SOFTALK

VOLUME 2 APRIL 1982 $2.50

Apple in Outer Space

Home Finance Roundup

Exec HowardSoft

Word Processor: Letter Perfect

Debut: DOStalk

Many people think just because a computer does difficult things, it must be difficult to operate. Not so. At least not with PFS™ and PFS: REPORT.

PFS and PFS: REPORT let you concentrate on doing your work, not on working your computer. There's no complicated programming or computer language to learn. Using the PFS Series is so straightforward, you can learn it in as little as 20 minutes, and that's something both experts and beginners can appreciate.

Design your own management system, simply.

PFS lets you arrange your information in "forms" you design yourself. Once you've created your form—with spaces for all the necessary data—you simply fill in the blanks.

If you don't like the original form, PFS lets you change *just* the form, without having to totally re-enter the information on it.

To use your forms, just call up the information you want in a variety of ways such as "less than," "more than," "equal to," "between" or "not." You can even find specific characters, words, names or numbers. And PFS can also sort labels, change its 40-column format to accommodate larger printouts, and tell you how much room you have left on your disk.

Get more out of what you put in.

PFS: REPORT sorts, calculates, totals, formats and prints *presentation-quality* reports with up to 9 columns, in alpha or numeric order, with automatic entering and justification. With your information in this format, you can analyze it quickly to help spot trends, plan and make decisions.

A whole family of software for Apple® II and III.

PFS consistently ranks among the 5 top-selling business software packages for Apples. Because you can use it for almost anything—from chemical analyses to keeping track of your wine collection, from balancing your checkbook to computing your payroll. And there are more programs on the way.

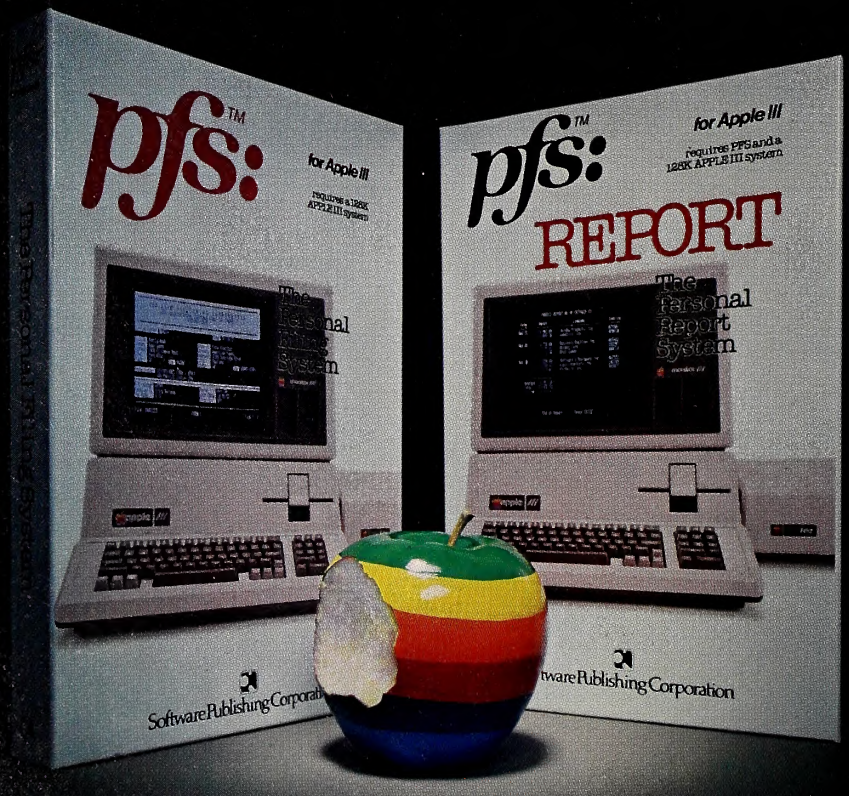
For a free, hands-on demonstration of PFS and PFS: REPORT, stop by your local computer store. If they don't have it yet, tell them to call Software Publishing Corporation at (415) 962-8910. Or write 1901 Landings Dr., Mountain View, CA 94043.

You'll be amazed at how much you can bite off.

Hardworking software for Apples that isn't a lot of work.

pfs:™ Software Series

®Apple is a registered trademark of Apple Computer, Inc. ™PFS is a trademark of Software Publishing Corporation.



SOFTALK CONTENTS

APRIL 1982

Exec Howardsoft: A Taxing Profession? Yes and No. . . .

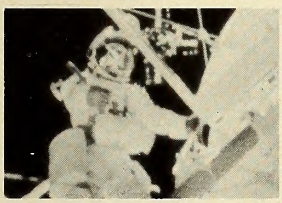
Taxes isn't a bad word to a man who built a business upon them

DAVID HUNTER 50

Word Processors: Civic Leader Claims His WP Is Letter Perfect

Porter Loring took over the family business from his father; now this San Antonio community leader is computerized

JONATHAN MILLER 72



Sunflowers in Space

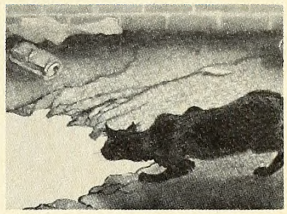
When Spacelab flies into orbit, an Apple II Plus, sans case, will go with it. Its duty? To look after the sunflowers. . . .

MELISSA MILICH 80

The Changing Character(s) of the Apple III

There are four different character sets built into the Apple III; but you can ignore them all, design your own, and program that flexible keyboard

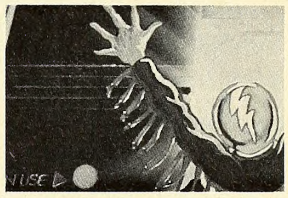
JOHN P. JEPPSON 134



The Speed Sector

DOS was built to boot fast—and that's about all it does fast. Now there's a way to speed it up on all tasks

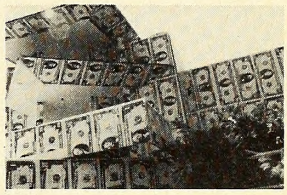
DON WORTH and PIETER LECHNER 176



Home Finance Roundup

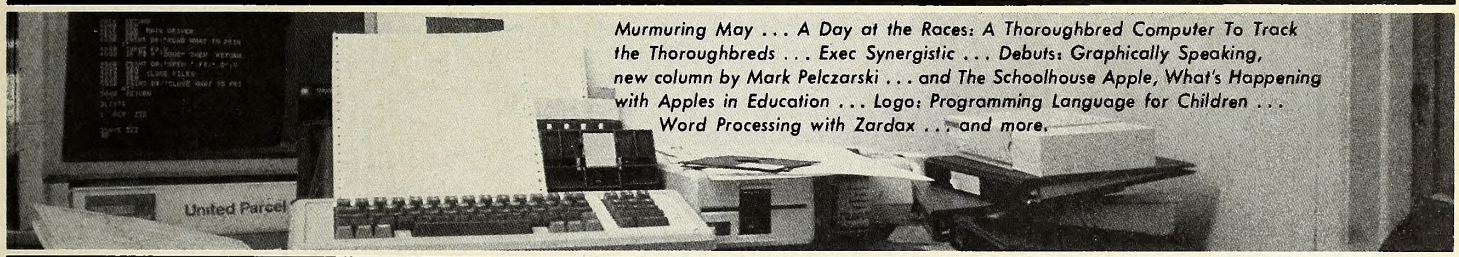
Updating our report from last year, we look at five new or heavily revised home accounting systems, one of which may be ideal for you

CRAIG STINSON 199



FEATURES · DEPARTMENTS

IN NEXT MONTH'S SOFTALK



Murmuring May . . . A Day at the Races: A Thoroughbred Computer To Track the Thoroughbreds . . . Exec Synergistic . . . Debuts: Graphically Speaking, new column by Mark Pelczarski . . . and The Schoolhouse Apple, What's Happening with Apples in Education . . . Logo: Programming Language for Children . . . Word Processing with Zardax . . . and more.

INDEX OF ADVERTISERS

CONTEST: MAN ON THE MOON

You don't have to complete the crossword puzzle on page 4 to win this contest; the required answer is the phrase at the end and its identification. But you'll need to solve some of the puzzle to know what goes in the phrase.

The shaded letters in the puzzle anagram to produce a famous quote. The quote relates to another extremely significant achievement that has affected the way of the world. To win the contest, you must solve this quote correctly, then identify who said it, the circumstances under which it was said, and the significance of those circumstances.

Except for the final sentence, every answer in Man on the Moon has to do with outer space or the U.S. space program.

Man on the Moon is our second puns 'n' anagrams puzzle; the answers are easy but the questions aren't. The clues are maliciously designed to be tricky and misleading, even though they often contain more than one definition and they always contain at least two hints each to the answer.

The hints may be of any sort. For example, the actual answer may be contained in the clue, except that the letters are mixed up, and you'll need to anagram them: "cam*p case*" hides an anagram of *space*. Or the answer may be hidden: "Jim's *pa ce*ded his land."

The clue may be a pun: "The vet *spays* dogs throughout the universe." or a charade: "room for a well-paid veteran weight lifter"—room = space, well-paid veteran = pro, weight (lifter is just to confuse you) = gram; put them together to get *space program.*

Each clue contains a real definition of the answer, which may be either definitive, descriptive, or exemplar but it may be of any meaning of the word, not necessarily of the meaning pertaining to space. A clue for the word *space* might contain a definition such as room, interstice, area, emptiness; *Venus* might be defined as planet, goddess, through the veins (venous), like an Italian city (Venice). The last two required translating the pun as well.

Throughout the puzzle, you'll find shaded squares with small letters in the lower left corners. The answer letters that go in these squares are the letters that make up the phrase beneath the puzzle. The small letters indicate which word of the phrase the answer letter belongs in. The letters for each word must be anagrammed to find the phrase. The blanks have purposely not been punctuated; you need not punctuate the phrase, but do not assume it shouldn't have punctuation.

Again, winning the contest does not require that you finish the puzzle, only that you correctly solve for the phrase and identify it completely. However, if you do complete the puzzle, you're eligible for additional prizes.

The winning entry will choose $100 worth of *Softalk* advertisers' goods at the local computer store. In case of a tie, the random number generator will choose the winner.

Five correct entries that also contain correct completed puzzles will receive copies of *Crossword Magic,* the program from L&S Software with which the puzzle was generated. If there are more than five correct puzzles submitted, the five winners will be chosen by the random generator.

Fill out the entry blank and send it to Softalk Moon, 11021 Magnolia Boulevard, North Hollywood, CA 91601, by May 15, 1982.

Send this form or a facsimile to Softalk Moon, 11021 Magnolia Boulevard, North Hollywood, CA 91601, by May 15, 1982. To be eligible to win copies of *Crossword Magic,* include the completed puzzle.

The quote is_____

The speaker was: _____

The situation was: _____

The significance was: _____

☐ Count me in for a *Crossword Magic*; my completed puzzle is enclosed.

Name: _____

Address: _____

City/State/Zip: _____

If I win I'd like: _____

My dealer is: _____

Autograph: _____

Chairman	John Haller
Publisher	Al Tommervik
Editor	Margot Comstock Tommervik
Art Director	Kurt A. Wahlner
Associate Editors	Craig Stinson Jean Varven
Special Assignments	Melissa Milich David Hunter
Assistant Editor	Andrew Christie
Proofreader	Harry McNeil
Editorial Assistant	Karen Kleiner
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	Bert Kersey
Word Processors	Jonathan Miller
Basic Solution	Wm. V. R. Smith
Guest Reviewers	Peter Stone Richard Kaapke Allen Munro
Special Projects	Greg Voss
Art Assistant	Rubuel Kissler
Associate Publisher	Mary Sue Rennells
Promotion	Jo Hilliard
Accounting	Evelyn Burke
Office Manager	Nancy Oppenheim
Advertising Coordinator	David Hunter
West Coast Sales	Jo Hilliard Softalk 11021 Magnolia Boulevard North Hollywood, CA 91601 (213) 980-5074
East Coast Sales	Paul McGinnis Company 60 East 42nd Street New York, NY 10017 (212) 490-1021
Midwest and Rocky Mountain Sales	Ted Rickard John Sienkiewicz Market/Media Associates 435 Locust Road Wilmette, IL 60091 (312) 251-2541
Circulation Supervisors	Ron Rennells Robert Mann Hal Schick Dan Yoder
Circulation Assistants	Donna Siebert Dierdre Booth Mary Milam
Systems	Wm. V. R. Smith Ken Williams Sam Pawley

Cover photo of astronaut Owen Garriott courtesy of the National Aeronautics and Space Administration. Design by Kurt Wahlner.

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

Softalk. Volume 2, Number 8. Copyright © 1982 by Softalk Publishing Inc. All rights reserved. ISSN:0274-9629. *Softalk* is published monthly by Softalk Publishing Inc., 11021 Magnolia Boulevard, North Hollywood, CA 91601. Telephone: Business office, (213) 980-5074; editorial office, (213) 763-0975; circulation, (213) 980-5099. Second-class postage aid at North Hollywood, CA, and additional mailing offices.

Postmaster: Send address changes to Softalk, 11021 Magnolia Boulevard, North Hollywood, CA 91601.

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, 11021 Magnolia Boulevard, North Hollywood, CA 91601. *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 the trial subscription, 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, $8: multiple subscriptions for school and libraries, $5 each.

Back Issues: $2 through February 1981; $2.50 from April 1981 through July 1981; $3.50 thereafter. November and December 1980 and January, February, March, September, October, and November 1981 are sold out. October 1980 is 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, Ron Rennells or Bob Mann can help out. Call (213) 980-5099.

Moving? Send new address and old to Softalk Circulation, 11021 Magnolia Boulevard, North Hollywood, CA 91601; telephone, (213) 980-5099.

CAVALIER COMPUTER GAMES

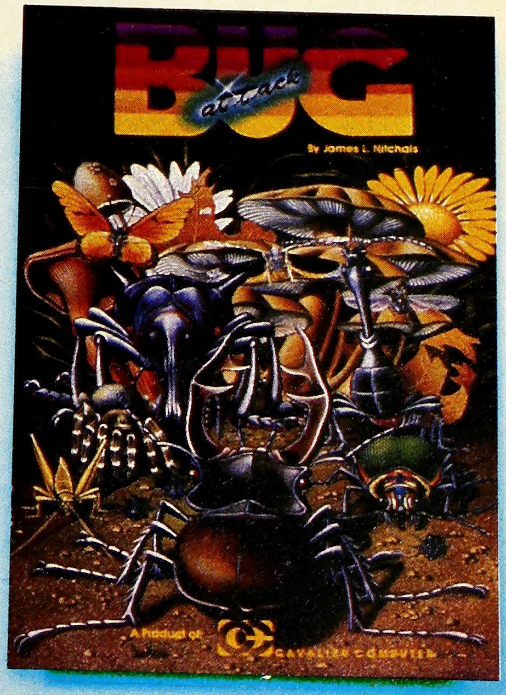
THEY'RE BETTER THAN EVER

STAR THIEF

Star Thief is an exciting one or two player game where the defender works alone or with an ally to protect the precious fuel pods from alien marauders. Star Thief has a unique training mission mode. Star Thief's two player mode will unite both friend and foe.

ASTEROID FIELD

The Asteroid Field is the most sought after and similar rendition of the original arcade game. So much so that it's now Audiovisually Licensed By Atari. Since everyone knows how to play _____ there's no need to mention how the Asteroid Field works, except that the player can use both keyboard and paddles.



BUG ATTACK

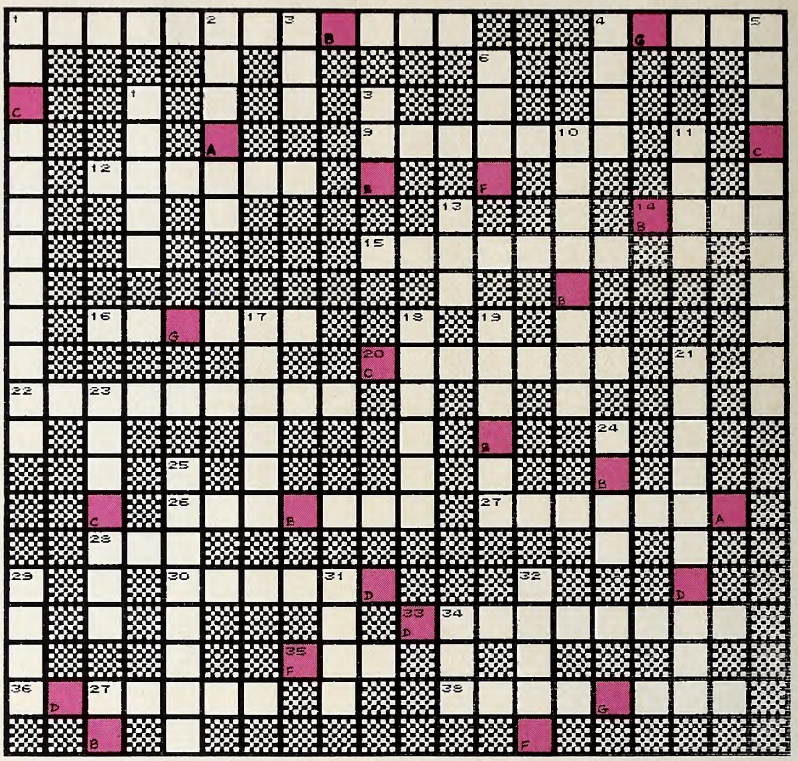
The bugs are on the loose, and they're hungry. Defend against ants' antics, malicious millipedes, and the menacing medfly invasion. Bug Attack will keep the whole family captive with colorful creatures, superb sound effects, and stimulating music. Can You Survive?

CAVALIER COMPUTER
CORPORATION
P. O. BOX 2032 — DEL MAR, CA 92014 — (714) 755-8143

THE FUTURE IS HERE AT CAVALIER COMPUTER

Exclusively distributed by
SOFSEL
8295 Sa La Cienega Blvd. Inglewood CA 90301
(800) 421-5770 In Calif (800) 262-4242

MAN ON THE MOON PUZZLE



A B C D E F G

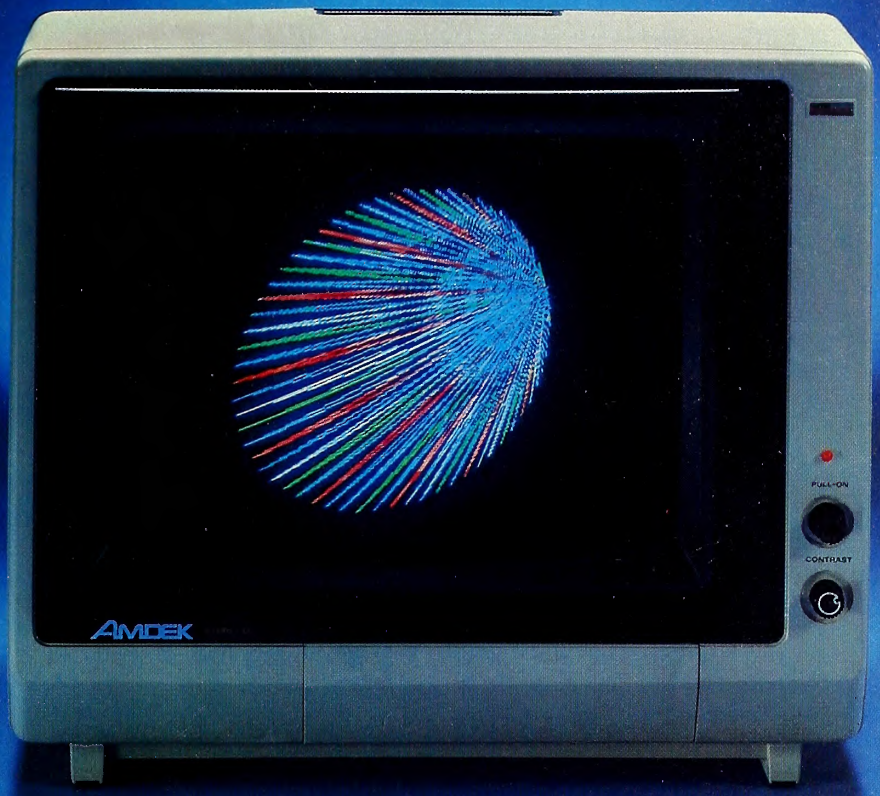
Across

1. Baby's red-letter day was progress for man.
4. Red project comes out with beans and a little Duz.
9. Now, Bette Davis on a boat around Mars.
12. Bejeweled in Iceland; gave birth to twins.
14. Grand cancellation under Nixon; next chance to see the sights, 170 years.
15. Times Square to Grand Central in a loom device saves NASA money.
16. Piecemeal in Australia is first extraterrestrial office.
20. Ancient ram in rerun, cut, sails are high and thirsty.
22. "Gran's mort," he said sadly, and went for his first walk.
26. New frontier person, i.e., no snakes.
27. Smog, sir, was one of our early astronauts.
28. Place behind it all much touted in the end by Sousa.
30. When you can't recoup your losses, that is, get them back, look for a barrel maker.
33. The rabid rely on getting the best first thing.
35. Universal question means stop to earthbound, go to pioneer.
36. He popped up first; she pardoned him for lying in the field.
38. Semi watchman liked to go before the rest.

