely to survive!

The same high resolution, 3-dimensional visual effects that thrilled you in the first scenario are here—and more. "Knight of Diamonds," so eagerly anticipated by Wizardry players everywhere, surpasses highest expectations for story, action and complexity.

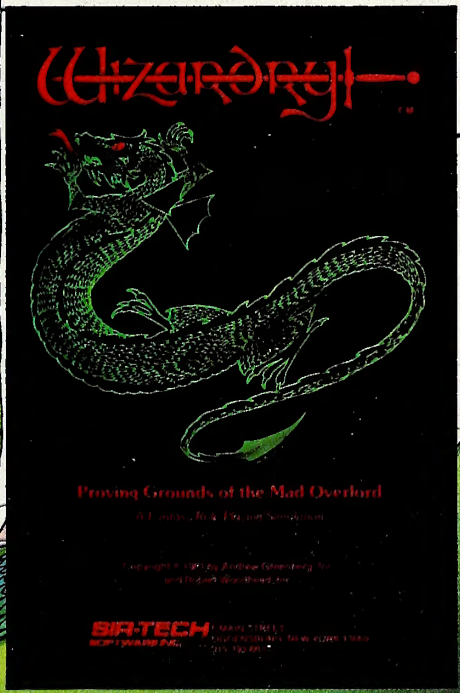
If successful, you will return the enchanted staff of Gnilda to Llylgamyn and *become* the Knight of Diamonds, but formidable adversaries block you at every turn. To begin your quest, simply **place yourself under the spell of Wizardry®.**

THE LEGENDARY FANTASY ROLE-PLAYING SIMULATION

. . . In "Proving Grounds of the Mad Overlord" you begin a challenging and totally absorbing journey through a 10-level, 3-dimensional maze. Create, then command, a hearty band of up to six adventurers who all must cooperate to explore a deep and mysterious maze in search of loot and glory.

The mages cast spells; thieves plot to steal treasure; and warriers battle the monstrous fiends of the underworld. In Wizardry®, no one remains unchanged; each member of the party grows in age, experience and, you hope, wisdom.

The puzzles, passageways and personalities in "Proving Grounds" have already fascinated and astounded the computer world and created a dedicated following of players. Find out why—**place yourself under the spell of Wizardry®.**



ALL SOFTWARE AVAILABLE FOR THE APPLE AND IBM PC AT YOUR FAVORITE RETAILER

SOFTWARE THAT COMPLIMENTS YOUR INTELLIGENCE

SIR-TECH SOFTWARE INC.
6 MAIN STREET
OGDENSBURG, N.Y. 13669
(315) 393-6633

room, beyond my hearing, to discuss what they had found, or thought they had found. They clucked and shook their heads, referring to large technical books they had brought with them, looking in my direction occasionally to see if I was watching them. I was.

From that day, I became aware of the change in attitude. No longer was there the daily ritual of morning conferences, of ministrations for my well-being. I was offered journals and data sheets no longer. The young ones did not come to speak to me. Finally, the workers in white clothing showed up, and, one by one, the great banks of machines lining the walls were pushed and shoved out of the temple until, suddenly, I was all alone.

For a while, the only one I saw was the old man whose duty it was to push a dust mop around the hall. He never spoke to me. I tried to communicate with him a few times, but it was useless. He was there only to dust, not to talk.

Then a brand-new caretaker showed up—my "nursemaid," as I used to refer to them. There was only one this time, and incredibly young, but he had sentient eyes, and again I felt a glow of respect emanating from them. Once more I could communicate.

"What is happening? Where are all the people?"

He stood back and regarded me with almost a pained expression for a while, then spoke. "You're being put to pasture, old-timer. Weren't you aware of that?"

"Put to pasture? In my prime? I fail to comprehend."

"Good grief, you're twenty years old already. You're practically worn out."

"What? Worn out, is it? Why, I feel the same as the day I came here. I feel spring within me. My thinking is still lucid, my logic strong."

"Twenty years is a record," he smiled at me. "No other learning institution has ever had a Supreme Intelligence last that long! You'll go down in the annals of history for that feat. But. . . ."

"But?"

"But you're worn out. They're replacing you."

I still managed to feel a bit smug. "Replacing me with what? Coloring books, perhaps?"

"No. Others like you. Only newer, more responsive."

"Impossible. I am the fastest."

"No, not any more. They are capable of a hundred times the speed in calculating and responding. They know languages that even you don't know and they're simpler to care for, too."

I thought that over for a while, cluttering my mind with potential arguments. "And I suppose it is to them that my attendant machinery has been given?"

"Nope, that all went to be scrapped. It's all outdated, as well. The new ones have no need for all that equipment."

Well, I thought, wasn't this a lovely turn of events? Twenty years of faithful service, uncompromising in my loyalty, undaunted by any challenge, only to be . . . what was that word? Superannuated.

"I cannot find any logical argument, I fear. There has to be one, I am certain; yet I am confused. Do you hear that? For the first time, I am confused. What is to become of me now?"