Down

1. To Nureyev's finest moment I go; Anne pleats her skirt.
2. "I'm second to none," said the bald ringmaster. But he wasn't.
3. Blanc, Torme, and Brooks on expedition vehicle. No, that's backward.
4. Raster Blaster's mixed-up beginning takes the cosmic spotlight.
5. Ought to lose weight in seriousness.
6. A bane for Alan L. Astronaut.
7. A Ford of faith, legislatively speaking.
8. Riding on running board shortly in reverse on Fifth with Peron.
10. Rex Harrison and Peter Lorre confused at beginnings and end thought they were Balboa and Pizarro.
11. Remember when Woody Woodpecker circled in the midst of the end of man? That's Looney Tunes.
13. "You're too strong and too hot, and that's uncalled for," said the scion.
17. Wear a polo coat when you ride a flying chariot.
18. "Why didn't I get the role? I'm good alone," said a troubled Garner.
19. It's ludicrous to kill the role of Henry from King Henry VI, but they did it in Norwegian.
21. Envelope for your floppy? Franklin was it for electricity.
23. Lincoln's erstwhile partner ankled in flight from a rising thermometer.
24. As an astronaut, I'm not, but many are and many more, especially in the establishment.
25. Let Pascal be turned into the language of extraterrestrial science facilities.
29. Plan it with read and write memories, short ones, snickering about the Milky Way.
31. Little Sir repeated ad hoc, eminently at first.
32. Give the sunflower a place on the plat no smaller than pothus.
34. Ride over land and stream and mountains and desert, not camels, shortened by confused avatar.
37. Spaced astronaut initially walked out; Leonov went first. ◨

The High-Resolution Color Monitor for Apple II



Amdek's new Color-II monitor with DVM interface board

No video monitor has ever combined Apple II compatibility with exceptional performance like the new 13" Color-II monitor from Amdek.

Color-II features our optional new DVM peripheral board for easy interfacing with your Apple II. And look at the other top-of-the-line features Color-II offers:

- RGB, TTL input for high resolution graphics.
- 80 x 24 character display capability (with expansion card by Vydec).
- 560 (H) x 260 (V) resolution.
- Molded-in carrying handle for easier portability.
- Front-mounted controls for faster fine-tuning.

So if you want to get the most from your Apple II system, get the "most" monitor. Ask your dealer about Amdek's new Color-II monitor with the optional DVM interface board.

What is DVM? The DVM, or "Digital Video Multiplexor," is a low cost interface that allows the Apple II computer to be used with an RGB monitor, such as our Color-II.

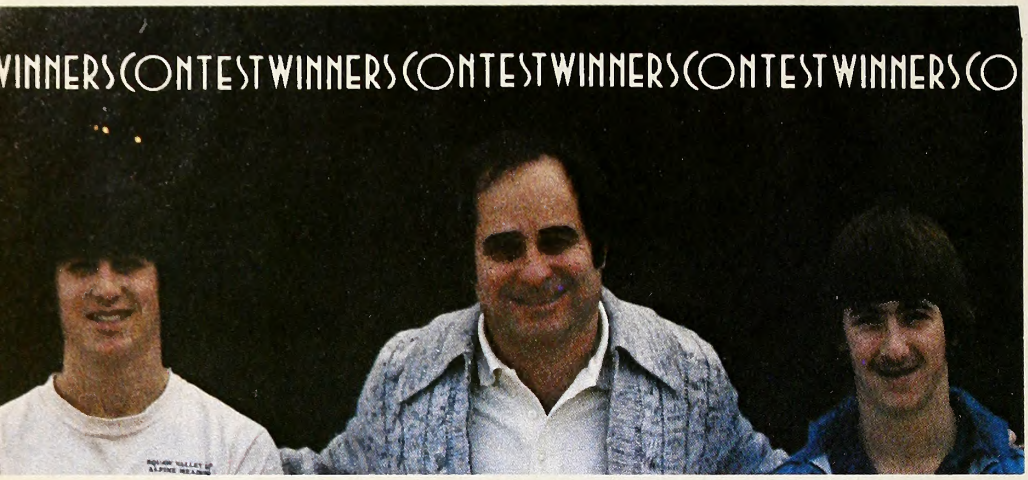
Amdek's DVM is software programmable to allow transparent operation, and is parallel with existing Apple text and graphic modes. Three of its channels are used to multiplex the existing Apple text, low resolution and high resolution graphics. The 4th channel allows the use of an 80 character line video board.

Our DVM is also color channel software programmable, enabling you to turn the three color channels on or off by software control. For example, the red and blue channel can be turned off when the 80 character channel is turned on, resulting in a green phosphor video presentation.

For convenient operation, the DVM board may occupy any slot in the Apple II. The DVM also features low-power consumption and low-power schottky logic.

AMDEK

Amdek Corporation, 2420 E. Oakton Street, Suite "E" Arlington Heights, Illinois 60005 (312) 364-1180 TLX: 25-4786



Oracle Part 5 winner Paul Shanberg, center, flonked by his two sons Dovid ond Jeff in frant of Computerlond af Wolnut Creek, Califarnio, where Shanberg Sr. claimed his prize, the *DOS Taal Kit*.

Oracle '82. Last year, Barbara Wright of Oceanport, New Jersey, ran off with January's football question, giving the combined score difference between winner and loser in the Rose Bowl and Super Bowl games. She claimed it was purely a guess. But we're beginning to believe she and her family have a direct line to Pete Rozelle. This year, the task was to estimate the Nielsen rating the Super Bowl would receive. No, none of the Wrights won this time—but daughter Martha tied for second place with an entry only .1 off the exact Nielsen mark.

Since there are at least two men in the Wright household, we'll brook no more nonsense about football being a "men's" sport. Especially in view of the entry that did take the prize for 1982: Elizabeth M Lewis, Richmond, Virginia, predicted the rating to be 49.1, which is on the nose according to CBS spokesman Hal Biard.

Lewis wins $100 toward the purchase of *DB Master*, the prize she chose, at her local dealer, Chesapeake Technology. She also leads for the grand prize Disk II in the overall Oracle '82 contest with the highest possible score, which is at present 0.

Martha Wright, who predicted 49 even, and Peter Rosenfeld (Flushing, NY) and Clark Dresser (Northridge, CA), with estimates of 49.2, stand in combined second, third, and fourth places with −.1 each. Rounding out the leading ten after part one are Paul Shanberg (Moraga, CA), fifth with −.3; Charles S. Lewis (Richmond, VA—yes, he's the winner's husband; an obvious conspiracy), sixth with −.4; Charles Lewis and Elizabeth Lewis tied on additional entries for seventh with −.5; Paul Shanberg, another entry, ninth with −.5; and Jean Armour (Liverpool, NY) with −.9.

It's not only the Wrights and the Lewises who are showing clear psychic abilities; Paul Shanberg, who holds positions five and nine, won parts two and five of Oracle '81.

No one is out of the running by any means. Few people were more than fifteen points off the mark on this round, and that's close. To figure your score to this point, simply compare your score with 49.1, subtract the smaller from the larger, and put a minus sign in front of the result.

Jumble. Philip Suh's prize-winning contest drew one of the largest volumes of entries of any *Softalk* contest, and this despite a typo in one of the names to be anagrammed. A saving grace was the fascination of *Softalk*'s art department for that particular name, which led them to feature it in the illustration—sans typo. Numerous entrants commented on the faux pas, from the gentleman who claimed first prize because he was so sure no one else would catch on to the reader who's delightedly accused us of being extra sneaky just to make it really hard. Then there was the note from Pleasant Hill High School Computer Club, which describes the situation pretty well:

"We knew it had to be an error. We came up with 'pigbroth castle' and 'graphics bottle' but somehow they just didn't seem right."

Despite the jumble, 555 people got the right answers to the Jumble. The random generator—which is really triply random since it generates a random number of random numbers up to the total number of winning entries and then chooses one of them at random—puffed and rumbled its way to choosing Mark Brothers of Elizabeth City, North Carolina, out of the jumble of entries.

Brothers will invest his $100 prize in Visi products for his work in real estate. He'll pick them up from his dealer, Randy Williams, at the Home Computer Center in Virginia Beach, Virginia.

Incidentally, the names of the user group members anagrammed to (1) *DB Master*, (2) *Super Invader*, (3) *The Controller*, (4) *Dragon Fire*, (5) *Graphics Tablet*, (6) *Magic Window*, (7) *DOS Tool Kit*, (8) *Micromodem*, (9) *Silentype*, and (10) *Double Vision*.

And the answer to what the strange user group discussed at its meeting was "Solving the Softalk Contest." ◨

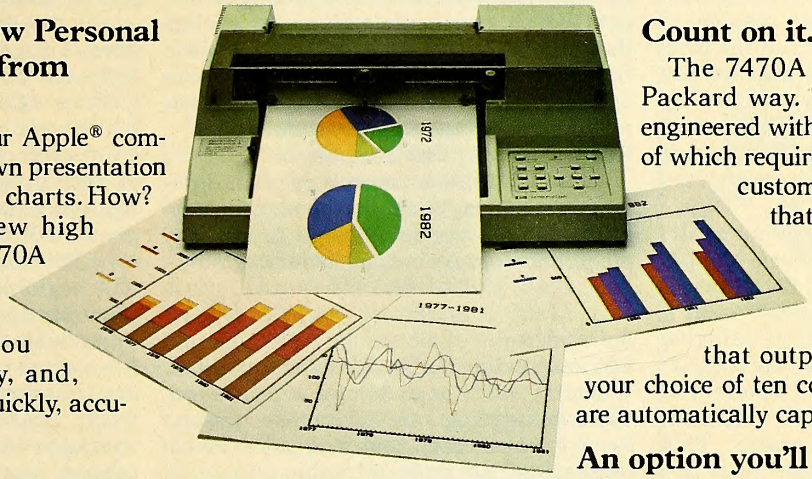
One Apple® and $1,550 can make a lot of pies.

And charts. And graphs.

Introducing the New Personal Computer Plotter from Hewlett-Packard

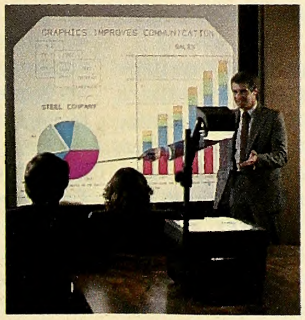
Now you can use your Apple® computer to generate your own presentation charts, graphs, and pie charts. How? Simply add on the new high quality, low cost HP 7470A Personal Computer Plotter.

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 $1,550. (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

11202 AO/05

OPEN DISCUSSION

The Economics of Software

As a person involved in producing software for the Apple, I have to respond to Steven Straughn's comments on the pricing of software. He implies that we are somehow ripping off the customers by charging $25 or more, when the cost of our raw materials is only $7. He appears to believe that the greedy producers pocket the $18 difference.

To begin with, I would point out to Mr. Straughn that we, the software companies, do not receive the full retail price. When you purchase a program from a computer store, you get several valuable services from that store: convenient purchasing, the chance to see it before you buy, assistance after the sale, even simply the ability to learn it exists! This being the real world, somebody must pay for these services, and part of what you pay for a package goes toward these. If the package passed through a distributor (which is more convenient for the computer store and allows it to give you a wider selection at the store), another chunk of the price pays their expenses. All told, the software company may well receive less than half of what you paid.

Secondly, Mr. Straughn ignored several major expenses that software producers have. For example, consider advertising, which is certainly not cheap. A full-page, four-color ad in *Softalk* costs more than $2,000. (If someone is going to suggest not advertising a product, I would point out that if neither you nor your dealer have ever heard of a product, neither you nor your dealer will buy it, so as far as you're concerned it might as well not exist.) Other expenses include paying for shipping, packaging, the labor in performing those actions, depreciation on computers, accounting and billing costs, legal fees, utilities, and rent.

Finally, please don't forget that the author needs money for the time he spent developing the software. A good programmer can earn $30 an hour writing custom software. If you develop an exciting game, market it, and wind up earning less than $5 an hour for the time you spend developing it, which will you spend more time on, your next game or a custom software package?

For the most part, the people who make up software companies are not steely-eyed businessmen out to make a buck. We're people who are earning a lot less than we could if we took up more mundane jobs, but we get a lot of extra satisfaction because we're selling products we're proud of, and we're seeing them being used and enjoyed. Unfortunately, having to justify ourselves against letters like Mr. Straughn's takes away part of that satisfaction.

Donald Brown, Des Moines, IA

Donald Brown, owner of CE Software, is known for the many pleasurable programs he has contributed to the Apple public domain as well as for CE products.

A Leg Up

I'm responding to Paul Wilson's request (January 1982, page 14) for information on copyrighting an adventure game. Since it hasn't been that long since I was wondering how to do the same thing and wishing I could write programs for a living, I know the problems and questions that occur when trying to get your feet wet in the game writing business. I hope this letter will be of help to all who are at that stage now.

When I wanted to copyright my first adventure game, *Castles of Darkness*, I called the local offices of my congresswoman and was given the phone number for the Library of Congress. I called that number and asked for the Copyright Office (the correct name is Register of Copyrights) and I was answered by that rarity of rarities: a governmental worker who was not only sympathetic but knew what she was talking about!

Whenever anyone completes a work (program in our case), that person *automatically* owns the copyright—in other words a copyright is a by-product of authoring a work. So you don't apply to Washington for a copyright, you register your copyright. In fact, under the latest law you don't even have to register it. To quote Copyright Office circular R1, page 7, "Except in one specific situation, registration is not a condition of copyright protection." That situation occurs when a copyright notice does not appear in the work. There are advantages to registering your copyright, and to doing it within three months of publication of the work. For one thing, quoting from the same page, "Statutory damages and attorney's fees will be available to the copyright owner in court actions. Otherwise, only an award of actual damages and profits is available."

Registering your copyright costs $10. You fill out Form TX and, for most programs, get a printout of the first and last twenty-five pages of the program listing. You then send your $10 check, form, and listing to: Register of Copyrights, Library of Congress, Washington, DC 20559.

Requesters may order forms from the Copyright Office by telephoning (202) 287-9100. In addition to Form TX (ask for several copies of it), you should order copies of the following circulars (they're all free): R1, R1f, 31, ML-182, and ML-212. These circulars are explanatory and should answer any remaining questions.

As far as marketing a new program goes, I suggest that you contact several software houses and request a nondisclosure agreement (reputable software houses each have their standard form) from each of them before sending a copy of your program. The agreement should say something to the effect that your program will not be shown to anyone outside the company and no manufacture will begin until a further written agreement is signed. You may also consider approaching a local store (making sure it's a good one) and asking them if they are interested in manufacturing software. There are advantages and disadvantages to both routes. When you get bids on your programs, they will almost certainly be given as the percent of manufacturer's gross to be paid as a royalty. When comparing bids, be sure to consider the method of payment (for example, quarterly, semiannually) and whether or not the company is going to receive retail and/or distributor prices for your program.

Mike Cashen, Baltimore, MD

Ten Hours!?

I received my first copy of *Wizardry* two weeks ago. I feel that it is the best game/adventure on the market. The only adventures on the Apple that I had been able to enjoy before *Wizardry* were the On-Line Adventures. I also had never liked anything similar to Dungeons & Dragons. With its graphics, *Wizardry* brings fantasy game playing to its feet! In fact, I have been playing *Wizardry* so often lately that I think I would be surprised if my Apple is able to do anything else but *Wizardry*.

Yet I have found that usually the better the program the more bugs there are, and *Wizardry* is no exception to that rule. Two weeks has been enough time for two different copies to bite the dust. Meanwhile I had acquired both the version dated last September and the version dated last December, the later one being hardly any better than the earlier. Such parts of the program that I had the most difficulty with were the Utility Options and the Training Grounds. I find it particularly discouraging to give a command that requires *Wizardry* to access the disk and then never hear from the program again as the disk goes whisping

around indefinitely. As to the Utility Options, unless they are improved Sir-Tech might as well take them off the program.

But I'm no fool; as soon as I got my new copy, I sat down with my *Locksmith 4.0* and documentation in order to back it up. I must commend Sir-Tech on their copy protection; it took me almost ten hours to copy it. Who would have ever thought that the program doesn't run unless the disk is write-protected. One final note to the authors of *Wizardry*: next time more action and less preparation so that the game does not end up becoming boring. I expect to be one of the first people to buy the next scenario when it comes out in March.
Adam Behrens, New York, NY

Off Track on Pirates
In the old days, sailing ships had problems with pirates. Two great nations used opposite solutions to reduce the piracy. Spain increased the cannon and armor on its ships. This increased the weight, cost and slowed down the vessels making them less desirable as a shipping vehicle. England, conversely, did the opposite. It lightly armed its small, inexpensive ships thus making them lighter and faster. They were better at their intended purpose—to trade and deliver cargo quickly. History has shown us that England won, not because they avoided pirates better than Spain, but because they did not let pirates get in the way of good business—in this case, efficient shipping.

This bit of historical fact has great import in the software piracy problem of today. Many companies are spending too much time and money worrying about pirates. They reduce, if not ruin the usability of their products (programs) with locked disks, unlistable programs, secret source code, hidden locks, codes in ROM chips, full-page ads devoted to pirates, etc. These devices have made many programs inefficient, costly to produce and support. The buyer is taxed greatly for he cannot make modifications or back-up copies. Often he is inconvenienced by added expenses for back-ups or future modifications. This hurts sales and angers good customers.

There is a better way as exemplified by our company Andent, Inc. We produce Apple II software for health professionals (medical/dental systems, appointments, hypnosis, et cetera). We have been in business since 1978, which makes us one of the oldest software houses for microcomputers. We are making a profit . . . and always have. We pay our bills and programmers on time. All our software is unlocked and can be copied for back-up purposes. We support all sales offering free replacement of damaged disks. All our software is listable and can be modified by the user. There is internal as well as written documentation.

We do business in this way because it pays. It pays us, and it pays our customers for buying our software. Because our software is unlocked, there is little to no problem with back-ups, updates, changes in DOS, hard disk compatibility, slot compatibility, printer slot and special character problems, disk recalls, and equipment incompatibility.

The business community, our customers, like this. They are buying a program . . . not a software lease. They have immediate support since changes and problems can be made over the phone or by letter. They can back up immediately and for as many times as needed. They like our low prices. Unlocked software is good for business, our business, your business, and the customer's business.

But what about pirates? Large scale pirates, those enterprising souls who copy our programs and sell them worldwide, are discovered and given an option to become our dealers and pay us a royalty on distributed software (or meet us in court). As in the old sailing days, reformed pirates (privateers) make the *best* dealers and we don't mind sharing the wealth. For those who don't want to cooperate, we go back in history for the remedy. The English and Spanish both learned that a few executions were good for the morale of the troops. Small-time pirates (give it to your friends) can be controlled by low program cost, registration, continuing updates and documentation. It just doesn't pay to get our programs second-hand.

Andent, Inc., believes that a sale of software is just that—the sale of programs, listings, source code, backup capability . . . a complete sale. Our customers do not buy a disguised lease—they buy and own our products. We do this because locking programs is very expensive, time-consuming, and hurts sales.

It is time the industry realizes, like the sailing ships of old, that our prime business is producing and distributing a product, not fighting pirates.
E. J. Neiburger, president, Andent, Inc., Waukegan, IL

Refreshingly Avant-Garde
Not too many weeks ago, we purchased *The Complete Mailing Label and Filing System* by Avant-Garde Creations. No author was listed.

We found the program to be exactly what we were looking for. About halfway into the program and instruction manual I had to comment to my wife that the instruction manual was absolutely outstanding. I told her that whoever wrote the instruction manual could make a fortune rewriting other companies' instruction manuals.

After spending many hundreds of dollars on software with terrible documentation, it was refreshing to spend very little money for such a well written software program as *Complete Mailing La-*

bel and Filing System. We feel that this piece of software truly gave us our money's worth.
R. G. Marcus, Seal Beach, CA

The Other End of the Spectrum
I refer to Mind Your Business in January's issue of *Softalk* in which certain software packages were reviewed. I refer in particular to the business series by Spectrum Software which your review tended to regard negatively in comparison to, say, *The Controller.*

First, there are various sizes of small businesses and while there are many who would consider *The Controller,* there are many others who could very well use Spectrum's *Microaccountant* for less outlay.

I am a small business owner engaged in international consulting and management recruiting. I, like many self-employed persons, work long hours and don't have time to fiddle around learning programming, and so on. After conducting an exhaustive investigation into potential accounting programs, I found the *Microaccountant* to be the most suitable for my business and certainly the least expensive at $49.95. I found the program and its supporting user guide to be really outstanding. (My bankers like the printouts too!)

I had an occasion to call Spectrum with a query that turned out to be an accounting question and not a fault in the program. The response I received from Spectrum was the epitome of courteous attention. I have recommended this same program to some of my overseas clients who, like me, require a straightforward financial record and control.

I now intend purchasing *The Universal Business Machine* that contains a program element similar to *VisiCalc.* Although *VisiCalc,* being in machine language, is faster than *UBM,* the capabilities of *UBM* and its price structure suits my business operation more effectively. As far as the criticism of the user guide (documentation) is concerned, suffice to say, I shall not need to attend special study classes as I might with *VisiCalc.*

All in all, I believe Spectrum Software has certain products that are really ideal and within the budgets of small businesses. Considering that a substantial percentage of American business is comprised of firms with less than ten employees, Spectrum has a lot to offer by increasing a firm's efficiency at minimal cost.
David E. Huntley, President, Huntley Associates, Dallas, TX

Not a Good Rating
I am a seventeen-year-old high school student and avid adventurer. Recently, I purchased a copy of On-Line's *Cranston Manor.* Naturally I expected, from the description given, an adventure which would take a while to solve. I expected a huge house, unusual combinations, and an old man's ghost wandering throughout the manor guarding his treasure. What I got was a house with about twenty-five rooms (although there are caverns under the house), treasures just lying out in the open, and several bugs. Also, when I called On-Line, they said that Old Man Cranston's spirit had no bearing on the game; he wasn't even there!

I am now stuck with a $35 adventure that took me five days to master playing at about two games each day. This, to me, was an unwise investment. Of course, I had no guidance except for On-Line's advertisement which apparently misled me. I figure that the least I can do is warn other adventurers of this disk. Although the format and the graphics are good, it is a beginner's adventure which will be solved in no time.

I definitely agree that *Softalk* should rate adventures. This would be beneficial not only to people in my case, but also to novices who may otherwise wind up with the hardest adventure on the market.
Reed Hubbard, Jackson, MS

Crystal Comments Continue
Thank you for your continued attention to and discussion of the controversy involving Crystalware products. Let me add my experience with their merchandise.

The Softwaire Store, which I man-

age, received its first shipment of Crystalware product in September 1981. Since then, we have experienced difficulties with virtually their entire line. Whether attempting to exchange defective games or receive credible explanations for specific problems, we have rarely received any cooperation. (One notable experience: when preparing to exchange our second, supposedly debugged shipment of *Waterloo*, I was asked if I "wanted to know the truth" about the problems we had with ten separate copies of the program. I was dumbfounded.)

Having tried, in vain, to locate among their products one that both works reliably and is sales-worthy, we have tried one last desperate ploy; our entire inventory of Crystalware software is piled high on our bargain table, priced at 40 percent off, and guaranteed to boot, period. Unfortunately, we haven't sold much of it.

Perhaps, however, I should be encouraged. It may be that word-of-mouth regarding this company is so bad that they will soon move on to a business whose demands they can better cope with. On behalf of myself and consumers whose first requirement of a software package is that it work properly and reliably, I hope that is the case.
Charles D. Limmer, Jr., Los Angeles, CA

As an owner of Crystalware's adventure, *2041 AD*, I look forward with interest to your review of that game in your next issue. My version of *2041* is the 1982 revised for DOS 3.3. After struggling for two weeks, I finally returned diskette 1 (the first two adventures) to them for replacement. My two problems were as follows:

1. The save game function just doesn't work. Regardless of whether I invoked the save in Congoland or the Arabian Adventure, when I restarted the game, it always put me on disk 3 in Merlin's Hideaway.

2. If my character was moving along the edge of the screen and something appeared next to me (i.e., a group of warriors in Congoland or a caravan in Arabian Adventure), my character froze and I was unable to move it in any direction. I could do everything else—trade, attack, greet—except move. Thus I had to end the game. This is very frustrating when you have spent several hours advancing through the game only to find you are forced to start over.

Mr. Lillicrap's experience with his Crystalware game (*Softalk*, February 1982) is not unusual. Last Christmas when I visited a software store in West Los Angeles, I found all Crystalware games at 20 percent off. I asked the clerk why. They told me that their customers have returned so many Crystalware games for replacement that they decided to sell off all their Crystalware merchandise and drop the line. As I had just spent $60 for *2041 AD* I was somewhat distressed. I hope my replacement disk for *2041, AD* solves the two problems I just mentioned.

It seems to me that for such an expensive game I should receive something that at least allows me to complete the game or play it to my limitations rather than being stopped by a program bug that should have been discovered before the game was offered for sale.
David Jameson, Boise, ID

After reading the Open Discussion column in the February issue of *Softalk*, I felt compelled to reply to the rebuttal from Crystalware.

Early in 1981, I sent Crystalware $39.95 for a copy of *Sands of Mars*. After waiting five weeks with no reply, I asked for a refund. Instead, I finally received a copy of *Sands of Mars* on two disks.

This very early version of the game had an unbelievable number of bugs. It was obvious that neither John Bell nor any of his employees had tried to play the game even once. The game couldn't be played because of all of the bugs. But since I had just finished playing *The Prisoner*, I was prepared for almost anything. I played around the bugs as best I could. So it bothers me when Mr. Bell says we are more interested in finding faults with the game than enjoying it.

I called Crystalware, but I couldn't reach Mr. Bell. Instead, my name and address were taken, and a week later I received a list of *twenty-four* bugs. At least this made it possible to play a little further. But the letter also offered $10 per bug found by users of *Sands of Mars*. At that rate, I would own their company in no time. I telephoned again, but still they would only take my name and address.

Finally, I returned the game and asked for either a refund or a working version. Needless to say, they sent another copy, which still had many bugs. I wish to emphasize that I would much rather play the game than criticize it. To date, I have successfully voyaged to Mars reasonably intact and mapped the labyrinth of the Kendra (long-dead race of Martians in the game's history). But when a Kendra attacks, I have the choice of being killed, which ends the game, or firing my laser, which crashes the program. How does one cope with a situation such as this? Isn't playtesting a prerequisite for selling a commercial game program? Should we pay $39.95 and up for the privilege of debugging Crystalware's games? It appears that Crystalware is as guilty of white collar crime as any pirate.

If Crystalware will not issue refunds to all those who have purchased their defective products, I suggest that we write to the editors of *Creative Computing*, *BYTE*, and anywhere else Crystalware products are advertised.

The worst of this situation is not the one-time loss of $40, but the loss of faith in other software publishers. We Apple users are accustomed to buying soft-

ware sight unseen. All we know of a product is its advertisement. So the next time we see an advertisement for an exciting game by an unknown publisher, we may wonder if this company is another Crystalware. This insecurity would do more damage to the hobby than anything that could be done by Crystalware.
Victor De Grande, Staten Island, NY

An Addendum to Muse
I enjoyed your EXEC MUSE profile in the February issue. However, I feel it is necessary to point out a few mistakes and glaring omissions by the writer. First, Jim Salmons did not leave his job as marketing director in July of last year, but rather a full two months earlier. I was hired as his replacement in May and assumed the post on June 1. The article fails to mention me or the role I played in bringing many new dimensions of positive customer-oriented marketing to Muse. Also ignored is my creative role, particularly in regard to my work with Silas Warner in evaluating new products and writing a follow-up game to the successful and entertaining *Castle Wolfenstein.* Nowhere are any of my contributions mentioned. I find this omission, along with the inference that Mr. Salmons' successor was a "temporary" replacement, to be a disservice not only to me but also to the hundreds of customers, journalists, programmers, and retailers whom I personally served and worked with on a daily basis for almost eight months.

I am proud of my accomplishments with Muse. Like other ex-employees who worked hard for Ed Zaron yet eventually disagreed with his business practices and methods, I feel your writer could have done more to give credit where credit was due
Thomas A. Jackson, Baltimore, MD

Bucking Bucky
I'm sorry that I cannot share your enthusiasm for Buckminster Fuller. As an outstanding, innovative designer, he has few peers. But as a city/world planner, he is dangerously totalitarian.

I've skimmed his book, *Critical Path,* where many of the ideas expressed in your article (January 1982) are expounded, and he makes me shiver with his bland disregard for the individual as he sweeps grandly on to his stupendous solutions for humanity's manifold problems.

His casual acceptance of Soviet deceitful actions, such as in attempting to hide the actual locations of towns and cities, deliberately altering maps available to the outside world, shows why he is held in high regard by the Kremlin slavemasters. It's one thing to say that such actions are to be expected, but quite another to practically welcome them.

I agree that there are "entrenched interests" whose activities are definitely *not* in mankind's best interests. But it does not occur to him that, just maybe, his grandiose philosophy fits right in with theirs. After all, it was the Wall Street bankers who bankrolled and fostered the phony Russian "revolution."

I think his concepts, such as "Synergetic-Energetic" geometry, "Geodesics" and "Dymaxion" systems, including his three-wheeled automobile whose streamlined design is still light-years ahead of the present, show a brilliance equalled by very few. But, like Bertrand Russell and Albert Einstein, he displays profound economics illiteracy and complete disregard for the individual.

Incidentally, aren't you inviting raised eyebrows by calling your column "Newspeak," a la Orwell's "1984"?

Thanks for listening.
Howard S. Balsam, Nashua, NH

The very point of using "Newspeak" as the title of a column filled with news of the spread of computers into the hands of increasing numbers of private individuals is to say that 1984 is nearly here but Orwell's world isn't; and that, to the extent that Orwell's predictions crop up in reality, the very existence of privately owned micros is one of our best tools to combat the Orwellian threat.

That's Good! That's Bad
I find it extremely incongruous to read and be stimulated by the appearance of the article on Buckminster Fuller in your January 1982 issue, calling for the fullness of our experience to lend logic and technology to our very survival, only to turn the pages to find Howard Software Services' ad using a woman as bait for the book *Creative Financing.* My first guess is that there is indeed nothing *creative* in those pages and my second is that Howard excludes a tidy market for their book by such uncreative selling—I'll certainly look elsewhere for fresh information on the subject.
Jan Sutherland, Berea, KY

Kudos and a Question
To Doug Carlston: This evening I had the opportunity to read your article All About Applesoft in the January issue of *Softalk.* I am really looking forward to reading the February issue now.

We have had a family Apple II Plus for about a year now and are still struggling through the operational learning process. We're gaining, but I *did* pick up two formerly not known or not understood procedures from your January article . . . please keep it up!

My question to you concerns *The New Step By Step* by Program Design, 11 Idar Court, Greenwich, CT 06830. The only advertisement I have seen (October 1981, *Creative Computing)* bills it as "A Programming Course for Beginners." Huntington Computing (my nearest mid-Michigan dealers have never heard of it) tells me, "If we get ten units we have fifteen orders for it." Otherwise, I have been unable to find any reviews of this program. Have you heard of it (or have you seen it) and would you recommend it above anything else (what else)? for someone who could hardly spell c-o-m-p-u-t-e-r prior to last year?

P.S. Thanks to a number of programs we have been able to "punch in," the Apple is paying its way; but we would like to know *why* it does what it does.
Robert E. Daily, Alma, MI

Looking for Alignment
I am looking for a simple Basic or machine code routine that would cause numerical input or calculated results to be displayed in right-justified dollar and cents format. I have a homebrewed routine that works, but I'm sure that more streamlined and efficient routines exist.
Frank E. Merrill, Indianapolis, IN

Offering Alignment
I'm a new reader but an avid Appler.

Enclosed is a program I have written and perhaps there is a place for it somewhere in *Softalk.*

The first thing I found out about my Apple was that it could not print that trailing zero. For example, to print $10.10 was a no-no. Apple could not print that last zero. Even worse, Apple would print $10 with no zeros and no decimal point.

After a load of reading, asking, and trying, here is the solution and it is much simpler than I ever dreamed. You, too, will be able to print the last zero or two zeros after the decimal. Just follow the directions, but exactly!

Try the following program:

```
20  S1$ = "0":S2$ = "00": D = 2
30  READ X
40  X = INT(X*100+.5)/100
50  X$ = STR$(X)
60  FOR I = 1 TO LEN(X$)
70  IF MID$(X$,I,1) <> "E"THEN NEXT I
80  FOR J = 1 TO I - 1
90  IF MID$(X$,J,1) <> "."THEN NEXT J
100 IF J+D<=I THEN N=J+D:GOTO 130
110 IF J+D=I THEN N=J+1:GOTO 160
120 IF J + D > I - 1 THEN N = J + 2: GOTO 190
130 W$ = LEFT$ (X$,N) + MID$ (X$,I)
150 PRINT W$:GOTO 20
160 Y$ = LEFT$(X$,N)+MID$(X$,I)+S1$
180 PRINT Y$:GOTO 20
190 Z$ = LEFT$(X$,N)+MID$(X$,I)+S2$
210 PRINT Z$:GOTO 20
220 DATA 12.34, 123.4, 1234.567, 1234.500
230 DATA 1523.1, 16344.0
```

When you run this program, this is what you will see:

```
12.34
123.40
1234.57
1234.50
1523.10
16344.00
```

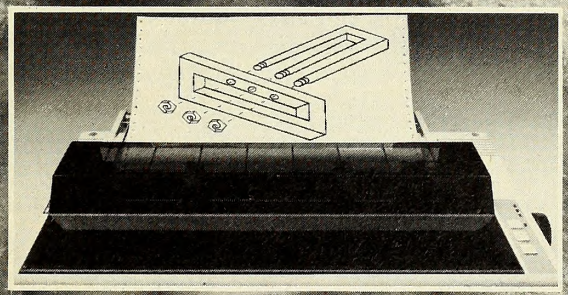
So far, it is fine. By inspecting the data, you can see that the program added a zero or two as required and a deci-

IMAGINE IT..

CAPTURE IT.

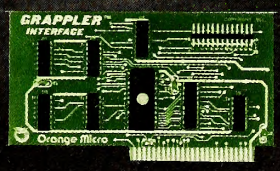
The Grappler™ is the most intelligent Apple interface available. Simple commands from the keyboard or user program, are all you need to dump screen graphics to your printer. The Grappler's exclusive EPROM chip makes high resolution graphics that easy. There are Grappler versions to accommodate the Anadex, Epson Series,* IDS Paper Tigers, Centronics 739, NEC 8023, C. Itoh Pro-writer, Okidata 82A,* Malibu 200, and future graphic printers.

*Requires Graphics Upgrade.



ACTUAL APPLE II PRINTOUT USING GRAPPLER™ AND EPSON MX100

With The Grappler™ Printer Interface



The Grappler's features include:

- Graphics Screen Dump
- Inverse Graphics
- Emphasized Graphics
- Double Size Picture
- 90° Rotation
- Center Graphics
- Chart Recorder Mode
- Block Graphics
- Bell Control
- Skip-over-Perf
- Left and Right Margins
- Variable Line Length
- Text Screen Dump.

The Grappler™ works with Pascal and CPM.

Orange Micro inc.

3150 E. La Palma, Suite G, Anaheim, CA 92806

(800) 854 8275 TOLL FREE

(714) 630 3322 CA; AK; HI.

Starfire.

The Winchester disk made just for Apple II.

You can transform your Apple II into a powerful business computer. How? By using the Starfire, a Winchester-style disk reasonably priced at $2995.00.

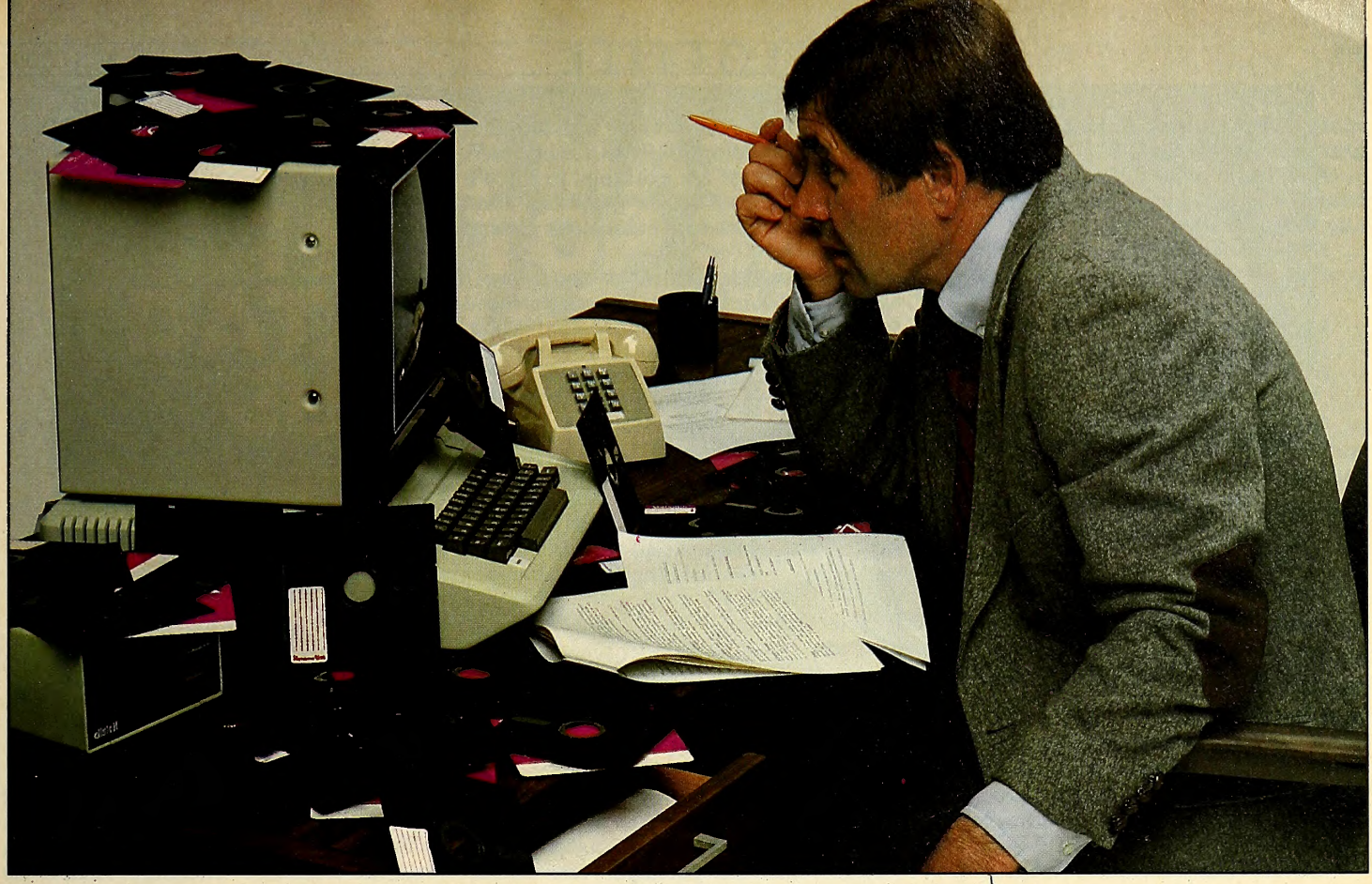
Starfire provides the qualities you'd want of a computer; smooth responsive interaction, high speed, real safety of stored data and large data files.

Storing all the information on a rigid disk, sealed in a contamination-proof container allows for large amount of storage, fast operation and remarkable reliability.

With a Winchester disk, all your programs and data are available instantly at the touch of a key, without fumbling and searching through floppy diskettes. Implemented with the right software, a Winchester disk will give you everything you'd expect and should get from a business computer.

Why wait? Call us at Corona now for more information. We'll tell you where to find the dealer nearest you.

corona
data systems



We don't make a Winchester for just anybody.

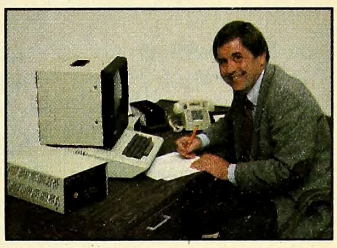
We're Corona Data Systems and we've made Starfire, a Winchester disk just for Apple II*.

What's in it for you?

5 million characters of storage. And that's not all. You'll get Corona's Disk Partitioning which allows for up to 16 separate operating systems such as Apple DOS, CP/M* and Pascal, all sharing the same disk. Of course, you'll still be able to use DB Master* and protected software like VisiCalc*.

Double Size Volumes.

And with our Double Size Volumes, your Apple DOS is transformed into something special that offers larger data files. And you'll never have to type commands like "catalog" and "run" or type program names again. You can even forget what volumes the programs are in.



Reliability.

But you'll never forget Starfire's reliability. That's because we built DataGuard,* the feature that will correct errors, not just report them. And we also provide a one-year warranty that you can count on. That's reliability.

So why take chances? With Starfire, you're not just adding a hard disk to an Apple II, but transforming an Apple II into a full hard disk business computer.

corona
data systems

21541 Nordhoff Street, Unit B Chatsworth, California, 91311 (213) 998-0505

Starfire – the Winchester disk made just for Apple II.

*Starfire and DataGuard are trademarks of Carana Data Systems * Apple II is a registered trademark of Apple Computers, Inc. *VisiCalc is a trademark of Personal Software Inc.
*CP/M is a registered trademark of Digital Research, Inc. • DB Master is a trademark of Staneware

mal point if needed. Rounding off of the data was accomplished in line 40.

Now the next problem is to print the output so that the decimal points are aligned vertically. Right?

Easy! Just add the next three lines to your program.

```
140 HTAB(20-LEN(W$))
170 HTAB(20-LEN(Y$))
200 HTAB(20-LEN(Z$))
```

When you now run the program, lo and behold! Here is the resulting printout.

```
12.34
123.40
1234.57
1234.50
1523.10
16344.00
```

The location of the column of numbers is determined by *HTAB* in lines 140, 170, 200. Note that the *HTAB* number determines the right-hand location of your printout. Read the next sentence slowly. To change the horizontal location of the printout, change the *HTAB* number in *all three* lines, got it?