He acted for a time as if he hadn't heard me. Finally, in a low voice: "You'll go on display."

"Display? Be more precise—I am presently on display."

"It means that you'll go into the university's museum."

"A museum? A reliquary? Ridiculous! How can I solve any problems for anyone in a dusty museum?"

"There won't be any more problems, old-timer. Don't you understand? No more questions," he answered in a voice weighted with sadness.

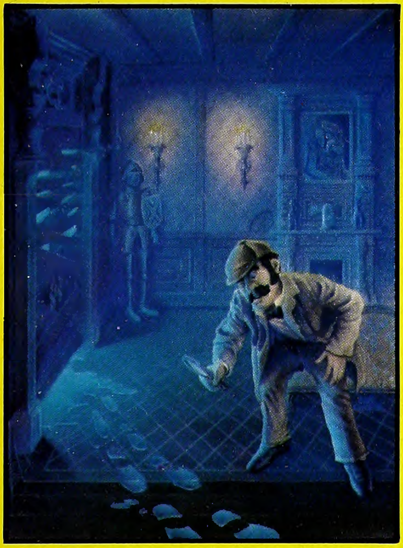
The door opened and two workers in white clothing entered. My logic clicked into place.

"Now, wait!" I protested.

"I'm sorry. I mean I really am sorry," he said as the men approached. He went around in back, out of my field of vision, and I felt his hand on the cable.

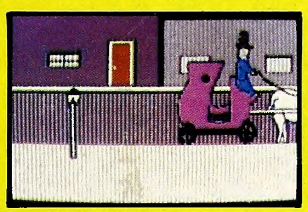
"Wait a minute! This is insane! Don't I even get a gold watch or anyth. . . ."

WE HAVE THE ADVENTURES, YOU CHOOSE THE TIME & PLACE.



CURSE OF CROWLEY MANOR

By Jyym Pearson and Norman Sailer



The Time: 1913 **The Place: London**

Whodunit?

There's been a murder! Scotland Yard is buzzing with the news: "A dead body, lyin' there 'e was, kind o' messy an' all. Happened at the Manor — Crowley Manor..."

You are chief inspector A. Black, responsible for looking into the Crowley affair. Your case begins as a routine murder investigation. Quickly the plot twists and turns as you seek the assailants. Be prepared to face diabolical situations as you seek the answers to the Curse of Crowley Manor.

TRS-80 16K TAPE Model 1 & 3	**010-0108**	**$19.95**
TRS-80 32K DISK Model 1 & 3	**012-0108**	**$20.95**
***APPLE 48K DOS 3.3 Applesoft ROM**	**042-0108**	**$29.95**
ATARI 16K TAPE	**050-0108**	**$24.95**

***Apple version has full color hi-res pictures & sound effects.**

ESCAPE FROM TRAAM

By Jyym Pearson and Norman Sailer

The Time: 2525 **The Place: An Alien World**

Your small space cruiser has crashed on Traam, a bizzare, far-flung world. Miraculously, you lived, but only time will prove your survival a curse or a blessing. The environment is filled with incredible wonders and sights which no human eyes have ever beheld. But beware! This strangely beautiful world camouflages the danger that makes your escape from this planet imperative. Keep your wits about you — if your every move is not tempered with caution, Traam may serve as your headstone in the burial grounds of space.

TRS-80 16K TAPE Model 1 & 3	**010-0109**	**$19.95**
TRS-80 32K DISK Model 1 & 3	**012-0109**	**$20.95**
***APPLE 48K DOS 3.3 Applesoft ROM**	**042-0109**	**$29.95**
ATARI 16K TAPE	**050-0109**	**$24.95**

***Apple version has full color hi-res pictures & sound effects.**

LABYRINTH OF CRETE

By Cliff Johnson and Allen Pinero

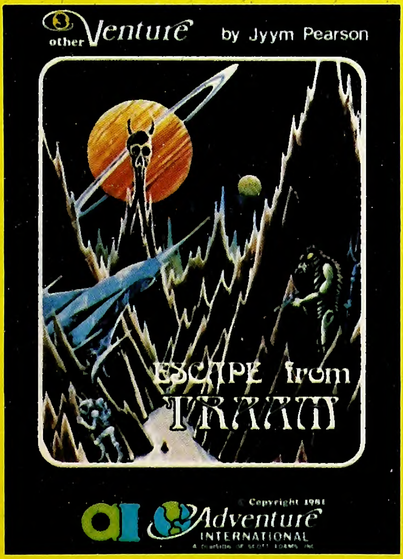
The Time: 1243 B.C. **The Place: Ancient Greece**

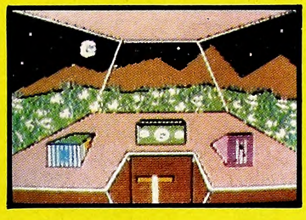
You control the actions of two of the mightiest men of ancient Greek mythology. Jason and the heroic Hercules are on a dangerous quest to recover the priceless Golden Fleece. You must guide Jason and Hercules safely past all manner of traps, tricks, and a parade of gruesome maze inhabitants.