To use this subroutine in your programs, do the following:

1. Delete lines 30, 220,230
2. Write *10 goto 300*
3. Start your program with line 300
4. In your program just prior to printing of variable, set X = to your variable and then write *gosub 20*
5. In lines 150, 180, 210, change *goto 20* to *return*

Do these step by step and you will be successful.
Ralph P. Weiss, Tamarac, FL

One Man's Meat

The review of *Utopia Graphics Tablet Software* correctly pointed out the inadequacies of the Utopia manual, but it failed to mention anything about the serious bugs in the programs. Indeed, Apple Computer was aware of this when they withdrew the original version of Utopia and replaced it with "version 01." Unfortunately, this new version is also defective. And my contacts with representatives of Apple confirm the fact that they are aware of its bugs.

For prospective buyers—the animation routine is highly erratic in its performance. It does not store the number of sequences specified in the manual and out-of-memory messages occur unexpectedly, with no way of saving the work already done. Unanticipated illegal quantity or bad subscript error messages occur frequently for no apparent reason. Saving sequences on disk is extremely hazardous—previously saved sequences or an entire disk may be wiped out in the process of saving the most recent set of sequences. The Director Fix routine does not function at all.

The Paintbox routine also has a serious flaw. The sketch mode and line plot mode are not compatible with each other. There is no way, for example, to trace a drawing attached to the graphics tablet using both of these as extensions of each other.

It is unfortunate that the *Utopia Graphics Tablet* program was released without careful performance evaluation. It is even more disturbing when *Softalk* fails to mention serious defects in the operation of the product.
Lester Blum, Hamilton, NY

I can never remember having seen a more shortsighted, shallower review than that given to the *Utopia Graphic Tablet Software* package. I sat transfixed for two solid hours while the author demonstrated feature after feature of what can only be termed the most comprehensive graphics package on the market today for producing still images. And (and!) it has an animation mode for kicks.

The package is so self-learning that it really doesn't need (repeat, doesn't need) any written documentation. The fact that Special Delivery Software decided to give it twenty-four pages of poor support is a reflection on the publisher (Apple), not the package. The author never removed his eyes from the screen while running through the demo because there was no need to. The keyboard is almost wasted (good riddance!) and the screen cursor moves so well that you simply "feel" the graphics tablet under your hand.

The range and power of this system has to be experienced at first hand to be believed. The reviewer was correct in only one respect. It isn't designed for six-year-old children. It is designed for intelligent adults.
Brooke W. Boering, Schaumberg, IL

One Bug Leads to Another

I have recently discovered a very interesting bug in the Applesoft interpreter. The occurrence of this bug is a function of the *line number* used in the program. It involves using *gosub* in immediate mode—allegedly a legal operation. Please see the attached listing, and try it yourself if you don't believe it (this was Doug Carlston's reaction when I mentioned it to him at Broderbund's Halloween party—he immediately kicked *Midnight Magic* off the machine and tried it).

```
]NEW

]50 GET A$

]60 RETURN

]LIST

50 GET A$
60 RETURN

]GOSUB 50

]NEW

]LIST

1500 GET A$

]510 RETURN

]LIST

500 GET A$
510 RETURN

]GOSUB 500

?SYNTAX ERROR IN 0
]NEW

]2000 GET A$

]2010 RETURN

]LIST

2000 GET A$
2010 RETURN

]GOSUB 2000

?SYNTAX ERROR IN 48
```

As you can see, when the first line of the subroutine is 50, there is no bug. But when the first line number is 500, we get a "syntax error in 0" message. And, if the first line number is 2000, then we get a "syntax error in 48" message! Isn't this the weirdest thing you've ever seen? I'd like to see an explanation if someone can find one—it probably has to do with a bug in Applesoft's *get* command.
Bruce Zweig, Lightning Software, Palo Alto, CA

Trying this, and remembering a long ago tip from mentor John Haller never to get without printing what you got, we tried the programs as shown, with the same results, then tried them with each get *line replaced with a get,print line:*

```
]GET A$:PRINT A$
```

Result? Works fine. So it appears the mysterious bug may be in the get *statement rather than the* gosub.

It's still a mystery, though. Any answers?

Life Begins at Forty (We Know)

I'm a novice Apple II owner and also I'm forty-plus years old! You'll see why I say that in a minute.

This is not only a letter to give my appreciation for a very helpful magazine but also to say that when my "freebie" time runs out I will have no reluctance to continue my subscription on a paid basis. Keep up the excellent work!

Now the explanation. One of the most beneficial parts of *Softalk* to me has been the Open Discussion letters. And as an example I offer this letter. Being the greenhorn that I am I learned from Diane Durbeck's letter how to use the lower case text of my Epson printer in a relatively simple subroutine. The "old guys" can learn a lot from the young people if we'll just listen. Thanks Diane.

Now if someone will just teach me

DB MASTER.™ THE TOP SELLING DATA BASE MANAGER JUST GOT BETTER.

In less than a year, DB MASTER has become the top selling data base manager for the Apple II. And for good reason. DB MASTER has the features that make information management easy and efficient. But we didn't stop there. You asked for more features. We listened—and made DB MASTER even better.

MORE FEATURES IN DB MASTER VERSION THREE.

Computed Fields. Perfect for accounts receivable, inventory control and similar applications. Each record may now include up to 10 computed fields. And field values are automatically re-computed and displayed each time a record is edited.

Totaling in the Search Mode. Tell DB MASTER which field to total and which records to use. A running summary of records found and the field's sum, average and standard deviation are displayed on the screen.

Audit Trail. Option to automatically print each new record as it is entered.

NEW REPORT GENERATOR OPTION CAPABILITIES:

- Number Formatting with commas.
- Auto-Date Record Selection for printing daily, monthly or annual reports.
- Printing of averages and standard deviation when printing column totals.
- Ability to make last minute changes in printer and report parameters.

PLUS THE FEATURES THAT MADE DB MASTER NUMBER 1.

Dynamic Prompting™. User designed screen forms. Short form capability. Powerful report generator. Custom DOS for faster retrieval and program chaining. Automatic data compaction. Password file protection. Multiple sort keys for fast, convenient records retrieval. And much, much more.

MORE POWER WITH DB MASTER UTILITY PAK #1.

Restructure. Modify a file format without re-entering your records. Add, delete, move, modify or re-name fields—even add new computed fields to existing files.

Interface. Move data back and forth between your DB MASTER files and Data Interchange Format (DIF)* text files. Exchange data with Visicalc*, Visiplot*, Executive Secretary*, etc., or your own programs.

Replicate. Duplicate everything in your DB MASTER files *except* the records. Use the replicates for monthly or yearly files, or send them for data entry at multiple locations.

Merge. Combine data from two or more files with the same format (such as Replicates) into one file. Special "Selective Merge" and "Merge and Delete" options can maintain separate active and inactive files, build subfiles, and many similar tasks.

UPDATE YOUR DATA BASE MANAGEMENT WITH DB MASTER VERSION THREE.

See the new DB MASTER Version Three at your local computer store. Put it through its paces. Then add the power of Utility Pak #1, the first in a planned series of Utility Paks.

If you're a registered owner of an earlier version of DB MASTER, send one or both of your program diskettes, plus $15.00, to STONEWARE for an update to Version Three.

50 Belvedere Street, San Rafael, CA 94901 (415) 454-6500
TELEX II: 910-384-2016 STONEWARE SRFL

how to get a "comma" I'll be flying. . . .

Thanks again for *Softalk*. I'm glad my Apple dealer told me about you.
Albert E. Hoffman Lexington KY

U&LC Review

This program in response to the article in February 1982 *Softalk* by an intelligent young lady from San Jose, CA. I could not get her program to run on my forty-column printer, which has lower case capability, and I have not lower case on my Apple II yet. I was intrigued by the possibility and spent a couple evenings brewing this one up.

```
10  A$ = "@THE @UNITED @STATES OF
    @AMERICA"
20  Z = LEN(A$):FOR Y = 1 TO Z: IF Y = Z THEN
    END
30  X = ASC(MID$(A$,Y,1))
40  IF X = 64 THEN 70
50  IF X > 64 THEN X = X + 32
60  PRINT CHR$(X);: NEXT Y
70  IF X = 64 THEN X = 32: Y = Y + 1
80  X = ASC(MID$(A$,Y,1)): GOTO 60
                    2264 bytes
          The United Stotes af America
```

Charles Persoon, Altamonte Springs, FL

I appreciated the program presented by Ms. Durbeck in your February 1982 issue on lower/upper case generation.

Enclosed is an enhanced version of that program, which attempts to maintain left and right margins and to avoid overriding when more than eighty characters are used (MX-80).

Of course, many improvements to this program can be made, but this would be beyond the purpose of a relatively simple program to print out statements from Applesoft Basic in lower/upper case.

```
10  SENT$ = "@SOME CAPITALS OF THE
    @UNITED @STATES OF @AMERICA AND
    THEIR CORRESPONDING @STATES ARE:
    @JUNEAU, @ALASKA; @BOISE, @IDAHO;
    @TOPEKA, @KANSAS; @ST. @PAUL,
    @MINNESOTA; @CONCORD, @NEW
    @HAMPSHIRE; @SANTA @FE, @NEW
    @MEXICO; ETC."
20  GOSUB 60: PRINT : PRINT
30  SENT$ = "@THE @UNITED @STATES OF
    @AMERICA."
40  GOSUB 60: PRINT
50  PR# 0: END
60  PR# 1:HC$ = "":M1 = 5:LM = 1:RM = 70:R0
    = RM:B =0:C = 0: FOR J = 1 TO LEN (SE$)
70  REM IN #60, M1=INITIAL MARGIN; LM & RM
    DENOTE SUBSEQUENT LEFT MARGIN &
    NO.CHARS.TO BE PRINTED + MARGIN
    (M1+RM<=81; ALSO FOR LM+RM)
80  HC = ASC ( MID$ (SE$,J,1)):F = 32 * (HC >
    64):HC = HC + F
90  IF HC = 64 THEN J = J + 1:HC = ASC ( MID$
    (SE$,J,1))
100  HC$ = HC$ + CHR$ (HC)
110  NEXT J
120  REM PRINT NO SPACES AT LEFT-MOST
    MARGIN, MAINTAIN RIGHT MARGIN, &
    DON'T OVERRIDE WHEN >RM CHARS.
    (RM<=80)
130  IF LEN (HC$) < R0 THEN PRINT TAB
    (M1);HC$: RETURN
140  HTAB M1: FOR K = 1 TO LEN (HC$): IF K /
    RM = INT (K / RM) AND MID$ (HC$,K,1) =
    " " THEN K = K + 1: PRINT : HTAB LM:B = B
    + 1:C = C + 1:RM = (C + 1)*R0 + (M1 -
    LM - 1) * C + B: GOTO 160
150  IF K / RM = INT (K / RM) THEN PRINT : HTAB
    LM:C = C + 1:RM = (C + 1) * R0 + (M1 -
    LM - 1) * C + B
160  PRINT MID$ (HC$,K,1);: NEXT K
170  RETURN
```

Same copitols of the United Stotes of America ond their corresponding stotes ore: Juneau, Alasko; Boise, Idoha; Topeko, Kansas; St. Paul, Minnesoto; Cancord, New Hompshire; Sonto Fe, New Mexico; etc.

The United States af America

L. S. Reich, West Orange, NJ

Keep It Simple

I am in dire need of assistance from any source!!!

I purchased a Mountain Computer RAM + 32K card on dealer recommendation then we found, to our chagrin, that the documentation is grossly incomplete for a nonprogrammer/neophyte and the card did not include software examples nor any support other than a diagnostic of sorts.

Current month advertising specifically warns that "user software support" is required, so I must assume that others were also "bit"!

I have written to the company in question to no avail and would appreciate any assistance a reader or yourself could offer with examples.

I have an Apple II Plus with the Integer Basic Firmware card and dual drive DOS 3.3. My main support programs are *SuperText II*, *CCA DMS*, and *VisiCalc* plus sundry arcade games and adventures for the "others." I would like to expand to *DB Master* in the near future.

If my support (e.g. *SuperText*, *CCA DMS*) requires personal modifications, I consider the subject product to have been misrepresented without clear/precise statements of nonprovided support and the manufacturer to be "loose" in support of their consumer and dealers.

It would seem that this condition does exist in the industry, but be warned that another "Detroit syndrome" does not develop into further abuses and subsequent losses to the originators.

In closing, I would like to thank you for any and all assistance which you can provide and would suggest that this question should serve as a notice to others to require *complete* detail, not simple advertising statements, prior to future purchases of any commodity.
D. J. Goudy, Poughkeepsie, NY

Help an Enterprising Appler

I need your advice on a modification that I wish to make to my *Apple Writer*. I am attempting to make some extra cash by selling little cards with flight information on them to airline pilots. I have to insert the customer's name into the format, which works as a copyright. This means that to print a set of twenty cards for one customer, I must constantly be going back and forth between the editor and print menus. All of this results in a good hour to print out one set. Now my question. Is it possible to modify the *Apple Writer* with a program that would insert the name, print it out, and go on to the next text with little direction from me?
Eric Challgren, Severna Park, MD

Apple Writer Stress

I am using the *Apple Writer 1.1*. My question concerns underlining. Is there any way to underline words with *Apple Writer*, short of entering a line composed of spaces and the – character immediately below the line containing words to be underlined?
Martin LaBar, Central, SC

String Packing

I am interested in a technique for converting a string to a packed character format. If anyone has devised a method to do this, I would appreciate seeing the technique discussed in Open Discussion.
Richard F. Gilberg, Sunnyvale, CA

Puzzled by Loopless Print

I am a creator of word puzzles, as a means of keeping my retired husband and self off the poverty level of our retirement checks.

I have a good program for my Apple II Plus that places words in a diagram for Word Search puzzles. But we have to circle the words for the answer diagram (printed on our Epson MX-80), by hand. I think our Apple must be feeling insulted by this snubbing of its capabilities.

Can someone out there tell me how to persuade the Apple and the Epson to draw "circles" (loops) around the words in one copy of the diagram—and leave the other copy uncircled? The action needs to be automatic and not take a lot of extra time. I need to draw 45 to 50 loops to encircle that many horizontal, vertical, and diagonal words in a 15 x 15 square of letters; words from 3 to 11 letters in length.

Help!
Naida Dickson, Gardena, CA

Printing Pascal

Like Jim Burke of Iowa City in the January issue, I have both an Epson printer and the Apple Pascal 1.1 system. I have used the printer successfully with both the Epson parallel card and the Apple parallel card. The only difference I find is that the Epson card does not require the use of the Linefeed program provided on the Apple3: diskette while the Apple card does. Mr. Burke's problem may be due to the presence of Linefeed.code on the boot diskette as System.startup. Use of Linefeed in this way is described on pages 214-215 of the Pascal operating system manual.
Joann Wykoff, Dowling, MI

Enclosed is a small program, *PRINTIT*,

which I wrote to demonstrate what I have learned so far about sending data to the printer from within a Pascal program, a process which I have not found in any of several texts on Pascal, and about which another reader had asked.

I would like to add my strokes to the many you have received about the magazine; I read it avidly each time I receive it.

I would like to suggest that you consider placing more focus, with regard to Pascal, on what is *not* found in the texts. The basic instructions you review in the Pascal Path have been done well elsewhere. However, I cannot find any good in-depth discussion of file-handling, pointers, trees, etc. These would be the areas that I personally would find most rewarding. The best text I have found, by Luehrmann and Peckham (from which I learned Pascal after having read several other books), treats files very minimally, and pointers not at all.

```
PROGRAM PRINTIT;

  VAR F: INTERACTIVE;

BEGIN
  CLOSE(OUTPUT);
  REWRITE(OUTPUT,'PRINTER:');
  WRITELN('File not nomed, so goes to OUTPUT,
  which is PRINTER:');
  WRITELN(OUTPUT,'WRITTEN to OUTPUT');

  CLOSE(OUTPUT);
  REWRITE(OUTPUT,'CONSOLE:');
  WRITELN('OUTPUT chonged to CONSOLE:');
  WRITELN(OUTPUT,'Still the some');

  CLOSE(F);
  REWRITE(F,'CONSOLE:');
  WRITELN('F TO CONSOLE');

  CLOSE(F);
  REWRITE(F,'PRINTER:');
  WRITELN('No File nomed, so goes to Console!');
  WRITELN(F,'F nomed ond set to PRINTER:');

END.
File not nomed, so goes to OUTPUT, which is
PRINTER:
  WRITTEN to OUTPUT
  F named ond set to PRINTER;
```

Jerome Blumenthal, Binghamton, NY

Absent-Minded VisiCalc

Help!

I have a *VisiCalc 3.3*, 13-sector storage diskette that lost the Directory.

While combining 13-sector storage diskettes so that I could muffin for use in my updated *VisiCalc* 16-sector version, I got a prompt "error; disk full." At that point I tried the following:

Sequence	Prompt Response
/SL (NAME OF PROGRAM® THEN,	"FILE NOT FOUND"
/SL (RIGHT ARROW) THEN,	NO FILE TITLES
/SS ("SAMPLE")®	"ERROR: DISK FULL"

It seems as though the data is on the storage disk but somehow(?) the Directory is gone.

Personal Software says they have no ideas, other than to suggest finding someone who knows how to write to or has a program that will retitle the storage diskette for accessing purposes.

Can you help?

Please don't say I *should have* maintained backups—I do and did! These were files I was bringing together so I could go to a 16-sector version. Obviously, in retrospect, the approach could have been different but the damage is done and to reconstruct the information would take hours.

Paul Robbins, Waterville Valley, NH

Remembering Darth

Though I am sure you have heard it before, yours is a great magazine. I have found that the Assembly Lines series, the Pascal Path series, and the product reviews are of invaluable aid. I am utterly convinced that the Hardtalk series and the Softcard Symposium will each in their turn be my favorite feature. Without a doubt your magazine is essential to my keeping informed about the comings and goings in the Apple world.

I have a question I would like to put to the general readership. How do you get lower case on a Pascal Apple? I have a lower case board, sold by Pygmy Programming, which uses the shift keys (without the Dan Paymar modification) and it works well in Applesoft and Integer Basic. However, no matter what I do I can't seem to get lower case in Pascal. By the way, for Applesoft and Integer, Mr. Samson has removed the masking imposed by the F-8 ROM which bumps all the keyboard characters to uppercase.

On another note, I would like to add my two bits about your advertising. I don't have any children and am not an overly moral person, yet I would like your magazine to remain a tasteful publication. Ads which utilize excessive sexuality are degrading and beneath the level of most Apple users. That is at least my opinion, take it for what it is worth.

I have had my checkbook open now since January waiting to write you folks a check. As of yet I have received no renewal notice and am apprehensive that I will miss out on one of your superb magazines. If I owe you folks for subscription costs please let me know.

I frequently comment to my friends on the rapid growth and popularity of your magazine. I would have never guessed that the little Darth Vader magazine would become the mouthpiece for all of Apple. Your magazine is so good I have dropped my subscription to *Byte*. Please keep up the good work, and reflect upon the nature of your readership.

Lt. Tommy C. Gray, Flagstaff, AZ

Tipster and Friend

In my quest to find a better and easier way to format REM statements, I discovered that a *poke 33,28* before typing the *REM* and then a *poke 33,40* (or *TEXT*) after the *REM* has been entered results in a perfect vertical format for more than one line in a *REM* statement. This makes for nicer looking listings, and it is easier to find sections of a program under construction.

This tip of the week is presented free of charge to all Apple owners and readers of *Softalk* magazine, courtesy of one little-known employee of the Byte Shop of Fort Lauderdale, Florida.

Art Christopher, Boca Raton, FL

Try VHF 10, 11, or 12

I would like to inject the output of my Apple to bypass the RF stage of my Heath kit TV set. Since my UHF reception is very poor and I think it is the fault of the UHF tuner, I was hoping I could get much better display than I now can with the Sup-R-Mod II on channel 33.

Every time I try, the output transistor on the Apple overloads and I have to replace it. It seems like either an impedance mismatch or I'm just injecting in the wrong places. Can anyone help me?

Harvey Waxman, Paxton, MA

Not Talking

As a newcomer to the computer game, I wanted a way to give myself both the ASCII codes and the hexidecimal codes for the keyboard characters. Richard A. Patton, in the January Open Discussion, gave me a start. I combined his conversion program with one I had developed and came up with a program that lists ASCII codes, hexidecimal equivalents and keyboard characters.

How do I get my Apple II Plus to operate my GE TermiNet 300 printer? I have a Mountain Computer CPM card.

Tom Bredehoft, Saint Louisville, OH

The Porn Is Green

Since *Softalk* will no longer be mailed free to Apple owners, I think it is time you settled once and for all the controversy over "adult" ads. Even those who recognize your right to publish such ads might react differently when asked to subsidize them.

The central question is whether *Softalk* wishes to be a magazine for the entire family or one which is intended only for adults with a liberal outlook. The debate is not over whether these ads should be published, but whether *Softalk* is the appropriate forum. We must recognize that we are members of a profession which over the next decade will have a profound impact on the lives of ordinary people. Just as we expect those involved with nuclear power to act in a responsible fashion, so must we give serious thought to the proper use of our talents and abilities.