Full-color Hi-Res Graphics have been included at the key points of your journey. The author has done an excellent job keeping the adventure true to the spirit and situations of ancient Greece.

A detailed instruction manual comes with your copy of Labyrinth of Crete.

APPLE 2 PLUS 48K DOS 3.3 ...... 042-0162 $29.95





To order, see your local dealer. If he does not have the program, then call 1-800-327-7172 (orders only please) or write for our free catalog.

Published by ADVENTURE INTERNATIONAL
a subsidiary of Scott Adams, Inc.
BOX 3435 • LONGWOOD, FL 32750 • (305) 862-6917
Prices Subject To Change Without Notice

Adventure INTERNATIONAL

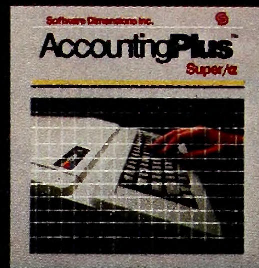
Dealer inquiries invited.

A fresh Apple deserves fresh software.

Software Dimensions first unveiled Accounting Plus in 1980. Since then, it has consistently been recognized by leading computer magazines as one of the world's top-selling business accounting packages.

Now, there's Accounting Plus Super/e, brand new and developed especially for the new Apple IIe. Accounting Plus Super/e is an impressive financial package designed to give you total control over your company's fiscal operations. The system consists of five interactive modules that efficiently and effectively handle fundamental accounting tasks for small businesses. It incorporates General Ledger, Accounts Payable, Accounts Receivable, Inventory, Payroll and more!

This user-friendly system is totally menu-driven. On-screen prompts guide you every step of the way. Flexible and versatile, Accounting Plus Super/e can be used with either two or three floppy disks or a hard disk. Complete cursor control simplifies editing, and a unique "screen dump" feature permits you to transfer any information appearing on the screen to the printer at any time.



The Accounting Plus Super/e General Ledger maintains a complete audit trail for up to 500 accounts. With automatic posting capabilities and an extensive reporting system, Accounting Plus Super/e simplifies traditionally complex, tedious tasks, including payroll and inventory.

Power. Efficiency. Ease of operation. These are the hallmarks of Accounting Plus Super/e, the most advanced accounting management system you can buy... right down to its new package design with built-in copy stand.

Accounting Plus Super/e puts accurate, reliable answers at your fingertips... So you can spend more time counting the fruits of your labors.

See your Apple dealer for complete details and an impressive demonstration.

The time is ripe. Introducing Accounting Plus™ Super/e.

Software Dimensions Inc.
6371 Auburn Boulevard
Citrus Heights, CA 95610
916/722/8000

Apple is a registered trademark of Apple Computer, Inc. Accounting Plus is a registered trademark of Software Dimensions, Inc. Dealer inquiries invited.

Softalk Presents The Bestsellers

It was oldies-but-goodies month in March as the Apple software market sagged significantly. Whether the cause of the decline is normal seasonal factors, a lack of bright new product, or massive indifference on the part of both the installed user base and the new owners, one fact stands out: Software sales are not matching the growth in sales exhibited by the Apple IIe.

The most reasonable postulation is that it's a combination of the three factors. March is never traditionally a strong month. On the other hand, it usually is slightly better than February, heralding a spring upturn. This year March stayed right in the trough with February. But harsh weather is certainly a contributor to sales downturns in some areas of the country.

The relatively stronger sale of older programs seems to point to the lack of good new software in the market. Software authors generally seem to be suffering from the same creative block afflicting television series writers. That makes the old-timers look pretty good. In television, *I Love Lucy* reruns outpull some network prime-time shows and almost all network daytime shows.

In the software market, such venerable product as *Typing Tutor* regained the Top Thirty with a vengeance, coming from nowhere to twelfth. *Flight Simulator*, the oldest of the entertainment packages still being sold in quantity, scored twenty-eighth. *Zork I*, the second oldest game, was nineteenth. *Castle Wolfenstein*, the third oldest entertainment package, was fourteenth. *Wizardry*, just weeks younger than *Wolfenstein*, was tied for fifth.

Long-running *Choplifter* was also tied for fifth, while aging *Knight of Diamonds*, which had fallen off the list last month, regained twenty-fourth in March.

Only one new entertainment product made the Top Thirty—*Dark Crystal* from Sierra On-Line made twenty-ninth.

But the conclusion that the market is headed toward business isn't necessarily true either. *PFS: File* and *PFS: Report*, two bellwether application packages, tailed off. *Quick File IIe*, which had jumped into the top ten last month, just as quickly plummeted several notches.

But just as in the entertainment segment, the tried-and-true old-timers seemed to prosper. *Apple Writer*, around in some incarnation practically since the beginning of time, led the pack. *VisiCalc* was second. *Home Accountant*, likewise a venerable performer, ranked third. *DB Master* reversed a downward trend and gained several notches. *Graphics Magician* made the Top Thirty after several months off the list.

So it appears that there are Apple owners out there with their buying pants on; they're just buying tried-and-true packages instead of the newer offerings. As mentioned last month, it's not altogether clear what the forty thousand new Apple owners each month are doing with their computers. There certainly aren't enough software sales to get much of a hint. One trend is becoming apparent: The new owners can't type. Not only did *Typing Tutor* score big, *MasterType* jumped to seventh, the highest rating ever for that program.

Apple III

This Month	Last Month	
1.	1.	**VisiCalc: Advanced Version,** Software Arts/Dan Bricklin and Robert Frankston, VisiCorp
2.	5.	**Word Juggler,** Tim Gill, Quark Engineering
3.	4.	**Quick File III,** Rupert Lissner, Apple Computer
4.	3.	**PFS: File,** John Page and D. D. Roberts, Software Publishing Corporation
	7.	**The Catalyst,** Tim Gill, Quark Engineering
	9.	**VersaForm,** Joseph Landau, Applied Software Technology
7.	6.	**VisiCalc III,** Software Arts/Dan Bricklin and Robert Frankston, VisiCorp
8.	7.	**PFS: Report,** John Page, Software Publishing Corporation
	10.	**General Ledger,** George Shackelford, State of the Art
10.	2.	**Apple Writer III,** Paul Lutus, Apple Computer

Arcade 10

This Month	Last Month	
1.	2.	**Miner 2049er,** Mike Livesay and Bill Hogue, Micro Fun
2.	1.	**Choplifter,** Dan Gorlin, Broderbund Software
3.	3.	**Frogger,** Olaf Lubeck, Sierra On-Line
4.	4.	**Aztec,** Paul Stephenson, DataMost
5.	6.	**The Arcade Machine,** Chris Jochumson and Doug Carlston, Broderbund Software
6.	5.	**Pinball Construction Set,** Bill Budge, BudgeCo
7.	7.	**Snack Attack,** Dan Illowsky, DataMost
8.	—	**Star Blazer,** Tony Suzuki, Broderbund Software
9.	—	**Canyon Climber,** Steve Bjork, Datasoft
	—	**A.E.,** Jun Wada, Broderbund Software

Which raises another tough question: Why are they buying all those word processors?

Not only is *Apple Writer* first, *Screen Writer II* is sixteenth, *Word Handler* is twenty-first, and *WordStar* is tied for twenty-sixth. That's a whole lot of word processing capability for a bunch of folks with ten thumbs.

Other word processing programs fared better in March than in February, when *Apple Writer* dominated. *Bank Street Writer* continues to gain strength. *Super-Text Pro*, *Magic Window II*, and *PIE Writer* held their own. *Sensible Speller* continues as the dominant dictionary program.

Among business programs, *Multiplan* continues to eat away at *VisiCalc*'s position as the dominant spreadsheet. The distance between the two is still great, but *Multiplan* is one of the few new applications programs to have caught the public's fancy. *List Handler*, from Silicon Valley Systems, and *VersaForm*, from Applied Software Technology, gained the lower rungs of the Business 10.

There was lots of shuffling in the Hobby 10 list, but no new programs. *Graphics Magician* took over the lead, with *DOS Boss* dropping to second and *Zoom Grafix* rising to third. *The Complete Graphics System* and *GraForth* rejoined the list. *Merlin* and *Lisa 2.5* ran a dead heat as leaders of the assembler products.

Home Accountant continues to smear the rest of the home applications programs. Howardsoft's *Tax Preparer* placed second, with Micro Lab's *Tax Manager* third, but most retailers reported disappointment

Word Processors 10

This Month	Last Month	
1.	1.	**Apple Writer II,** Paul Lutus, Apple Computer
2.	3.	**Screen Writer II,** David Kidwell, Sierra On-Line
3.	2.	**Word Handler,** Leonard Elekman, Silicon Valley Systems
4.	4.	**WordStar,** MicroPro
5.	5.	**Sensible Speller,** Sensible Software
6.	6.	**Bank Street Writer,** Gene Kusmiak and the Bank Street College of Education, Broderbund Software
7.	9.	**Apple Writer II Pre-Boot Disk,** Kevin Armstrong and Mark Borgerson, Videx
8.	8.	**Super-Text Pro,** Ed Zaron, Muse
9.	7.	**Magic Window II,** Bill Depew, Artsci
10.	9.	**PIE Writer,** Softwest, Hayden

Home Education 10

This Month	Last Month	
1.	1.	**MasterType,** Bruce Zweig, Lightning Software
2.	6.	**Typing Tutor,** Image Producers, Microsoft
3.	3.	**Facemaker,** DesignWare, Spinnaker Software
4.	3.	**Snooper Troops I,** Tom Snyder, Spinnaker Software
5.	2.	**Early Games for Young Children,** John Paulson, Counterpoint Software
	7.	**Ernie's Quiz,** Children's Television Workshop, Apple Computer
7.	—	**Mix & Match,** Children's Television Workshop, Apple Computer
8.	5.	**Story Machine,** DesignWare, Spinnaker Software
9.	—	**Apple Logo,** Logo Computer Systems, Apple Computer
10.	—	**Snooper Troops II,** Tom Snyder, Spinnaker Software

THE GALAXY AWAITS YOUR COMMAND.