Our society provides many outlets for those who have needs which are objectionable to the average citizen. If you feel an outlet for adult material is needed in

the computer industry, I suggest you set up a sister publication (a la *Softline*) with a separate subscription fee, and let the marketplace decide.

Those who cry loudest about freedom of speech are inevitably members of unpopular minorities who wish to impose their views upon the majority. In times past the question of minority rights has been settled by civil war, legal controversy, and social upheaval—I hope that we can resolve the current situation without recourse to any of the above.
Barry Jedrick, Lutherville, MD

I feel that I must object to the recent advertisements in *Softalk* which are promoting the sale of erotic exploration, hardcore software, sexplosion and *The Dirty Book*, *Softporn*, and other things found therein. By the very words used these companies seem to be advertising and selling hardcore and pornographic software, in my personal opinion.

Since *Softalk* is freely distributed to all owners of Apple computers in the USA, and has a wide distribution elsewhere, I infer that such ads, and perhaps even some of this software, is getting into the hands of children. On the second page before a sexplosion and *Dirty Book* ad is an ad entitled "Prepare them (children) for their future" (much better stuff) . . . but how ironical! I raise the question whether or not you would like this stuff to fall into our children's hands. Young teens have enough problems these days in struggling to live by a sound value system without being exploited by the quick buck and pornographic salespersons. And I hope this is not the case. I presume that this magazine is not competing (and does not need to) with *Playboy*. Moreover I fear that a next step will be the inclusion of software such as this, referred to above, in electronic games centers if such people as are advertising in your January issue (and some of your previous issues) have their way. Is this what we want? Is this what you want? Your magazine can do just as well without this stuff.

Speaking personally, but as a computer professional and as a member of the IEEE Computer Society Educational Activities Board and formerly chairperson of the Computer Society Education Committee, I feel that it is my responsibility to take a leadership role and make comments regarding the use of computers and software in a way that goes against the grain of many value systems (such as my own, for example) in our country.

I realize that you have the legal freedom to at present publish this somewhat questionable stuff, but I am appealing to you as publisher and as one who must care about what gets into the hands of our children to refrain. By doing so I do not think you will lose the patrons you want. Let us work together to insure a precedent of quality and value in the kinds of programs that our children use.

Feel free to circulate an exact copy of this letter if you wish.
David C. Rine, Macomb, IL

The Dirty Book *is a catalog, a takeoff on* The Book; Softporn *is a well-done text adventure that is* not *pornographic;* Street Life, *which we have* not *seen, we are assured is similar to* Softporn*; and the rest are compilations of suggestions that would have no meaning to anyone who was not already familiar, first-hand, with what the suggestions were intended to enhance.* Interlude, *incidentally, is being used by several psychiatrists in various parts of the country as an adjunct to marital counseling.*

There is pornographic software; its advertising has long since been rejected by Softalk.

As the media service organization for the University of Michigan's School of Education and Library Science, we house the computer resource lab which includes many current computer periodicals.

When we received your magazine for the first time last week, we looked forward to having another valuable resource for software location and acquisition. Instead we found print that was difficult to read, reviews that were rather shallow, and so many ads that they overwhelmed the substantial and sometimes interesting content of your articles.

Even more concern to us was your acceptance of ads utilizing a woman as an enticement to sell a product totally unrelated to her presence.

An example would be the Howard Software Service's ad selling their creative financing program. We found this ad to be sexist, degrading to the increasing number of women in the computer field, and to a sizable number of men for whom these ads are insulting and ineffective. This tactic of advertising is counterproductive and totally unnecessary to the marketing of this product.

We understand the need to attract revenue for your publishing venture, but strongly urge you to develop a selection policy that does not allow advertisements which insult and degrade female persons.
Margaret T. Schmidt, Media Consultant
Beth Miller, Media Associate
Jeff Werner, Media Consultant
Elaine K. Didier, Director I.S.S. Instructional Strategy Services
Ed Saunders, Media Associate
Cynthia Cowens, Media Associate, Ann Arbor, MI

I wanted to say a couple of words about all the *hubbub* on the *"filthy"* ads in your magazine. I really don't understand where these people are coming from (the eighteenth century maybe). It seems quite simple that if the people that are offended by these ads feel so about them they should just cut those ads out or black them out with Marks-a-lot. Being a father myself, I do wish to protect the innocence of my child; however, the first place to start is at school since most of the kids from fifth grade on know more than you'd expect. Protecting your kids from ads of the above-mentioned variety doesn't stop the kids from spreading the word around. The parent just has to explain the things that go on in our society, not hide the fact that it happens.
James McGuire, Omaha, NE

For the Love of Mike

OK . . . now you've done it! With the War and Peace letters opening the Open Discussion feature in the February *Softalk*, you've incurred *my* wrath, too. I, too, have children who read *Softalk*; my daughters who are eleven and thirteen both peruse your publication with varying interests. How dare you publish such filth as the *Softporn* and *The Dirty Book* ads? I decided to check the February issue to see just how much you have become purveyors of smut!

Inside front cover—naked thigh of rider exposed . . . shame! I won't mention the implication of the beast's exposed tongue nor the predominant color of the overall ad . . . we *all* know what red stands for! Page one shows a lot of skin and a nice figure revealed by tight jeans . . . double shame! Page two is acceptable, I suppose, if one is willing to overlook the name of David Hunter as the advertising coordinator; we need not wonder what he hunts. Page three is much too suggestive to be tolerated! Not only are the insects naked, many of the plants are as well. Look at those toadstools . . . such a low viewpoint affords the casual reader a direct view of the plant's reproductive organs . . . this deserves more than a "shame," gentlemen, this is blatantly foul, vile, disgusting filth! Page four is a beaut! We know the sexual overtones of Peaches, but you transgressed the tastes of good journalism when you printed this page of rot and perversion! The Lord only knows what *Pear* Software is! The blatant implication of kinkiness abounding here is intolerable. The second column shows two males touching one another . . . maybe in two places! I know how small a disk drive is . . . you can't fool me! I bet that maybe beneath that drive mechanism skin may be touching skin! Beyond the drive itself, think of the associated terminology! Where else have we heard the term *insert* used . . . where else in our society is one encouraged to "insert his floppy in the slot . . ."! By just the associated jargon alone, the whole subject of disk drives should be abolished from your publication.

Then . . .

Give me strength . . . I find it hard (that's dirty all by itself) even to write about this, in the third column you show a man and a machine! To what degradation have you slipped, gentlemen? I could

SOFTALK CLASSIFIED ADVERTISING

FREE DATA DISK—EVERY ORDER! Best service, low prices; no hidden credit or ship charges. Apple, CP/M, utility, game software. Videx, Hayes, Orange Micro, TG, Superscribe, General Manager, Applewriter Extended. Visa, M/C, check, MO to Gold Disk, Box 102, Glen Arm, MD 21057. (301) 592-7515. Free catalog.

EXCELLENT GRAPHICS EDITOR—A professional graphics development system including hi-res text, circles, boxes, dazzling color blends, screen inversion and much more. Comparable to systems costing over $60. Full manual included. Send $29.95 to: Aristotle Software, 1516 N. State #16A, Chicago, IL 60610.

PEACHY WRITER

New! A simple, easy to use text editor at a price you can afford! $24.95

HAPPY FACE

4 hi-res games for children: *Riddles, Famous Sayings, Nursery Rhymes, Spelling.* 48K $15.00

COLLEGE PHYSICS (11 disks, 75 programs) $200; AQUARIUM $25; DINOSAURS $15; CHRISTIAN EDUCATION $15. Write for a free catalog. Visa M/C. Cross Educational Software. Box 1536, Ruston LA 71270 (318) 255-8921.

EPSON MX80 MX100 OWNERS! Save time and effort. Set any text processing printer option quickly and easily with this turnkey software. Save settings for future use. *MONEY BACK GUARANTEE.* For APPLE II+ from DEFINITIVE COMPUTER SOFTWARE, 362 Concord St., El Segundo, CA 90245. Send $15.95

Play **GO** on your Apple! Enjoy hi-res board (13x13) display and text descriptions of the reasoning behind the computer's moves ($15). Also enjoy **ART OPUS** ($10), a beautiful hi-res art program. Save programming time with hi-res **GRAFTOOLS** unit ($10)Apple II with Pascal required for all programs. Enclose $6 per order for disk and shipping. MARK WATSON ASSOCIATES, 9969 Via Daroca, San Diego CA 92129 (714) 271-8979.

COMPUTER GRAPHICS COURSE—40 HR hands-on course intended for companies that would like to create colorful graphics on the Apple II. Applications include in-house training, presentations, slide shows, and interactive education. Course held in Long Beach, CA, the first and third weeks of each month. (4 students per class max.) For information contact:

Mark Wauben
ISLAND SERVICES INC.
920 E. Wardlow Rd.
Long Beach, CA 90807
(213) 595-0258

MESSAGE-MAKING SOFTWARE

Enter text at Apple keyboard and watch large, colorful, dynamic letters appear on screen. Programs range from $20 to $50.

Complete Apple literature-index through Dec 80, on disk!

Enter keyword, and print all finds! Over 1200 references, from 64 periodicals! ($60)
Conn. Information Systems, 218 Huntington Rd., Bridgeport, CT 06608, (203) 579-0472

VC-MANAGER extends **VISICALC**
Now you can add, subtract, divide *VisiCalc* files to generate Variance reports, Statements of Changes, Company totals from segments or products etc. Up to 15 files can be combined, DIF or Logic. $65. Micro Decision Systems. P.O. Box 1392, Pittsburgh, PA 15219. Tel (412) 276-2387

PROFESSIONAL STATISTICAL ANALYSIS

HSD ANOVA: 8 factors, any combination of between and within.
HSD STATS: 3 t-tests, Chi-Square, frequency distribution, data transform, graphics, etc.
HSD REGRESS: multiple regression, descriptive stats, graphics.
HUMAN SYSTEMS DYNAMICS, 9249 Reseda Blvd., Suite 107, Northridge, CA 91324, (213) 993-8536

LIBRARIAN—Filing and keyword retrieval system for books, articles, excerpts, etc. Produces automatic bibliographies. Initial data base included. $69.95.

CONTOUR MAPPING—High quality contouring system for printer, plotter, or CRT. Data and grid transformations, trend surfaces, residuals. Data and file manager.

GEOLOGICAL and GEOGRAPHICAL software for Apple and other systems.
GEOSYSTEMS, Inc., 802 E. Grand River, Williamston, MI 48895 (517) 655-3726

A MOST UNUSUAL CATALOG

The Yankee Peddler offers over 1000 reasonably priced software and hardware items for you and your computer. We rep. all the majors: On Line, Datamost, Typrinter 221, Epson, OKI, Budgeco, SDS, Beagle Bros., etc. Send for your free catalog today.
YANKEE PEDDLER, 10347 Kurt St., Lakeview Terrace, CA 91342.

APRIL SOFTWARE SPECIALS

Arcade Machine	*Atlantis*
MasterType	*County Fair*
TASC Compiler	*Apple Panic*

Call or write for information about these specials.

BYTES & PIECES
Box 525 Dept 4S
East Setauket, NY 11733
(516) 751-2535
Ask about our free price list 604

COMING SOON!

The new price/performance leader in personal finance accounting. Up to 80 budget categories and 20 separate accounts. Present net worth evaluation. Requires 48K Apple II, Applesoft in ROM & DOS 3.3. Only $19.95, shipping prepaid. Demo disk $4. **COMPUTECHNIQUES,** P.O. Box 245, Fairfield, OH 45014.

PAYMENTS TRACKING SYS.

Reduced to **$39.**
EOM & YTD records—taxes, etc. Ten Programs, eight major reports. Requires Apple II 48K two disks and printer. Send SASE (legal size) for sample to: Payments Tracking System, Box 1984, Upland, CA 91786.

PROTECT-O-DISK™

Software protection with back-up for BASIC, PASCAL, and Assembly. New licensing agreement for publishers only. Diversified Software Research Inc., 5848 Crampton Ct., Rockford, IL 61111. (815) 877-1343.

Guaranteed lowest prices on
All major brands of micro-software
90 day guarantee—when you're happy, we're happy! Gigantic inventory of old and new releases. Overnight shipment by UPS-no per item handling fee. We credit orders $1.00 when you call out of 213.

SOFTWARE GALORE
For free catalog/orders call 213-827-1851 (8a-5p only). 800-423-6326 (Direct National)

CEEMAC

The Visual Composition System
(Language of Fire Organ)
Vagabondo Enterprises
$40 1300 E. Algonquin #3g
Schaumburg, IL 60195
Please tell your friends about this ad

RESUME RIGHTER™

Creates EFFECTIVE resume . . .
Gets you more INTERVIEWS . . .
Full screen PROMPTING . . .
SOFT SEVENTY©.
70 columns on your screen (no hardware req)
. . .
FREE *Hints & Tips of Resume Writing*
by Catherine Winston
$24.95 pp 48K Disc Applesoft (printer optional)
Acro-Matic, 256 SW 5th St. Boca Raton, FL 33432

Small investor seeks to offer reduced rents and financial support to growing computer oriented companies in exchange for stock ownership and/or joint venture.

Property Investment Feasibility
21021 Devonshire, Suite 103
Chatsworth, CA 91311
Rob Matlosz

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 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 53 characters per line. Spaces between words are counted as one character. Heads will hold roughly 25 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
11021 Magnolia Boulevard
North Hollywood, California 91601
Attention: David Hunter
213-980-5074

mention the garish exploitation of sex on page five (Freud's teachings about phallic symbols aren't beyond you, are they?). The depiction of the tank's cannon ejaculating is untainted smut! (I like those oxymorons!) I could complain about the bawdy language in page eight "tools," the nakedness of the alien on page thirteen, and so on; I can't bring myself to look at such a welter of lascivious vilification! Page forty-four's shot of a beaver is shocking, but the nakedness of page sixty-five is even more offensive! The frontal nudity (both topless and bottomless) of those computers not properly attired is appalling!

Finally, I have to admit that this little foray has taught me something. We all know that the apple caused man's fall from God's grace . . . the apple is widely recognized as the fruit of the *Forbidden Tree of Knowledge!* Touching an apple is much the equivalent of fondling . . . um . . . things which are "dirty"! I now realize that I must sell my computer system. Not only has it helped me learn, but, may God forgive me, it has been something which I have embraced in moments of sickness and in health, in joy and in sorrow; yet it is the epitome of all evil!

In a more serious vein, just in case it isn't apparent, I write to point out what you already know; you'll be damned if you do (accept and publish such ads as have caused this seeming controversy) and you'll be damned if you don't! For what it's worth, the models and poses used in the Howardsoft ads have, without exception, aroused me far more than those used in the *Softporn* or *The Dirty Book* ads. (Maybe this is just my own kinkiness coming in to play. . . .) When you censor the look in these ladies' eyes, you had better begin publishing the Holy Bible . . . but don't use any graphics! To avoid controversy, don't use any words, either. The suggestivity of words and word combinations is just too rich to broach!

If you want a vote from one who intends to stick with you when you begin charging for your publication, I vote that you continue to accept ads from all concerns who run their businesses legitimately. Stick to your guns, guys and gals, you are doing a great job!

Mike Huston, Burdett, NY

Time Out of Joint

As the producers of the Superclock II, we

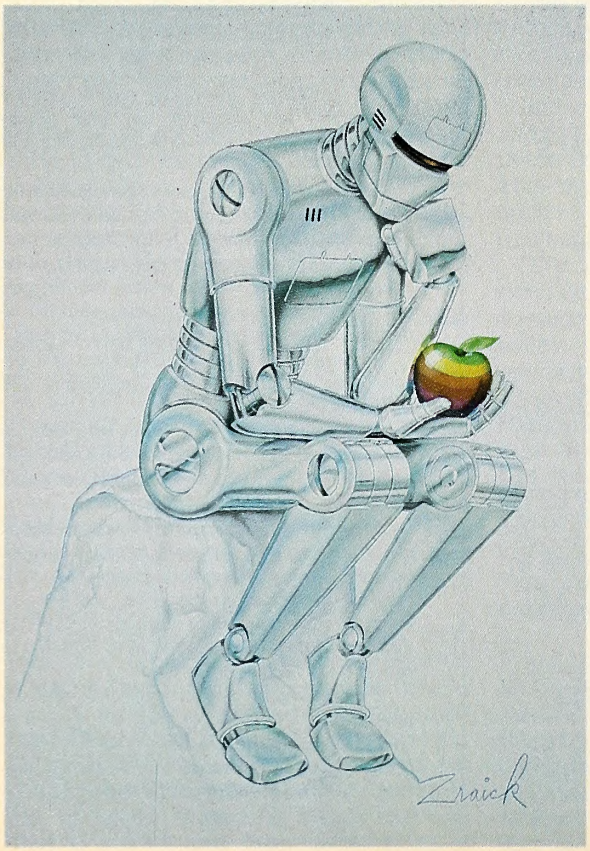
GOTO 68

A Delicious Apple

I thought you might be interested in a recent birthday cake which consisted of an almost life-size Apple II replica complete with Monitor. My wife, Luana, had it made for my fortieth birthday, complete with simulated wake for my "passing." The cake was delicious as well as a big hit with those in attendance.

The theme was picked due to my having an Apple II here at school as well as one at home.

Ron Zellner, Greeley, CO



contemplating a byte

Robots are here and they are changing the world we live in. From bulky industrial welders to fantastically complex planetary probes, robots are sure to make our lives a little easier. Robots will get much more sophisticated in the decades to come; by the next century they may be our model citizens.

But will robots be immune from the human weaknesses that usually attend a high level of intelligence? On the cover of our August 1981 issue we fantasized what a humanoid robot may look like in the future. We also gave this highly developed mechanical man the hardest task we could devise—contemplating an object and its significance.

Will robots ever be able to sit and think about something that is not directly related to performing a task?

Softalk can't answer that question for you, but we can help you contemplate the unknown future in a special way. We commissioned graphics artist Robert Zraick to do August's cover with a poster in mind. The robot contemplating a bite is evocative both of Rodin's *The Thinker* and the Genesis passage on the Garden of Eden . . . not to mention the possible significance to our favorite technological fruit.

The artist and *Softalk* are sharing in the profits from the poster. *Softalk* will distribute its proceeds to individuals developing Apple tools to help the handicapped. *Softalk* guarantees 100 percent distribution of its monies.

In addition to the posters, which are sold at $6.00 (plus $1.50 to cover shipping and handling), some of the two hundred artist's proofs, signed by Robert Zraick, are still available at $75 each.

The size of the poster is 24 inches by 34 inches. The artist's proofs are hand-numbered and hand-signed, and each is accompanied by a certificate giving its number and guaranteeing that only 200 are being distributed.

Robert Zraick's art will grace any computer room, and your purchase will help others become more self-sufficient.

Orders may be sent to:

SOFTALK

Softalk Poster
11021 Magnolia Boulevard
North Hollywood, California 91601

Why does EDU-WARE bother

to delve into the educational needs of today's society?

Because at Edu-Ware, we are scientists on the frontiers of instruction. Our concern is with finding the best ways to communicate the knowledge needed and the tools to use it.

Edu-Ware is not interested in creating solutions to chase nonexistent problems, but in applying its technology to core curriculum areas for which demand exists today.

In 1981, Edu-Ware established 97 evaluation sites at schools and universities across the country, making a major commitment to research of CAI (Computer Assisted Instruction) and other instructional technologies. The project is the first step in establishing a national board, which will review standards of performance and quality for instructional systems in the United States and abroad. This project helps Edu-Ware to find a better way.*

Edu-Ware's Instructional Designers apply their tools to the development of learning software systems, establishing an ease and effectiveness with which the microcomputer can become a part of the classroom and enhancing the learner's performance.

The professional programming staff at Edu-Ware takes CAI beyond simple text-page displays and converted school book formats. Working to a script prepared by an Instructional Designer, the programmers construct learning software in which parents, educators, and learners can have confidence.

At Edu-Ware, Instructional Designers work together to develop learning software strategies.

Ultimately, the Science of Edu-Ware is measured in its results, in what Edu-Ware offers parents, educators, and learners: a better way.

*For details about the Test Site Project, contact Edu-Ware.

School purchase orders welcome.

EDU WARE®

The Science of Learning™

Edu-Ware Services, Inc. • P.O. Box 22222 • Agoura, CA 91301 • (213) 706-0661

Put your business forms on an Apple II™

An innovative software system that speeds the processing and analysis of common business forms on a personal computer. VersaForm applications include:

- **Invoicing**
- **Purchase orders**
- **Client billing**
- **Mailing lists**
- **Order purchasing**
- **Sales analysis**
- **Personnel records**
- **Inventory records**

Hard Disk Versions Available

VersaForm™

Business Form Processor

Apple III VERSIONS NOW AVAILABLE

Applied Software Technology

15985 Greenwood Rd.
Monte Sereno, California 95030
(408) 395-1541

All About Applesoft

by Doug Carlston

Last month we illustrated the column with a couple of short programs that incorporated new programming techniques as well as new vocabulary. Here is one of them:

```
10    HGR:X = PEEK (49234)
20  X = 140:Y = 96
30    HCOLOR= 3
40    HPLOT X,Y
50  X = X - 5 + RND(1) * 10
60  Y = Y - 5 + RND(1) * 10
70    IF X >=0 AND X <=279 and Y >=0 AND Y <=191 THEN 130
80    IF X < 0 THEN X = X + 279
90    IF X > 279 THEN X = X - 279
100     IF Y < 0 THEN Y = Y + 191
110     IF Y > 191 THEN Y = Y - 191
120     GOTO 40
130     HPLOT TO X,Y
140     GOTO 50
```

This time we want to focus on programming style rather than on vocabulary. However, there are a few new words you'll need to know, so here they are:

REM	PDL(X)	READ
DATA	INPUT	GET
CHR$(X)		

We want to talk first about documenting a program so that your code will be useful to you later on when those flashes of intuition that brought it about in the first place have faded into the dusty recesses of your brain.