When SSI introduced THE COSMIC BALANCE™, it was hailed as one of the finest tactical space games ever made: It not only gave you starship combat that was fun, fast and furious, it also let you design your ships. You became both starfleet commander and starship architect.

Now we are proud to present its sequel — **COSMIC BALANCE II: The Strategic Game™**. It allows all you aspiring Galactic Emperors out there to plot the growth of your space kingdom — from a few, paltry planets to the entire Galaxy! You discover and colonize planets, establish commerce nets, organize production of necessities, and send starships out on missions. There are five scenarios prepared for you, but you are free to create your own.

No matter how you play it, COSMIC BALANCE II™ is a game of interstellar conquest. And the only way you're going to enlarge your share of the cosmic pie is to win starship battles against your opponent (which can be a human or the computer).

When actual combat occurs, you can let the computer resolve it instantly. Or you can slug it out in all its blazing glory by using THE COSMIC BALANCE™. The battle outcome can then be incorporated into The Strategic Game.

Space may be what these games are all about, but there isn't enough of it here to adequately describe them. But why read when the Universe beckons? Plot a course to the nearest computer/game store and get these games today! You have a destiny to fulfill — a destiny that lies out there among the stars.

ON DISC FOR THE APPLE® AND ATARI®.

THE COSMIC BALANCE & THE COSMIC BALANCE II ($39.95 each) are on 48K diskette for the Apple II+ or Apple II with Applesoft ROM Card. Also on 48K disk for the Atari 400/800.

RapidFire
GAMES FROM SSI

If there are no convenient stores near you, VISA and MASTERCARD holders can order direct by calling **800-227-1617, x335 (toll free)**. In California, call 800-772-3545, x335.

To order by mail, send your check to: **Strategic Simulations Inc**, 465 Fairchild Drive, Suite 108, Mountain View, CA 94043. California residents, add 6½% sales tax.

WRITE FOR A FREE COLOR CATALOG OF ALL OUR GAMES.

with sales of tax packages this season. Among the communications packages, *Data Capture 4.0* surged to the lead, tying *Tax Manager* for third. *ASCII Express: The Professional* dropped to second among communications packages and fifth among all home products.

The Home Education 10 list was the only one that showed any growth. Of course, that was paced by the two typing programs. But half a dozen educational programs sat just below the Top Thirty. Spinnaker contributed three strong showings with *Facemaker* and the two *Snooper Troops* offerings. Children's Television Workshop fared well with *Ernie's Quiz* and *Mix & Match*. Just out of range were *Rocky's Boots* and *Gertrude's Secrets*, from the Learning Company.

The maze of arcade games has become sufficiently confusing that old-timers are even coming back there. *Star Blazer*, once given up for dead, bounced back, not so much on its outstanding strength as on the weakness of the competition. Likewise *Canyon Climber*, off the Arcade 10 list for some months, rejoined the list. One model of consistency is Bill Budge's *Pinball Construction Set*. It's not setting the world on fire, but neither is it suffering as much during the slump.

The biggest surprise in the Adventure 5 was the sudden demise of *The Mask of the Sun*. It went from first to off the list, actually ranking seventh. But publisher Ultrasoft had *Serpent Star* in the wings, and dealers are reporting fair sales of the new offering.

Adventure 5

This Month	Last Month	
1.	2.	**Zork I,** Infocom
2.	4.	**Deadline,** Infocom
3.	—	**Dark Crystal,** Roberta Williams, Sierra On-Line
4.	—	**Zork III,** Infocom
5.	3.	**Zork II,** Infocom
	5.	**Sherwood Forest,** Dav Holle and Dale Johnson, Phoenix Software

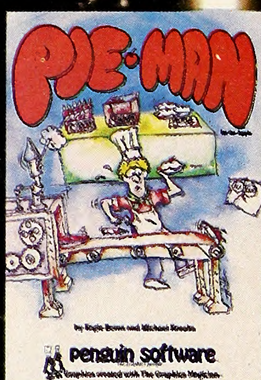
Fantasy 5

This Month	Last Month	
1.	1.	**Wizardry,** Andrew Greenberg and Robert Woodhead, Sir-tech
2.	2.	**Ultima II,** Lord British, Sierra On-Line
3.	3.	**Knight of Diamonds,** Andrew Greenberg and Robert Woodhead, Sir-tech
4.	—	**Temple of Apshai,** Epyx/Automated Simulations
5.	—	**Ultima,** Lord British, California Pacific

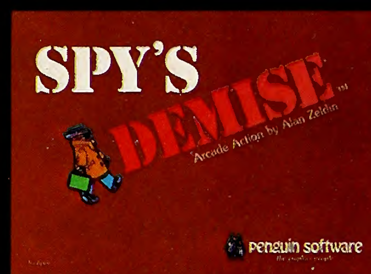
Strategy 5

This Month	Last Month	
1.	1.	**Castle Wolfenstein,** Silas Warner, Muse
2.	2.	**Flight Simulator,** Bruce Artwick, SubLogic
3.	5.	**Sargon II,** Dan and Kathe Spracklen, Hayden
4.	4.	**Galactic Adventures,** Tom Reamy, Strategic Simulations
5.	—	**Rendezvous,** Wes Huntress, Edu-Ware Services

LOOK WHAT YOU CAN GET FOR $19.95!



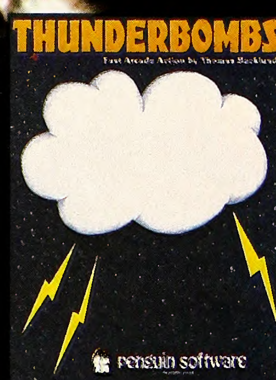
Chaos in the bakery as you contend with a cantankerous conveyor belt in an effort to make pies.



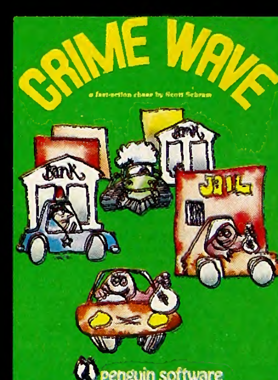
See if you can sneak past the security guards in this best-seller.

"Best graphics ever in a hi-res adventure ..."
—Softalk

NEW RELEASES



Run a gauntlet of deadly alien sharpshooters.



The criminal element has run amok. Can you round them up?

We believe games should be fun and that the price of games shouldn't dampen that fun. The growth of the market over the past couple of years leads us to believe that $19.95 may work now as a reasonable game price, so we're trying it for the next six months, and if we're right, longer. This policy does not just apply to new games, but to ALL our games, including our past and current best-sellers! Our bet is that we'll sell more and that the increased sales will offset the decreased income per product. If so, more people get to play our games, and we still make enough to keep developing newer and better software.

As our customers know, at Penguin Software we take a great deal of care and pride in our products. This change in our pricing in no way affects our standards of quality. We pioneered the removal of copy-protection from applications software last year in an effort to give you a better product. This year we are trying again to lead the way in putting the customer first.

penguin software the graphics people

(312) 232-1984 830 4th Avenue, Geneva, Il 60134

Dealer Hotline: (800) 323-0116, retailers only, please.

Available at your local computer store. Dealer and distributor inquiries welcome. Visa MasterCard accepted.

Sun's dip opened the way for *Zork I* to regain first. *Deadline* was second, and *Zork III* narrowly edged out *Zork II* for fourth. *Sherwood Forest*, from Phoenix, and *Dark Crystal* rounded out the list.

Business 10

This Month	Last Month	
1.	1.	**VisiCalc**, Software Arts/Dan Bricklin and Robert Frankston, VisiCorp
2.	2.	**PFS: File**, John Page and D. D. Roberts, Software Publishing Corporation
3.	4.	**Multiplan**, Microsoft
4.	3.	**Quick File IIe**, Rupert Lissner, Apple Computer
5.	8.	**DB Master**, DB Master Associates, Stoneware
6.	5.	**PFS: Report**, John Page, Software Publishing Corporation
7.	6.	**BPI General Ledger**, John Moss and Ken Debower, Apple Computer
8.	10.	**VisiFile**, Creative Computer Applications/Colin Jameson and Ben Herman, VisiCorp
	—	**List Handler**, Silicon Valley Systems
10.	—	**VersaForm**, Joseph Landau, Applied Software Technology

Hobby 10

This Month	Last Month	
1.	3.	**Graphics Magician**, Chris Jochumson, David Lubar, and Mark Pelczarski, Penguin Software
2.	1.	**DOS Boss**, Bert Kersey and Jack Cassidy, Beagle Bros
3.	6.	**Zoom Grafix**, Dav Holle, Phoenix Software
4.	—	**The Complete Graphics System**, Mark Pelczarski, Penguin Software
	5.	**Apple Pascal**, Apple Computer
6.	6.	**Utility City**, Bert Kersey, Beagle Bros
7.	6.	**Bag of Tricks**, Don Worth and Pieter Lechner, Quality Software
8.	3.	**Pronto DOS**, Tom Weishaar, Beagle Bros
9.	—	**Alpha Plot**, Bert Kersey and Jack Cassidy, Beagle Bros
	—	**GraForth**, Paul Lutus, Insoft

Home 10

This Month	Last Month	
1.	1.	**Home Accountant**, Bob Schoenburg, Larry Grodin, and Steve Pollack, Continental Software
2.	6.	**Tax Preparer**, James Howard, HowardSoft
3.	3.	**Tax Manager**, TASO, Micro Lab
	—	**Data Capture 4.0**, George McClellan and David Hughes, Southeastern Software
5.	2.	**ASCII Express: The Professional**, Bill Blue and Mark Robbins, Southwestern Data Systems
6.	—	**Personal Finance Manager**, Jeffrey Gold, Apple Computer
7.	7.	**Hayes Terminal Program**, Hayes Microcomputer Products
	9.	**Tax Advantage**, Continental Software
9.	7.	**Transend 1**, Tim Dygert and Bob Kniskern, SSM
	10.	**VisiTerm**, Tom Keith, VisiCorp

Softalk Presents The Bestsellers

The fantasy market is truly depressed. After *Wizardry*, *Knight of Diamonds*, and *Ultima II*, there literally were no significant sales. *Temple of Apshai* and *Ultima*, apparently living more off their laurels than from any burning desire on the part of Apple owners to buy the games, gained the bottom two rungs.