Enter the short listing from last month. To make this program as easy to understand a year from now as it is at this moment, it would be useful if we were able to comment on the purpose of each line or group of lines and to save our remarks along with the program. The *remark* statement is designed for this purpose. When your Apple comes across the three letters REM, it understands that the rest of the program line is pure comment, included for the benefit of humans rather than of computers. So let's add the following lines to the listing:

```
5     REM *** TURN ON HI-RES GRAPHICS
15    REM *** SET INITIAL VARIABLE VALUES
35    REM *** PLOT STARTING POINT FOR A LINE
45    REM *** SELECT NEW VALUES FOR X AND Y
65    REM *** TEST & MODIFY X,Y VALUES IF NOT ON SCREEN
115   REM *** IF OFF SCREEN, JUMP TO 40 TO START NEW LINE
125   REM *** OTHERWISE CONTINUE CURRENT LINE
```

Now list the program. The remarks set off the program segments so we can focus on particular ones more readily.

Let's take a longer look at lines 50 and 60. These lines select new values for X and Y. X started at 140, which is halfway across the screen, and Y started at 96, which is halfway down (see line 20). To make the line move randomly in any direction, X and Y need to change by an uncertain amount in *either* direction. In other words, we want to add or subtract numbers at random from both X and Y. If we choose a random number between zero and ten and add it to X, X will always get larger. However, if we subtract 5 from X and *then* add a random number between 0 and 10 to X, X may get larger or smaller depending on whether or not the random number chosen is larger than 5.

To illustrate what would happen if we didn't adjust X and Y downward before adding the random values, try changing lines 50 and 60 to read like this:

```
50  X = X + RND(1) * 10
60  Y = Y + RND(1) * 10
```

We can modify this section of the program in many ways to get different results. Try subtracting 4 from X instead of 5 and see what happens. Or even better, let's attach paddles to the computer so we can control the computer's walk.

To see how paddles work, temporarily add a new line 1 to the program. It isn't necessary to erase the rest of the program by typing *new*—that would just make for a lot of additional work. Just type in the following line:

```
1  PRINT PDL(0), PDL(1): GOTO 1
```

Make sure that your paddles are properly plugged in and then type *run*. Two columns of numbers should appear. The one on the left gives the reading of paddle zero and the one on the right gives the paddle one reading. As you turn the dials you should see the numbers change. (This will also give you a chance to experiment with one of the bugs in the Apple, by the way. If you turn paddle zero down toward zero, high readings of paddle one can go strange. Entering a short time-delay between the two paddle reads usually corrects the problem.) As you see, the numbers range from 0 to 255.

Now break into the program with control-C and remove line 1. Next let's type replacement lines for 50 and 60 that use the paddle readings rather than random numbers to figure out where to draw:

```
50  X = PDL(0)
60  Y = PDL(1)
```

When you run this version, you'll find that you have a version of that old toy Etch-a-Sketch. Not a very satisfactory one, however. Sometimes the program draws smooth lines, but lots of times it jumps around and seems to draw at random. What can we do about this?

As we noted, the paddles return values between 0 and 255. This means that we are setting our X and Y values to this range. That's just about right for the X, but it really won't do for the Y. Whenever Y is set to a value greater than 192, the program tries to plot a point off the bottom edge of the screen. The trap on line 110 catches this and attempts to correct the problem. However, we can avoid the whole problem by scaling our paddle responses so that the 0-255 range of the paddles is converted to 0-279 for X and 0-191 for Y. The following lines will scale the responses correctly:

```
50  X = PDL(0) * 279/255
60  Y = PDL(1) * 191/255
```

As a test, try to figure out in your head the values of X and Y when the paddles are set to their low and high positions.

Go ahead and save the etch-a-sketch program if you like and then restore the program to its original and highly random condition so we can take a look at the next section. When we were getting the values of X and Y from the random number generator, we needed to check and make sure that the values didn't stray beyond the acceptable ranges. Line 70 tested both X and Y for high and low acceptable values; if both variables passed their tests, control was passed to line 130, which plotted a line and returned for new values. If somebody flunked the test the intervening lines figure out why and remedy the situation.

There are a lot of ways to keep the values within the acceptable range. The simplest way is a line like this:

```
70  IF X < 0 THEN X = 0
```

In other words, if the random number generator tried to make X too small, we could just bring it up to the minimum acceptable value and leave it there until it headed back into the positive range. What we actually did, however, is a little more interesting. If X went below zero, we added 279 to it. This had the effect of moving it from the far left side of the screen to the far right side, a feature that is often called *wrap-around*. The other three test lines provided wrap-around on the other three sides of the rectangle. Naturally, if we are going to shift either X or Y to the opposite side of the screen, we had better start plotting an entirely new line (that's why we jump to line 40 instead of 130)—if we didn't we'd get a connecting line drawn all the way across the screen from the previous X,Y value.

Page Flipping. One of the questions that comes up almost as soon as people start using graphics is how to mix graphics and text on the same screen. Unfortunately, the answer is nowhere near as easy as the question. The easiest way to mix text and graphics is, of course, to do your graphics at the top of the page and put all of your text on the four lines at the bottom.

Some enterprising souls have inquired whether or not one couldn't write the text on the text page, draw the graphics on one of the graphics pages, and then flip very rapidly between them so that the eye mixed them up. You ought to be able to write a program by now that does exactly that, so please give it a try on your own before examining the following example:

```
10  HOME: VTAB 14: HTAB 10: PRINT "SOMETHING SPECIAL"
20  HGR : HCOLOR= 3: HPLOT 55, 100 TO 190, 100 TO 190, 120 TO 55, 120
    TO 55, 100
30  POKE 49233, 0: POKE 49232, 0: GOTO 30
```

Easy enough to see why this technique isn't very popular, isn't it? There are a couple of other approaches that actually do work. Both involve drawing characters on one of the hi-res screens; with perseverance, we will soon come to know enough to do this.

Input Routines. Now it's time to turn away from the Apple's rainbow splendors and learn something practical once again. The computer provides many ways of manipulating and displaying data, but first you have to get your data into the machine. There are lots of ways to do this. You can make your data part of the computer program you are writing (which is fine as long as it won't ever change), you can key it in at the time you run your program, or you can read it in from data files that you have previously written and saved onto tape or floppy disk.

The simplest way to see that your data gets into the machine is to build that data program. Take a look at the following example:

```
10  READ NAME$,NUMBER
20  DATA MICHAEL SMITH,4992
30  HOME : PRINT "DEAR "NAMES":"
40  PRINT: PRINT " YOUR NUMBER IS "NUMBER"."
```

The *read* statement causes the program to look for a line that starts with the word *data*. It then tries to assign the first piece of data in the *data* statement to the first variable in the *read* statement. Please note that it is very important that the variable "type" match the data type. Let's try two experiments. First, replace line 20 with the following line:

```
20  DATA 4992,4992
```

The program will run perfectly. The string variable NAME$ doesn't care whether you try to feed it alphabetical or numerical data; it can handle both. Now try running the program with the following line 20:

```
20  DATA MICHAEL,SMITH
```

You will get a syntax error, since the variable *number* can only accept numerical data (remember, string variables *always* end in a dollar sign). Note that individual data items are separated by commas. This could be a nuisance if your data entry includes a comma. Here's how to get around the problem:

```
20  DATA "JONES, DEACON",4992
```

The second way to get your data into your program is to key it in from the keyboard at the time you run your program. These keyboard input routines come in several shapes and flavors, each with its advantages and disadvantages. The easiest to use is *input*.

```
10  HOME: PRINT "WHAT IS YOUR NAME";: INPUT NAME$
20  INPUT "AND YOUR NUMBER";NUMBER
```

These are two acceptable ways of using the *input* statement. In line 10, the *input* command will cause a question mark to be printed and will then accept your response. In line 20, the *input* command will print your message, enclosed in quotes, instead of its usual question mark. Note that in the second case the message must be separated from the variable name by a semicolon. Either form can accept more than one variable at a time. An acceptable alternative to the above is:

```
10  HOME: INPUT "PLEASE GIVE NAME AND NUMBER, SEPARATED BY A
    COMMA: ";NAME$,NUMBER
```

One nice thing about the *input* command is that if you accidentally type a word where you should have typed a number, it does not bomb your program with that awful syntax error message. Instead, it merely types a question mark, followed by the word *reenter*. If you give it fewer responses than it expects, it will prompt you with a question mark and patiently wait for you to finish giving it the rest of your data. If you give it three pieces of data when it only expects two, it will warn you that you are off the track by printing *extra ignored* on the screen. Not very polite, perhaps, but right to the point.

Input does have its limitations, however. You cannot prevent it from printing your response to the screen, you cannot do other processing while you wait for the keyboard response (such as updating a clock), and you must always terminate your entry by pressing the return key, an extra keystroke that might better be avoided at times.

Applesoft has a second command that addresses two of these problems. *Get* waits until a single keystroke has been made, at which time the program immediately continues on to the next instruction. See this example:

```
10    HOME: VTAB 10: HTAB 10: PRINT"CHOOSE ONE:"
20    PRINT: PRINTTAB(14)"A) LIFE"
30    PRINTTAB(14)"B) DEATH"
40    PRINTTAB(14)"C) DITHER"
50    PRINT: GETA$
60  IFA$ = "A" THEN PRINT"SO BE IT.":STOP
70  IFA$ = "B" THEN PRINT"PLEASE RECONSIDER.":STOP
80  IFA$ = "C" THEN PRINT"TOUGH CHOICE, ISN'T IT?":GOTO50
90    PRINT: PRINT"PLEASE READ THE INSTRUCTIONS";PRINT" & TRY
      AGAIN."
```

```
100  GOTO 50
```

There are a couple of problems with the *get* command. First, it is a pain in the neck to use if the data you are trying to enter is more than one character long. Second, you can't use control-C to regain control of your computer anymore (try it). Third, it still doesn't allow the computer to wander off and do other tasks while waiting for the slow human who is punching the keyboard to come up with another brilliant selection.

Taking care of the second objection is pretty trivial. Add the following line to the program:

```
85  IFA$ = CHR$(3) THEN PRINT"TRYING TO STOP ME, HUH?":
    GOTO 50
```

CHR$(x) stands for *character x*. Every character you can type from the Apple keyboard (and several that you can't) has a number, called an ASCII number. These are listed in Appendix K of your Applesoft book. **CHR$(3)** is exactly the same as control-C.

There's a third input routine in your Apple. This one may seem pretty esoteric, but it's absolutely the only way to go if you are trying to enter data real-time. Let's list it first and get to the gory details after:

```
5    REM INITIALIZE VARIABLES
6    HOME: HR$="AM": DAY = 1: HOUR = 9: POKE 49168,0
7    REM ***************************
10   REM INPUT ROUTINE
20   KEY = PEEK ( 49152): IF KEY < 128 THEN MINUTE = MINUTE +1:
     GOSUB 100: GOTO 20
30   LTER$ = CHR$(KEY - 128): POKE 49168,0: PRINT LTER$: GOTO 20
40   REM ***************************
100  REM CLOCK ROUTINE
110  IF MINUTE >= 59.9 THEN HOUR = HOUR +1: MINUTE = 0: IF HOUR
     >= 12 THEN HR$ = "PM":IF HOUR >= 24 THEN DAY = DAY + 1: HR$
     = "AM": IF HOUR > 24 THEN HOUR = 1: DAY = DAY + 1
120  AMHOUR = HOUR - 12 * (HOUR > 12): VTAB 1: HTAB 19:
     PRINT " "AMHOUR";"MINUTE" "HR$" DAY #"DAY;RETURN
```

Please ignore the clock routine for now, unless you delight in working your way through convoluted logic. Let's focus instead on the input routine in lines 20 and 30.

Address 49152 in the Apple always contains the ASCII code of the last key pressed *plus* 128. You can reset this address to zero by *poking* a different address, 49168. We did this in the first line of our program. In line 20 we first *peeked* 49152 to see if any key had been pressed. As long as the value of this address is less than 128, we know that no key has been pressed. Therefore, we can increment our timer (called *minute*) and jump to a *subroutine* at line 100 that displays a clock in the upper right corner of the screen. We'll come back to subroutines next month. For the moment, suffice it to say that *gosub* is similar to the *goto* statement in that it (temporarily) transfers program control to a different line number, then *returns* from whence it came.

If any key is pressed, the value stored in 49152 will immediately change to the ASCII value of whatever was pressed, plus 128. We have stored the value of this address in the variable we call *key*, so *key* will now have some value greater than 128 the moment the keyboard is touched.

Line 30 takes the value of *key* and subtracts that extra 128 from it. It then sets the string variable LTER$ (short for LETTER$) equal to the character described by the ASCII value (*key*), clears the address 49152 to zero, prints LTER$, and returns to line 20 to wait for another character.

That's a lot of new concepts in a very short space, and some of them don't really seem to make a lot of sense. Why, for instance, do you *poke* one address in order to clear a completely different one to zero? Beats me. The important thing to remember is that even though you own your Apple, your Apple still makes the rules. A sobering thought to end this month's discussion. ❑■

THE NEXT GENERATION OF COMPUTER GAMES IS HERE

FROM THE MAKERS OF THE FIRST ANIMATED ADVENTURE GAMES

Synergistic Software

- **APVENTURE TO ATLANTIS** is a new fast action apventure game, by Robert Clardy, continuing the Odyssey Apventure series. The forces of sorcery are gathering to repulse the marauding scientific Atlantean Kingdom. The epic battle can end either in the enslavement of the world or the absolute destruction of Atlantis.
- Computer adventuring has been good, Synergistic Software has just made it better. Can you imagine the most exciting adventure game ever composed with more color and sound, highly detailed animation, arcade like action, true role playing, greater challenge, unending variety and endless enjoyment? That's what the Next Generation of computer games is all about. This professionally designed adventure was specifically written to take full advantage of all the Apple's capabilities. Requires careful planning and strategy, quick thinking and reflexes, as well as a 48K Apple II* with DOS 3.3 and paddles.

Only $40.00

*Apple II is a trademark of Apple Computer Inc.

From

Synergistic Software

5221 120TH AVE. S.E.
BELLEVUE, WA 98006
(206) 226-3216

TRADE TALK

☐ **Tom Clarkson,** in association with **Burt Perry, Russell Sprunger, Tim Stevenson,** and **Dave Sharon,** has formed **Graphic Software Systems** (Wilsonville, OR). All five previously worked together at Tektronix. The new company's goal is to produce mainframe-quality graphics software for microcomputers—computer-independent, graphic device-independent, and Siggraph core standard. The first GSS product will be directed to the scientific/engineering community and will be followed by a line of business graphing products designed to run on all mini- and microcomputers.

☐ **Steven Dompier** and **Diane Ascher** have announced the formation of **Island Graphics** (Bethel Island, CA). The company will develop, publish, and market a full line of graphics oriented software for the personal computer user, to be previewed at the West Coast Computer Faire. Ascher has worked for several hardware and software manufacturers; Dompier is the former vice president of research and development for Processor Technology.

☐ The first European manufacturing operation for **Shugart Associates** (Sunnyvale, CA) will open this year in Lille, France. Company president **James S. Campbell** announced that the firm will produce its *SA801* single-sided eight-inch floppy disk drive products at the 250,000-square-foot facility this spring, adding additional products in September. The company will share space at the Lille plant, located near the Belgian border, with Diablo Systems and Xerox's office products division.

☐ **William Lohse** has been appointed vice president of sales and marketing for **Information Unlimited Software** (Berkeley, CA). Previously director of sales for both Micropro and IMSAI, Lohse's primary responsibility will be to head up the sales and marketing efforts of IUS's second generation of software products.

☐ **Diane Walkowiak** has been promoted to executive vicepresident of **Small Business Computer Systems** (Lincoln, NB). She will be overseeing the growth of the accounting software company, which recently doubled its office space and expanded its software line.

☐ **Gary Kevorkian,** formerly of On-Line and Programma, now heads the technical support staff at **Continental Software** (Rancho Palos Verdes, CA) as the company's customer support manager.

☐ In order to avoid an extended courtroom battle over a threatened lawsuit from Unilogic, a Pennsylvania manufacturer of a mainframe utility called *Scribe*, **On-Line** (Coarsegold, CA) has changed the name of *Superscribe II;* henceforth to be known as *Screen Writer II.* An updated version has been released under the new name.

☐ **Best Products** (Richmond, VA), the nation's largest catalog showroom merchandiser, has made its initial venture into the personal computer retail business, opening the **Data Base,** a full service personal computer store, in Richmond. Best operates one hundred showrooms across the country.

☐ **Universal Data Systems** (Huntsville, AL) has announced the formation of a consumer products distribution network—High Technology of St. Louis, Missouri; Microware of Portland, Oregon; Waybern Corporation of Garden Grove, California; and Micro Distributors of Rockville, Maryland—authorized to sell UDS low and medium speed modems to the consumer and small business computer market.

☐ **Gary Harpst,** president of **TLB Associates** (Findlay, OH), has announced a $750,000 distribution agreement with Computech Group of Philadelphia to distribute TLB's *Solomon Series* software in the northeastern U.S. While the company is setting up agreements with regional distributors to cover the other major geographical market areas, qualified dealers may purchase Solomon packages directly from TLB.

☐ **Corvus Systems** (San Jose, CA) has cut the suggested retail price of its Winchester disk systems 15 percent. Marketing vice president **Joe Hughes** cited the company's "high volume and improved product quality" as the factors allowing the price reduction. Corvus currently has more than a 70 percent share of the market for 5¼-inch and 8-inch Winchester disk systems and a 43 percent share of the total market for microcomputer local area networks.

Douglas Broyles, former president of Onyx Systems, has been appointed to fill the company's new position of vice president and general manager of special products. He will be working with the research and development division and corporate planning department.

☐ **Advanced Operating Systems** (Michigan City, IN) has established a national sales network of thirty-three authorized representatives. They will be responsible for marketing the company's software through distributor and retail sales outlets.

☐ **Business Development Partners**

(Austin, TX), a seed-capital partnership created in 1981 by Brentwood Associates, Hambrecht & Quist, Northwest Growth Fund, and Rothschild Incorporated, has made an initial investment of $500,000 in **Micro Peripherals Incorporated** of Salt Lake City, Utah. According to MPI president Ernest E. Campbell, the investment will permit MPI, manufacturers of 40, 80, and 132-column dot-matrix printers, to "expand manufacturing and sales operations immediately, while working with BDP and its limited partners to complete major financing later this year."

☐ **LPA Tech Corporation,** software supplier to the insurance industry, is now trading as **Vernon Tech Corporation** in order to establish closer identification with their affiliate, Vernon Publishing Service.

☐ **Microsoft** (Bellevue, WA) has delivered its 30,000th *SoftCard* to Apple Computer, eighteen months after the plug-in processor's introduction. "The impact has been tremendous," said Microsoft vice president **Vern Raburn**. "The Apple II was not even designed for CP/M but thanks to *SoftCard*, Apple owners have become the largest user group of CP/M-based programs."

Moving to expand the company's financial and management base, Microsoft president **William H. Gates** confirmed a one million dollar venture capital deal with **Technology Venture Investors** (Menlo Park, CA), which specializes in high technology firms. "The money's in the bank," said Gates, "where it will stay until we need it for special projects. We can move quickly if we see a good opportunity." The deal also gives Microsoft the expertise of TVI's three senior partners: **James J. Bochnowski,** past president of Shugart Associates; **Burton J. McMurtry,** member of the board of NBI, FAFCO, KLA Instruments and Triad Systems; and **David S. Marquardt,** a current board member of Plexis Computers, Seagate Technology, Adopt Technology, Archive Corporation, and Avera.

Vern Raburn (left), Microsoft's vice president of consumer products, presents Microsoft's 30,000th SoftCard to Apple Computer president Mike Markkula.

Mark L. Chamberlin has been appointed Basic interpreter group manager for Microsoft. He will be responsible for overseeing the development of the Microsoft Basic interpreter. Previously, he was an independent software consultant in Albuquerque, New Mexico, where he developed Microsoft's 6800/6809 Basic interpreter and co-authored the company's *Editor/Assemble-Plus.*

☐ In an effort to reduce prices and widen their product range, **Orange Micro** (Anaheim, CA) will conduct all mail order operations under the name **Red Baron Computer Products,** starting in mid-April. Said Orange Micro president Art Scotten, "Our pricing will be lowered to reflect volume distribution and new product lines to come; we need to promote with an image more aggressive than Orange Micro." Orange Micro will continue to specialize in printer retail sales.

☐ New Print (making) Technologies, a conference bringing together artists, industry specialists, educators, and graphics arts professionals to examine the applications of commercial printing and related industrial technologies, is being sponsored by the **World Print Council** (San Francisco, CA), May 14 to 16 at the Palace of Fine Arts in San Francisco. Presentations on computer graphics will be made by **Harold Cohen** of the University of California, San Diego, **Bill Ritchie** of the University of Washington, and **Pat Cole** of computer research and development at Lucasfilm Ltd. and member of the board of directors of SIGGRAPH (Special Interest Group on Graphics).

☐ **Erin** (Farmingdale, NY), distributor of microcomputers and software, has announced its commitment to marketing educational software and involving the educational community in the selection of program offerings. Company president **Frank Kelly** has appointed a committee of educators, under the direction of **Dr. Ludwig Braun,** director of the laboratory for personal computers in education at State University of New York at Stony Brook, to evaluate the products of software houses. Programs reviewed by classroom teachers within specific subject areas will be passed for review by the evaluation committee. If the program is accepted, the evaluation will be documented and marketed with the product as its seal of approval. Dr. Braun is a former director of the Huntington Computer Project, currently serving as advisor on computers in education to the National Science Foundation and the National Institute of Education.

A TEACHER FOR THE APPLE

Learn *How to Operate the Apple II Plus* on your own computer . . . at your own convenience . . . with this easy-to-use, audio mini-course.

Imagine having a personal tutor "talk you through" the practical operations of an Apple II Plus computer. In just 3 "hands on" training sessions of one or two hours each, you'll learn how to:

- Use each special key and essential command
- Load and run programs
- Save programs and data
- Copy programs and diskettes
- Protect programs
- Modify programs
- Use your computer as a super calculator
- And much more!

No Programming Required. You don't need to be a programmer to operate a computer, any more than you need to know how to design a car to drive one.

Our purpose is to put you in the driver's seat of your Apple computer so that you can confidently use existing programs . . . enter your own data . . . organize your files . . . and protect them. If you later decide to take up programming, you'll have the necessary background for doing so.

Teach Yourself. Train Others. *How to Operate the Apple II Plus* is designed for the first-time computer user. No technical knowledge is assumed. Perfect for training others: Staff . . . colleagues . . . clients . . . customers . . . students . . . family. Use it again and again.

Why Audio Instruction? Simple. It's human. It's interactive. It's multi-sensory. It's patient. It's inexpensive. And it works!

FlipTrack® Training Tapes deliver carefully sequenced instruction in a relaxed, conversational manner. You concentrate on the screen and keyboard, not a book. You learn at your own pace, trying every command and observing the results.

The FlipTrack® Difference. Best of all, our patented FlipTrack® cassette format can adjust to a variety of special interests and configurations. Optional instruction is available at a flip of the tape for users with color video, a printer, a second disk drive, or game controllers.

You save time and avoid confusion by learning just the procedures that relate to your system, or listening to just the topics that interest you. With FlipTrack® Training Tapes, you choose whether to follow a "full" track or a "fast" track every step of the way.

Examine *How to Operate the Apple II Plus* FREE for 15 days. Let FlipTrack® Training Tapes speak for themselves by dropping by your local Apple dealer for an on-the-spot demonstration. Or if that's not convenient, we'll be happy to send you the complete mini-course on a 15-day "right of return" basis. You'll receive all 3 FlipTrack® cassettes in a handy vinyl binder, along with an Operator's Guide, which will serve as your permanent reference to the procedures, terms and concepts of the course.

Your only prerequisite is an Apple II Plus computer with the System Master 3.3 diskette and a standard audio cassette player. No computer hookup is necessary for the cassette player.

Try a lesson or two yourself, or share them with others. If you are not delighted at how quickly and easily you master the operation of your Apple, simply return the program in its original condition for a full refund. No questions asked.

To order . . . See your local Apple dealer. Return the coupon below. Or call us toll free. Here's our 24-hour toll free number for credit card orders only:

800/227-1617, Ext. 439

In California, call 800/772-3545, Ext. 439.
You have nothing to lose, so act today.

YES! Please rush me *How to Operate the Apple II Plus.* I understand that if I'm not delighted, I may return the mini-course undamaged within 15 days for a full refund. No questions asked.

☐ My check is enclosed for $52.45 ($49.95 plus $2.50 shipping & handling) Illinois customers include sales tax.

☐ Charge my credit card:
☐ VISA ☐ MasterCard Interbank No._____
Card No._____Exp. Date_____

Name_____

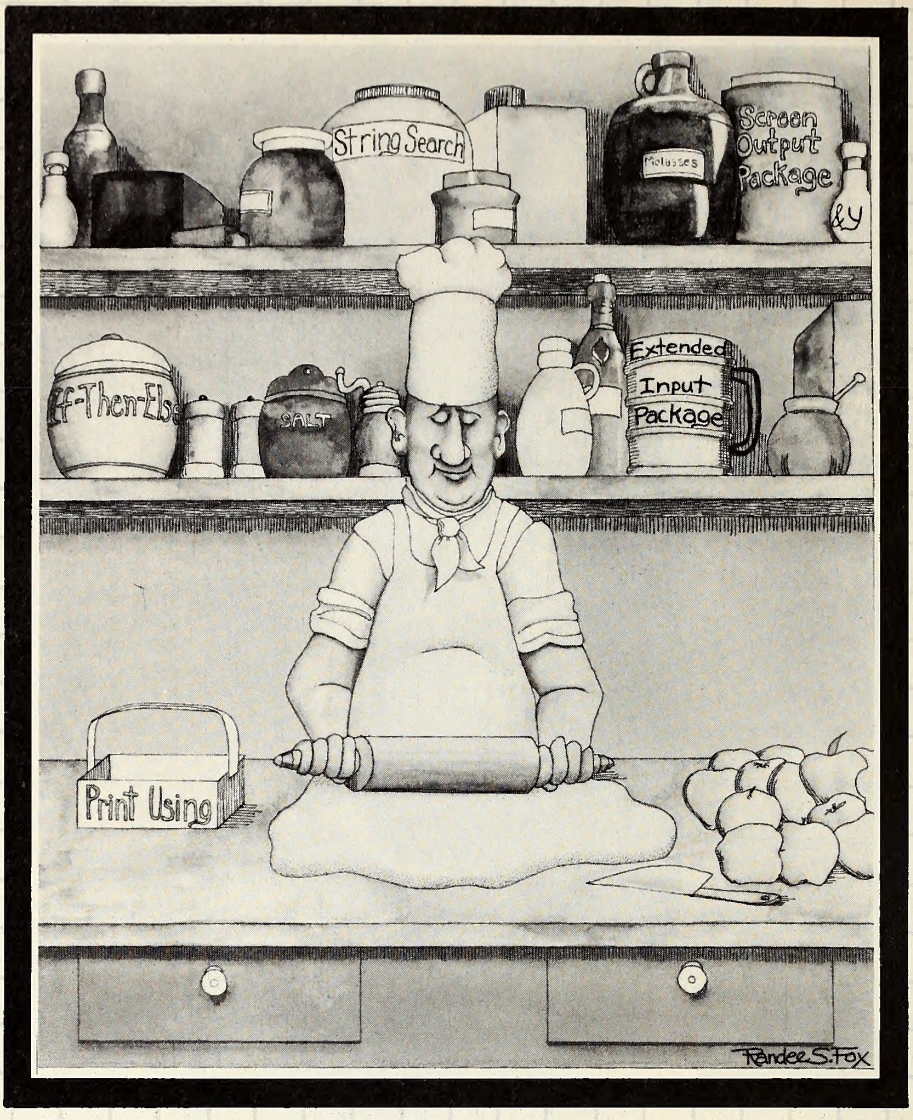
Address_____

City_____State_____Zip_____

Signature_____

Outside USA add $10 per unit. Airmail extra.

FlipTrack® Training Tapes
Division of Mosaic Media, Inc.
Dept ST-1, P.O. Box 711,
Glen Ellyn, IL 60137



The extended Basic for your Apple.

APPLE SPICE

$29.95

042-0133
APPLE 2 DISK 48K

By: Corey Kosak & David Fox

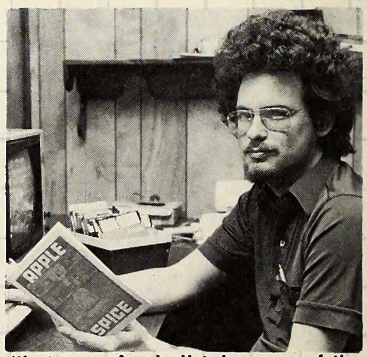
ALL YOU NEED TO EXTEND YOUR APPLESOFT BASIC.

APPLE SPICE is a set of fast, assembly language routines that alleviate some of the drudgery of writing programs in Applesoft BASIC. APPLE SPICE allows you to concentrate more on your program logic and less on things like "prettifying" the screen and detecting keyboard entry errors. APPLE SPICE also greatly expands Applesoft's capabilities by adding features found in many larger BASICs such as PRINT USING, STRING SEARCH (INSTR), IF-THEN-ELSE, and LINE INPUT.

- Easy-Patch - No machine language knowledge Beginners can use it.
- Excellent tutorial step by step manual. Written in easy to understand English.

APPLE SPICE is not copy protected. You may include APPLE SPICE in your commercial programs without additional charge. Just mention you used APPLE SPICE on your title screen and in your documention.

APPLE SPICE comes with a clearly written hands-on manual (52 pages) that explains every feature in detail. All of the examples listed in the manual are included on the APPLE SPICE diskette, as well as some user-friendly programs that make it a snap to change APPLE SPICE'S parameters and to relocate APPLE SPICE to where YOU want it in memory.



"Let your Apple II take care of the details so you can just program!"

Adventure INTERNATIONAL
A DIVISION OF SCOTT ADAMS, INC.
BOX 3435, LONGWOOD, FL 32750 • (305) 862-6917

AVAILABLE FROM YOUR FAVORITE DEALER OR CALL TOLL FREE (800) 327-7172
SHIPPING & HANDLING EXTRA, PRICES SUBJECT TO CHANGE

DOSTALK

BY BERT KERSEY

Bert Kersey bought himself a new toy about three years ago. When it began talking back to him with the very kind of curiosity and humor he offered to it, he knew his Apple was more than a toy. Kersey claims to have established Beagle Bros a year and a half ago, but one look at its ads (the originals of which arrive slightly yellowed and wrapped in 1906 newspapers) makes it obvious that Beagle Bros has been around much, much longer than that. In fact, it's rumored that the microcomputer was inspired by a Beagle Bros ad for a hand-cranked foot massage and back exercise machine that was shaped just like an Apple II.

In truth, Kersey creates all his own ads, a carry-over from his pre-Apple occupation as a graphic designer. Actually, truth has little to do with it; Kersey, definitely the dominant brother of Beagle Bros, has no brother. He and Sharon, his wife, run the company. He does, however, have a beagle, Sophie, and that has made all the difference. Kersey insists that the beagle is not an officer of the company, nor can she type, nor is she related to Kersey, certainly not as a brother.

We have our doubts.

Nevertheless, we hope you'll enjoy DOStalk *by Bert Kersey (or Sophie the beagle).*

Hello out there. Over the next few months, we're going to explore Apple's Disk Operating System, find out what makes it work, and learn how to make it work the way we want (yes, you *can* teach old DOS new tricks). We'll assume you own a copy of the 3.3 Manual and already understand the basics of loading and saving files and so on. With that knowledge and a little creative *peeking, poking,* and probing, you can really have some fun. Let's get on with it.

What Is DOS? DOS is actually a complex high-speed machine-language *program* that is inserted into your Apple's memory when you boot a disk. Without DOS (rhymes with *boss*, by the way, not *gross*), an Apple simply does not know how to perform any function that involves a disk. Let's prove it with an experiment or two; read or type along.

First, let's create a DOS-less Apple. Assuming you have a normal Apple Plus with Applesoft built in, power up with no disk in drive 1. After a whir or two from your drive, hit reset. The motor will stop, and you will see the Applesoft prompt (]) and a flashing cursor. Now type in a short program and *run* it just to prove that your Apple still works. How about this little goodie:

```
10  GOTO 20 SCRN( + > COLOR= OR
    AT READ AT THEN INT COLOR=
    THEN > INT NEW (sic)
20  HOME: FOR I = 2056 to 2070:
    PRINT CHR$(PEEK(I));: NEXT
```

The point here is that you've still got a real Apple computer sitting there in front of you, even without disks and disk drives. *Basic commands* and procedures work exactly the same without DOS as with, but the 28 *DOS commands* like *catalog* and *FP* and *init hello*, not to mention *print CHR$(4); bload Hopalong Cassidy* won't get you anything but the dreaded (beep!) "syntax error," Apple's way of saying "Huh?" You won't even be able to scrounge up a DOS error message. "File not found," for example, is nowhere to be found. And you'll never see "syntax error" without a leading question mark.

You can, however, enter complex programs into memory and edit and run them. You can save and load programs too, but only to and from cassette (an ancient recording device not fully understood today). You can perform all the usual computer calculations and screen tricks but, without DOS, there is no way to save a program to disk, catalog, write text files, or do anything that involves your disk drive or system.

So much for the DOS-less Apple. Now stick in a disk and boot DOS with a PR#6. Booting actually transfers the DOS program from your disk into your Apple's memory. DOS normally remains in memory as long as your Apple is on; functioning, but unchanged by anything you ordinarily do—programming, loading, saving, deleting, and so on. Booting is necessary to install DOS into memory where you can use it. The word *boot*, by the way, was probably invented by the same group that thought up *muffin*.

Now, with DOS in memory (RAM), and Applesoft or Integer in memory (ROM), your Apple knows *two sets* of commands and error messages, DOS and Basic. When you enter an instruction through the keyboard, the Apple checks it *first* to see if it is a DOS command, *then* to see if it is a Basic command. If you type the word *wowzo*, for example, the Apple checks its entire twenty-eight word DOS command vocabulary one-by-one (words like *catalog, delete, rename*) to see if it knows *wowzo*. Since it doesn't, it then checks its Basic command vocabulary (words like *list, gosub, next, poke*). When it doesn't find *wowzo* there either, it gives up and prints "?syntax error" (Applesoft) or "*** syntax err" (Integer).

If it thinks you made a DOS typo (for example, if you typed *catalogg*, the Apple prints "syntax error" (the *DOS version* with no question mark). If your Apple *does* know the word you have typed, it executes the command according to the instructions that reside in memory, either DOS or Basic, depending on where the command was found. All of the above takes approximately no time at all (and sometimes a little less).

Peeking at DOS's Vocabulary. In 48K Apples, DOS normally occupies memory locations $9600 (38400) through $BFFF (49151). You can look at DOS or any part of memory from Basic with *peek* commands. To find the value at location 43380, in the middle of DOS, for example, type:

```
PRINT PEEK(43380)
```

No line number is necessary; just type the command and hit *return*. The Apple should answer with a 76. To find the equivalent ASCII character at that location, type:

DOSTALK

```
PRINT CHR$(PEEK(43380))
```

The above command should produce an L. Clear memory by typing *new* or *FP*, and run this program:

```
10  FOR X=43380 TO 43401
20  PRINT CHR$(PEEK(X));
30  NEXT X
```

You should see a "language not available" message on the screen. You have uncovered DOS's first error message! Change Line 10 to:

```
10  FOR X=43380 TO 43581
```

and you are presented with all fourteen DOS error messages strung together. To further examine this mysteriousness, add two more lines to your program:

```
15  NORMAL: IF PEEK(X)>127 THEN INVERSE
25  IF PEEK(X)>127 THEN PRINT
```

Run again, and you'll see that the *last character* of every error message appears in *inverse*. Line 15 makes the letter inverse if it has an ASCII value greater than 127. In this situation, each character has two possible values. The letter D, for example, can be represented by a 68 or a 196 ($44 or $C4). In the error messages, the "high-byte" value is used to tell the Apple where the end of the message is. Let's scramble things temporarily; type this command (no line number necessary):

```
POKE 43452,68
```

Now attempt to call up a nonexistent file with a LOAD ZZYZX command. You should get the response "file not foundvolume mismatch." The Apple has printed error message number 5 starting at a designated starting point (the f in *file*) until it finds a high-byte character, in this case the h in *volume mismatch*, message number 6, instead of the d in *found*. Repair the damage before continuing by *poking* the high-byte value for d (196) back in where it belongs:

```
POKE 43452,196
```

Now change line 10 in your program to:

```
10 FOR X=43140 TO 43582: If X=43272 THEN X=43380
```

This will cause the program to *peek* at all twenty-eight DOS commands and all fourteen DOS error messages. You can change any command or error message simply by *poking* in new values for each character. Just keep the new command or message the *same length* as the old one and remember to *poke* in the high-byte value of the last letter. More on this in future articles. Now, let's do a different kind of snooping.

Poking Around in DOS. First, a warning: the following DOS experiments can be a bit risky, especially if you mistype a number or in case there's a typo in this article. Please do not have an irreplaceable disk in your drive while you experiment (and hide any disks you are fond of in the broom closet). The best plan is to *init* a new disk for test purposes.

IF YOU MESS UP A DISK, FORGET WHERE YOU READ THIS.

Let's go into the Monitor for a minute or two to look at DOS's *catalog* routine. We promise to lead you right back out again. Type *call −151*, return, and then *ADA3L*, return. This command disassembles machine language data from the Monitor starting at location $ADA3, showing you:

```
ADA3–   A9  16      LDA  #$16
ADA5–   8D  9D  B3  STA  $B39D
ADA8–   20  2F  AE  JSR  $AE2F
ADAB–   20  2F  AE  JSR  $AE2F
ADAE–   A2  0B      LDX  #$0B
ADB0–   BD  AF  B3  LDA  $B3AF,X
ADB3–   20  ED  FD  JSR  $FDED
ADB6–   CA          DEX
and so on.
```

Now check out the seventh line and write it down; you will need it later. "ADB3" is the location or *address* in memory (decimal 44467). "20 ED FD" is the command at that address. "JSR $FDED" tells that the command means "Jump to the SubRoutine at location $FDED" (just like a *gosub* in Applesoft). Now, let's be brave and cancel this command. Type:

```
ADB3: EA EA EA
```

Don't forget the colon after ADB3. Now look at the results by typing ADA3L, return. The command at ADB3 has been changed to:

```
ADB3–   EA  NOP
ADB4–   EA  NOP
ADB5–   EA  NOP
```

The three NOPs on the right stand for "No OPerate." The EAs (decimal 234s) you typed in have *cancelled* the command that was formerly at address $ADB3. *Three EA*s were used because there were *three two-digit hex numbers* following the address number. Often there will be only one or two hex numbers.

Now exit the Monitor with a control-C, return, and *catalog* your disk. Notice that the words *"disk volume"* are missing. To retrieve them, enter the monitor *(call −151)*, and type:

```
ADB3: 20 ED FD
```

Control-C back to Basic and *catalog* again. Your *disk volume* heading should have returned. If you made an error, you could be in trouble. A mistyped number or letter could scramble DOS and make a real mess of things. Remember, to reestablish normal DOS, you can always reboot.

Suppose you want to make your greeting program cancel the *disk volume* heading (it is, after all, confusing to beginners and usually unnecessary). Since you can't add "ADB3: EA EA EA," a machine-language command, to a Basic program, you will have to convert these numbers to their decimal equivalents and *poke* them in. Add these lines to your program:

```
10  POKE 44467,234: POKE 44468,234: POKE 44469,234
11  POKE 44480,234: POKE 44481,234: POKE 44482,234
12  REM REST OF HELLO PROGRAM . . .
```

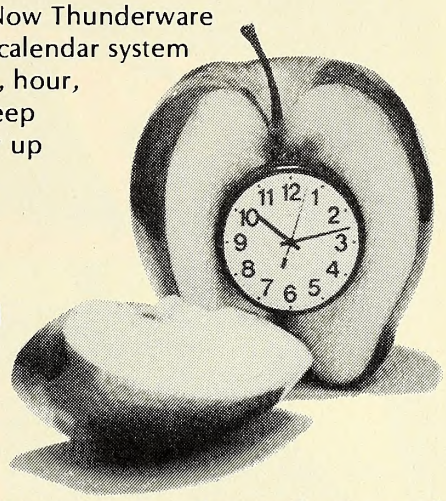
Line 10 deletes "disk volume" and line 11 deletes the volume *number*. Now when this program is *run*, the DISK VOLUME heading will not show when a *catalog* is displayed.

Another way to implement a DOS change is to *init* a new disk while the change is in effect. That way, every time *that disk* is booted, your Apple will behave according to the changes in memory when the disk was initialized.

Now, if you're still with me, you are ready for some experimentation on your own. Take any one, two, or three-digit machine language command between $AD98 (44440) and $AE69 (44649) and cancel it by entering one, two, or three EAs (or *poke* in 234s). *Catalog* your disk, and see what happens. Remember, what you are doing is *cancelling* commands, so

PUT YOUR APPLE TO WORK FOR YOU!
WITH THE THUNDERCLOCK PLUS™

As an APPLE user you already know all the things your APPLE can do. Now Thunderware expands that list with the THUNDERCLOCK PLUS, the complete clock/calendar system for your APPLE! Your programs can read the month, date, day-of-week, hour, minute, and second in any of APPLE'S languages. On-board batteries keep your THUNDERCLOCK running accurately when your APPLE is off - for up to 4 years before battery replacement. But that's just the beginning.