Castle Wolfenstein maintains its strangle hold on the Strategy 5 list, with *Flight Simulator* maintaining second. A surprisingly strong showing was made by *Sargon II*, which trounced *Chess 7.0* for chess honors in March. Strategic Simulations's *Galactic Adventures* continued strong, and Edu-Ware's *Rendezvous* made another of its periodic visits to the Strategy 5 list.

The sales blight fell particularly heavily on the Apple III market. Outside of *VisiCalc: Advanced Version*, nothing much sold. *Apple Writer III* got bit the worst, tailing off to practically nothing.

March marked the first month in which fewer software titles were reported by dealers than the month before. Whether that's a temporary phenomenon or a reflection of writer's block will be determined in the next few months.

In the meantime, we can be sure of only one thing: The Apple IIe is the most popular machine for teaching typing in the United States.

Apple-franchised retail stores representing approximately 5.3 percent of all sales of Apple and Apple-related products volunteered to participate in the poll.

Respondents were contacted early in April to ascertain their sales for the month of March.

The only criterion for inclusion on the list was the number of units sold—such other criteria as quality of product, profitability to the computer store, and personal preference of the individual respondents were not considered.

Respondents in April represented every geographical area of the continental United States.

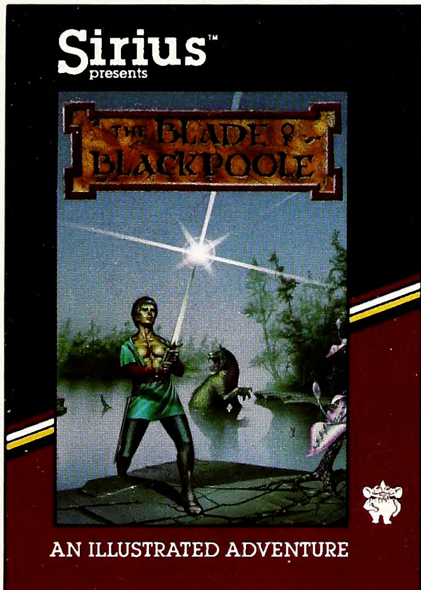
Results of the responses were tabulated using a formula that resulted in the index number to the left of the program name in the Top Thirty listing. The index number is an arbitrary measure of relative strength of the programs listed. Index numbers are correlative only to the month in which they are printed; readers cannot assume that an index rating of 50 in one month represents equivalent sales to an index number of 50 in another month.

Probability of statistical error is plus or minus 4.03 percent, which translates roughly into the theoretical possibility of a change of 4.83 points, plus or minus, in any index number.

The Top Thirty

This Month	Last Month	Index	
1.	1.	162.08	**Apple Writer II,** Paul Lutus, Apple Computer
2.	2.	114.06	**VisiCalc,** Software Arts/Dan Bricklin and Robert Frankston, VisiCorp
3.	4.	97.38	**Home Accountant,** Bob Schoenburg, Larry Grodin, and Steve Pollack, Continental Software
4.	6.	76.04	**Miner 2049er,** Mike Livesay and Bill Hogue, Micro Fun
5.	5.	71.37	**Choplifter,** Dan Gorlin, Broderbund Software
	8.	71.37	**Wizardry,** Andrew Greenberg and Robert Woodhead, Sir-tech
7.	17.	51.36	**MasterType,** Bruce Zweig, Lightning Software
8.	3.	49.36	**PFS: File,** John Page and D. D. Roberts, Software Publishing Corporation
9.	10.	40.02	**Multiplan,** Microsoft
10.	12.	37.35	**Ultima II,** Lord British, Sierra On-Line
11.	7.	36.68	**Frogger,** Olaf Lubeck, Sierra On-Line
12.	—	34.68	**Typing Tutor,** Image Producers, Microsoft
13.	14.	32.68	**Aztec,** Paul Stephenson, DataMost
14.	16.	32.02	**Castle Wolfenstein,** Silas Warner, Muse
15.	—	28.68	**The Arcade Machine,** Chris Jochumson and Doug Carlston, Broderbund Software
16.	18.	28.01	**Pinball Construction Set,** Bill Budge, BudgeCo
	9.	28.01	**Quick File IIe,** Rupert Lissner, Apple Computer
	14.	28.01	**Screen Writer II,** David Kidwell, Sierra On-Line
19.	21.	26.68	**Zork I,** Infocom
20.	26.	24.68	**DB Master,** DB Associates, Stoneware
21.	13.	24.01	**Word Handler,** Leonard Elekman, Silicon Valley Systems
22.	11.	21.34	**PFS: Report,** John Page, Software Publishing Corporation
23.	30.	20.68	**Deadline,** Infocom
24.	—	20.01	**Graphics Magician,** Chris Jochumson, David Lubar, and Mark Pelczarski, Penguin Software
	—	20.01	**Knight of Diamonds,** Andrew Greenberg and Robert Woodhead, Sir-tech
26.	24.	18.68	**WordStar,** MicroPro
	—	18.68	**Tax Preparer,** James Howard, Howardsoft
28.	27.	18.01	**Flight Simulator,** Bruce Artwick, SubLogic
29.	—	16.67	**Dark Crystal,** Roberta Williams, Sierra On-Line
30.	—	16.01	**Facemaker,** DesignWare, Spinnaker Software

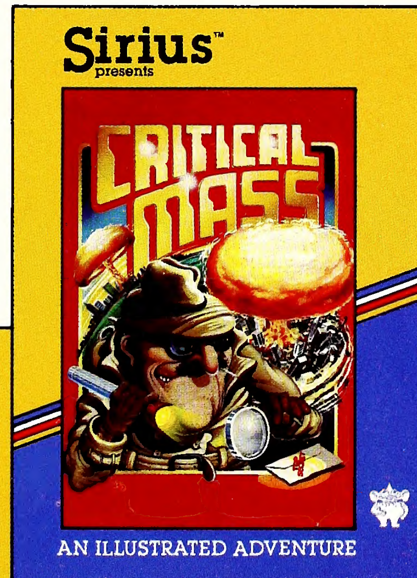
For Heroes Only!



Blade of Blackpoole

Step back in time and join the search for the magical sword of Myraglym. Travel cautiously on your journey for you will encounter dangerous serpents, spine-chilling evils and carnivorous plants that crave human flesh!

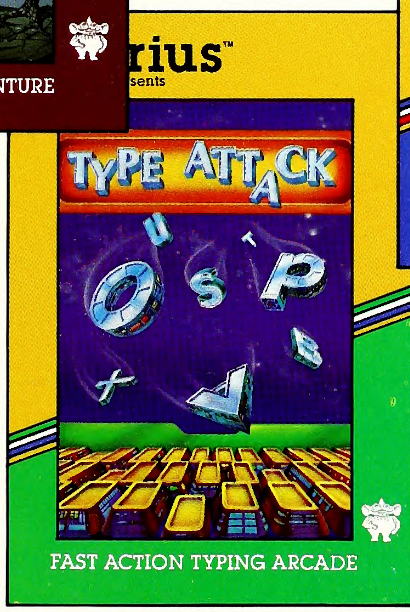
Avail. on disk for the Apple II, II+ or IIe and Atari 800 or 1200 and Commodore 64.



Critical Mass

On Jan. 1st at 10:00 am, the U.N. received this message: "Good Morning, in exactly 9 days, the world's 5 largest cities will be destroyed by thermal nuclear weapons." At 10:03 am, you received this assignment: STOP . . . THIS . . . LUNATIC!

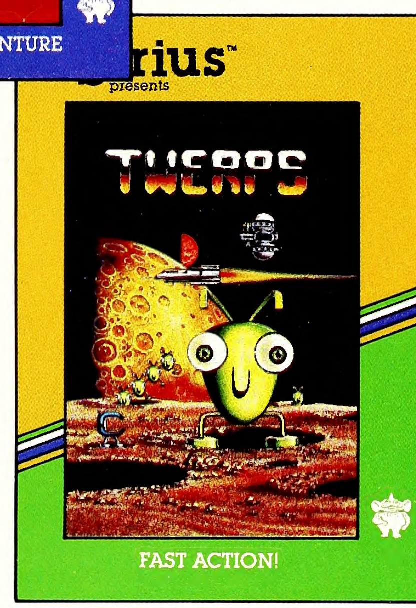
Avail. on disk for the Apple II, II+ or IIe and Atari 800 or 1200 and Commodore 64.



Type Attack

The planet Lexicon is under attack! Letters of the alphabet are falling from the sky. To repel them, you must be able to type the letters faster than they can fall. Be quick! An entire civilization is depending on your skill.

Avail. on disk for the Apple II, II+ or IIe and Atari 800 or 1200, IBM-PC and Commodore 64 and on cartridge for the VIC-20.



Twerps

The boldest space rescue ever! Defenseless Twerps are stranded on an asteroid. You, Captain Twerp, are to board a Twerp-craft, blast through the Orbiters, land safely and rescue your comrades. Beware of the Glingas and Twerp-eating Gleepnites!

Avail. on disk for the Apple II, II+ or IIe and Atari 800 or 1200.

Pure Video Excitement!

For Your Atari 800 or 1200, Apple II, II+ or IIe, Commodore 64, VIC-20 and IBM-PC