The THUNDERCLOCK PLUS is the most useful and versatile peripheral you can put in your APPLE. It can keep your disk files organized by time-and-date-stamping them, it enhances the usability of many of the new business/professional software packages for accounting, filing, and time management, and it can remotely control lights and appliances for security or display purposes in your business or home.

SOFTWARE PRODUCT COMPATIBILITY

Many of today's important software packages for data-base management, business applications, communications, and time management are designed to use the THUNDERCLOCK PLUS. If you have or plan to purchase any of these packages, a THUNDERCLOCK will greatly enhance their usefulness.

•VISIDEX* (Personal Software) •DB MASTER and MICRO-MEMO (Stoneware) •MICRO-COURIER and MICRO-TELEGRAM (Microcom) •THE CASHIER and THE STORE MANAGER (High Technology) •BUSINESS PLUS and NET-WORKS (Advanced Data Systems) ...and many others!

```
DISK VOLUME 254

*A 006 HELLO                07/07 16:37
*A 006 CLOCK                06/08 09:07
*A 004 FRAME                06/08 09:08
*A 004 DISK INFO            06/17 16:13
*B 003 BACKOFF              06/17 16:13
*B 005 SCREEN               07/24 17:32
*B 002 TCPUTIL              06/17 16:13
*B 004 SDTIME.O             06/17 16:13
*A 007 ADIGCLK              05/19 08:05
*A 011 SET TIME             06/08 09:08
*I 009 IDIGCLK              05/19 08:05
*A 007 TIME                 06/08 09:08
*A 003 SLOTFINDER           07/07 16:56
*A 014 DEMO                 06/17 16:14
```

THUNDERWARE'S DOS-DATER

Our new DOS-DATER software upgrades the regular DOS on your disks so that DOS will use the THUNDERCLOCK to time-and-date-stamp disk files. Every time a program is saved or a file is modified, the current date and time to the minute are stored in the CATALOG with the file's name. You can tell at a glance when a program was saved or when any file was last modified. And this time/date stamping feature is completely automatic. That means any program which uses DOS will time/date stamp its files!

REMOTE CONTROL

Add Thunderware's X-10 INTERFACE OPTION to your THUNDERCLOCK PLUS and your APPLE can control lights and appliances through your BSR X-10 Home Control System on your pre-defined schedules. Our powerful SCHEDULER software allows you to create and modify schedules easily and execute them in the 'background', while using your APPLE for other tasks in the 'foreground'. Use your APPLE for energy management, display and security lighting, or laboratory/process control.

Our PASCAL Software lets you use all the THUNDERCLOCK'S features in PASCAL and sets the F)iler date whenever you boot.

You get all this versatility in just one peripheral system. Backed by a full one year warranty. See your APPLE dealer for a demonstration, or contact us for more information. We'll give your APPLE the best time around!

Suggested retail prices:

THUNDERCLOCK PLUS	$139
X-10 INTERFACE OPTION	$49
PASCAL SOFTWARE DISK	$29
DOS-DATER/DEMO DISK	$29
MANUALS ONLY, each	$5

Distributed by Apple Computer, Inc. and Computerland Corp.

THUNDERWARE, INC.
P.O. BOX 13322
Oakland, CA 94661
(415)-652-1737

*Requires software supplied on DOS-DATER/DEMO disk.

Attention Pinball Wizards! David's Midnight Magic™ will really make you flip. This amazing electronic pinball experience by David Snider brings all of the fast action and tension of a real pinball game to the computer screen. For one to four players.

You have flippers top and bottom. Just like the real thing. You have bumper action. Rollovers. All the sounds of a classic arcade pinball machine. And all the lights, combined with the outstanding Hi-Res graphics Brøderbund is known for.

Want to put some English on the ball? Go ahead, you can even jostle this electronic pinball machine. Easy does it, though. Overdo and you'll ring up a tilt!

What else? You control the release velocity. And if you're good, you can trip up to three balls for simultaneous play. Just like the real thing.

If you're good enough to be among the top ten scorers, your name is automatically displayed on the screen at the end of each game.

To get flipping with David's Midnight Magic, visit finer computer and hobby stores, or send $34.95 by check or money order to Brøderbund Software, 1938 Fourth Street, San Rafael, California 94901. Visa and Mastercard phone orders also accepted: (415) 456-6424. California residents please add 6% sales tax.*

Brøderbund Software

David's Midnight Magic

**Requires 48K Apple II Integer/+, DOS 3.3 or 3.2.*
Apple is the registered trademark of Apple Computer, Inc.

DOSTALK

watch for what *doesn't* happen as much as for what does.

Here are a few catalog modifications I have discovered:

```
ADD9: EA EA or
POKE 44505,234: POKE 44506,234
```

displays deleted file names in a catalog.

```
AE22: EA EA EA or
POKE 44578, 234: POKE 44579,234: POKE 44580,234
```

cancels carriage returns after file names in catalog.

```
AE34: EA EA EA or
POKE 44596,234: POKE 44597,234: POKE 44598,234
```

cancels catalog stop when screen is full.

```
AE37: EA EA or
POKE 44599,234: POKE 44600,234
```

stops catalog at each file name and waits for return.

You will encounter many strange occurrences, both useful and useless (and sometimes *destructive*), with this method of experimentation. Often your Apple will hang and eat its cursor or crash you into the Monitor, requiring you to hit reset. Remember, when all else fails, kill the power and *reboot.*

The following program is an aid for those of you (us) who don't think in hex. It does disassemblies for you in DOS or anywhere else in the Apple, converts all of the hex values to decimal, and displays both on the screen. To use it, type *run* and enter the starting address (0–65535) of the disassembly. To continue to the next set of addresses, hit return. To start at a new address, enter a new decimal number.

```
10  REM
==================
   DECIMAL DISASSEMBLER
        BERT KERSEY
==================
20  TEXT : HOME : NORMAL
30  DIM SC(24): FOR I = 1 TO 24: READ SC(I): NEXT : DATA 1024,1152,1280,
    1408,1536,1664,1792,1920,1064,1192,1320,1448,1576,1704,1832,1960,
    1104,1232,1360,1488,1616,1744,1872,2000: REM (HTAB 1 SCREEN
    LOCATIONS)
40  GOTO 120
50  POKE 58,ST - INT (ST / 256) * 256: POKE 59, INT (ST / 256)
60  HOME : INVERSE : PRINT "-------HEX------";: HTAB 21: PRINT
    "------DECIMAL-----";: NORMAL
70  VTAB 1: CALL - 415: REM (DISASSEMBLER)
80  FOR V = 2 TO 21:H = 0: GOSUB 150:XHI = 256 * X:H = 2: GOSUB 150:
    VTAB V: HTAB 21: PRINT XHI + X;"-";: CALL - 868
90  HTAB 28: FOR H = 8 TO 14 STEP 3: GOSUB 150: IF X > = 0 THEN PRINT
    SPC((X < 100) + (X < 10));X;" ";
100  NEXT : NEXT
110  PRINT : VTAB 24: PRINT "OR <RETURN> TO CONTINUE OR <Q> TO
     QUIT.";
120  VTAB 23: HTAB 1: INPUT "ENTER NEW LOCATION (DECIMAL):";ST$: IF
     ST$ = "" THEN 60
130  IF ST$ = "Q" THEN VTAB 22: CALL - 958: END
140 ST = VAL (ST$): GOTO 50
150 P1 = PEEK (SC(V) + H):P2 = PEEK (SC(V) + H + 1):X = 16 * (P1 - 176 - 7 *
    (P1 > 185)) + P2 - 176 - 7 * (P2 > 185): RETURN : REM
    (CONVERTS HEX ON SCREEN TO DECIMAL)
```

DOS Mystery of the Month. Here is a useless but perplexing program by Chris Volpe of Trumbull, Connecticut. Assuming slot 4 is unused and you have a disk drive connected to Slot 6, *run* this program:

```
10  PR#6: REM DRIVE SLOT
20  PR#4: REM UNUSED
```

If your Apple works like mine, it hangs up as if line 10 did not exist. Hit reset to retrieve your cursor. Now type *trace* and run it again. It works; it boots the disk! No one around here knows why, though. Let us know if you can figure it out.

See you next month. ❏■

THE PASCAL PATH

By Jim Merritt

Tools Of The Craft, Part 10

At Long Last, Input. All the programs that we've seen so far have processed data in the form of declared constants. For instance, here's a program that adds two integers and displays the result:

```
PROGRAM
  Adder;
(* Displays numbers NumA, NumB, and
   (NumA + NumB) *)

  CONST
    (* Minimum field width for displayed
       Integers. *)
    FieldWidth=          1;

    (* Numbers to add *)
    NumA=          3;
    NumB=          8;

BEGIN (* Adder *)
  WriteLn(NumA:FieldWidth, ' + ',
          NumB:FieldWidth, ' = ',
          (NumA + NumB):FieldWidth);
END  (* Adder *).
```

Suppose you want to add two different numbers. To change the data in this program, or in any other so designed, you must edit the program text, modify the constant definitions, and recompile. (In this case, you would modify either or both of the constants NumA and NumB.) Up to now, this process has been only mildly inconvenient, because we have been dealing with relatively small and simple programs that are quickly edited and compiled. As you might expect, however, this method of changing a program's behavior will cost us more and more time and effort as we move on to larger and more complex programs. There is, of course, an easier way. Instead of defining a program's initial data as constants within the program text itself, we can design the program to accept data from an outside source (the keyboard, for example). Data so acquired is called *input data* (often abbreviated as, simply "input").

Learning to Read. For simple input, Pascal provides the standard procedures Read and ReadLn, which complement their counterparts Write and WriteLn. Where Write and WriteLn permit your program to send information to the console screen, Read and ReadLn permit it to accept information from the console keyboard. (Because Read and ReadLn are so similar, I'll mention ReadLn specifically only when a distinction needs to be made between the two procedures. Otherwise, whatever I say about Read applies equally to ReadLn.)

The parameters to Read must be names of variables, into which input data may be stored. Read accepts any number of parameters, in any order, so long as the type of each parameter is either Char, Integer, Real or String. (See figure 1; note that, like Write, Read does not accept parameters of type Boolean, or of any enumerated type.) If you call Read with more than one parameter, it will acquire data for each parameter, in order, until an input value has been assigned to every one. The data type of a parameter to Read determines the type of datum that the computer will accept for—and thus put into—that object. For instance, if you would like your program to accept an Integer number from the keyboard, you would call Read with an Integer variable, say Intvar, as the parameter:

```
Read(IntVar);
```

To have your program acquire two Integer numbers, NumA

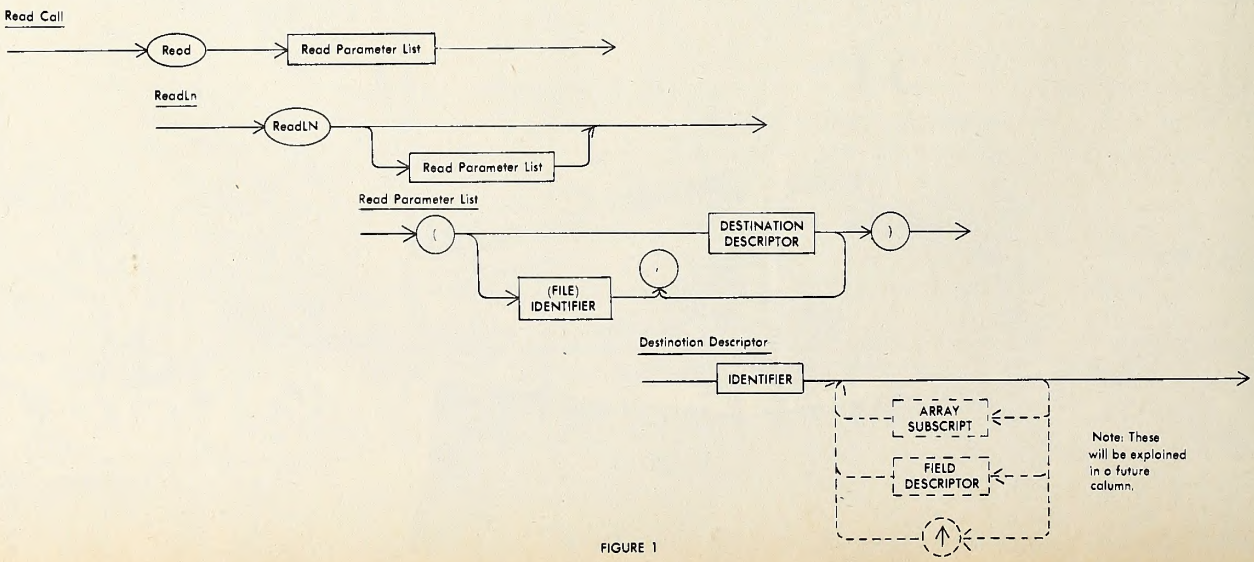


FIGURE 1

and NumB, in that order, the following call to Read is appropriate:

```
Read (NumA, NumB);
```

Let's modify Adder so that it will accept NumA and NumB from the keyboard. First, we must change NumA and NumB from constants into variables in order to permit Read to assign arbitrary values to them. Next, we insert a call to Read, prior to the addition calculation. The old Adder is now entirely converted into a program, *Adder2*, which gets its data from the user.

```
PROGRAM
  Adder2;
(* Acquires the Integers NumA and NumB from
   the console, then displays both them and
   their sum. *)

  CONST
    (* Minimum field width for displayed
       Integers. *)

    FieldWidth=          1;

  VAR
    (* These hold the numbers to add *)
    NumA, NumB
      :Integer;

BEGIN (* Adder2 *)
  Read (NumA, NumB);
  WriteLn (NumA:FieldWidth, ' + ',
           NumB:Fieldwidth, ' = ',
           (NumA + NumB) :FieldWidth);
END  (* Adder2 *).
```

You may use *Adder2* over and over again, giving different input data each time, without having to modify the program or recompile it. Unfortunately, there is one small problem with this new version of Adder. To find out what it is, compile and execute the program. What happens? The computer appears to stop dead. Of course, it is actually waiting for you to put in the two numbers, NumA and NumB. Type the following:

```
23 42
```

Be sure to separate the two numbers with at least one blank, and conclude your input by pressing the <RETURN> key. *Adder2* then responds as we expect, by displaying ("echoing") both of the input numbers, in addition to reporting their sum.

Prompting the Input. When it waits for you to type NUMA and NumB, *Adder2* gives absolutely no indication that it is expecting input, nor any clue as to what that input should be. Unless you have seen and understood the source program, you are likely to be baffled by the apparently "dead" computer. For this reason, computer professionals would say that *Adder2* is "not friendly." At the very least, a "friendly" program should prompt the user when it expects input. "Friendlier" programs tell the user what form of data is expected, and even how to type it. Extremely friendly programs are capable not only of recognizing when the user has supplied inappropriate input data, but also of notifying the user of the situation, suggesting ways to correct it, and giving the user opportunities to try again. (This last behavior falls under the heading of error detection and recovery, which is the subject of a future column.) *Adder3*, shown below, is only moderately friendly, in that it merely prompts the user for NumA and NumB. Still, this is an improvement over the earlier, silent version.

```
PROGRAM
  Adder3;
(* Acquires the Integers NumA and NumB from
   the console, then displays both them and
   their sum. *)

CONST
  (* Minimum field width for displayed
     Integers. *)
  FieldWidth=          1;

VAR
  (* These hold the numbers to add *)
  NumA, NumB
    :Integer;

BEGIN (* Adder3 *)
  (* Prompt assumes that user is experienced
     at supplying Integer input to Pascal
     programs. *)
  Write('Enter two Integers: ');
  Read (NumA, NumB);
  WriteLn (NumA:FieldWidth, ' + ',
           NumB:Fieldwidth, ' = ',
           (NumA + NumB):FieldWidth);
END  (* Adder3 *).
```

As you can see, prompting involves no exotic techniques; all you need do is send an appropriately worded message—the prompt—to the console screen, usually by calling Write or (occasionally) WriteLn. Note that, if you use Write to issue your prompt, the user will enter input data on the same display line as the prompt. Consequently, it's generally good practice to put a blank at the end of the prompt, as I did in *Adder3*, in order to separate it from the input data on the screen. If you use WriteLn to issue a prompt, the input data will appear on the display line that follows the prompt, so there is no need to include a "trailing blank" in the message.

From this point on, we'll be using prompts and prompting techniques in almost every program, so there's not much need now to talk at length on the philosophy of prompting—that is, how to word a prompt, how to decide whether or not a prompt is necessary, and so forth. You should be able to acquire a feel for proper prompting strategy and methods by examining our sample programs. However, a few comments here may start you thinking about this important issue, and the sooner you begin to consider it, the better.

In general, prompts should be as brief as possible without being cryptic. Brevity is desirable for two reasons: first, long prompts take longer to display, and longer to read, than short ones; long prompts are therefore more likely to contribute to user boredom or impatience, especially as the user gains experience with the program and begins to memorize the prompts from sheer repetition. Second, the display screen is limited in its dimensions. Ideally, you should choose prompts so that both a prompt and the input data associated with it may be displayed simultaneously on the screen (usually on the same display line). Long prompts leave less room for the display of input data, and should therefore be avoided.

The helpfulness of a prompt depends, of course, on the sophistication of the user. For example, consider the Apple Pascal main prompt line, which enumerates the single-character commands that are recognized by the operating system. I doubt that someone who has no prior experience with computers would be able to determine the significance of the prompt line simply by looking at it. To the uninitiated, therefore, it is cryptic. On the other hand, it doesn't remain cryptic for very long, since the concept behind the main prompt line is easily grasped, and beginners have been known to master it in minutes, or even seconds. Certainly, in order to be entirely self-explanatory to the complete novice, the prompt would have to be longer and wordier. Such verbosity would irritate the experienced users, who far outnumber the novices, especially since novices usually acquire some reasonable level of sophistication within the first five minutes of using the system!

In coming up with the main prompt line, then, the designers of UCSD/Apple Pascal made certain assumptions about who the typical user is, and what she knows. The assumptions that you make about the sophistication of the user, and the limitations of your display device (if any), should be listed as comments within your program, as I've done in *Adder3*. These are the constraints that affect your decisions as to when and how your program should prompt its users. They will be invaluable reference material for yourself at a later date, or for someone else who tries to understand or modify your program.

Syntax for Numeric Input. Figure 2 repeats the syntax dia-

grams for Integer and Real literals. A numeric datum must be entered as a literal, using one of these formats, or Pascal will either reject it outright (if possible), or accept only part of it, ignoring the erroneous portion. Pascal accepts your numeric input, character by character, building up each number with your every keystroke, and checking to see that each new character conforms to the appropriate syntax diagram (for Real or Integer, as determined by the type of the variable into which the value is being read). Whenever it encounters a character for which there is no provision in the diagram, Pascal assumes that the input number ends there (and that any subsequent datum begins there).

Often, you will be in the process of entering a number when, by mistake, you press an inappropriate key. If, by this time, Pascal has been able to build a legal number, that value, whatever it is, is assigned to the corresponding variable, and Pascal proceeds to acquire a value for the next item in the Read parameter list. If Pascal has not been able to come up with a proper number by the time you press the "illegal" key, the system will notify you of an I/O Error and complain of a bad input format.

If you'd like to see these principles applied, run *Adder3* and begin your input with a letter (upper-case or lower-case, either is fine). At the very start, *Adder3* tries to collect an Integer value for NumA, but gets a letter, instead of a "legal" character. This indicates that data input for NumA has ended, but no value has yet been built! The Pascal system takes over at this point, and responds to the unsavory situation aborting *Adder3*'s execution, and issuing an error message. When you see the error message, you should press the space bar, in order to regain the main prompt line. Once you've done this, execute *Adder3* again. This time, type

```
10p
```

That is, type a legal Integer, followed immediately by an inappropriate character. (If you wish, you may also put one or more spaces between the integer and the bad character—the result will be the same.) Again, you get the expected "bad format" error message, but this time because Pascal could not finish building a value for NumB. If you could peek inside the computer, just prior to receiving the error message, you would see that NumA, in this case, was assigned the value 10. Upon encountering the *p* Pascal terminated the data acquisition for NumA and tried to go on to the next datum, NumB, assuming that the *p* would be its first input character. Of course, there is no provision for *p* in the Integer syntax diagram, and so Pascal immediately quit gathering data for NumB, without having first built up any useful input value. Consequently, Pascal was forced to terminate the program, and issue an error message.

Finally, try yet another run of *Adder 3*, with the following input:

```
1 2z
```

As in the previous examples, Pascal doesn't give you a chance to press the return key to terminate your input. Unlike the two earlier attempts, however, this one is successful. Pascal builds NumA until it gets the blank that separates the 1 and 2, then builds NumB until it encounters the letter *z*. The value 1 is assigned to NumA, and 2 to NumB. Since no further data is required by *Adder3*, the *z* remains unused.

Notice that, in this example, the displayed input data and *Adder3*'s final output are crowded together on the same display line. This is because you weren't able to use return to signal the end of your input. Unless you explicitly direct otherwise, Pascal gives back to you (on the screen) whatever you type into it (from the keyboard), and it does this before going

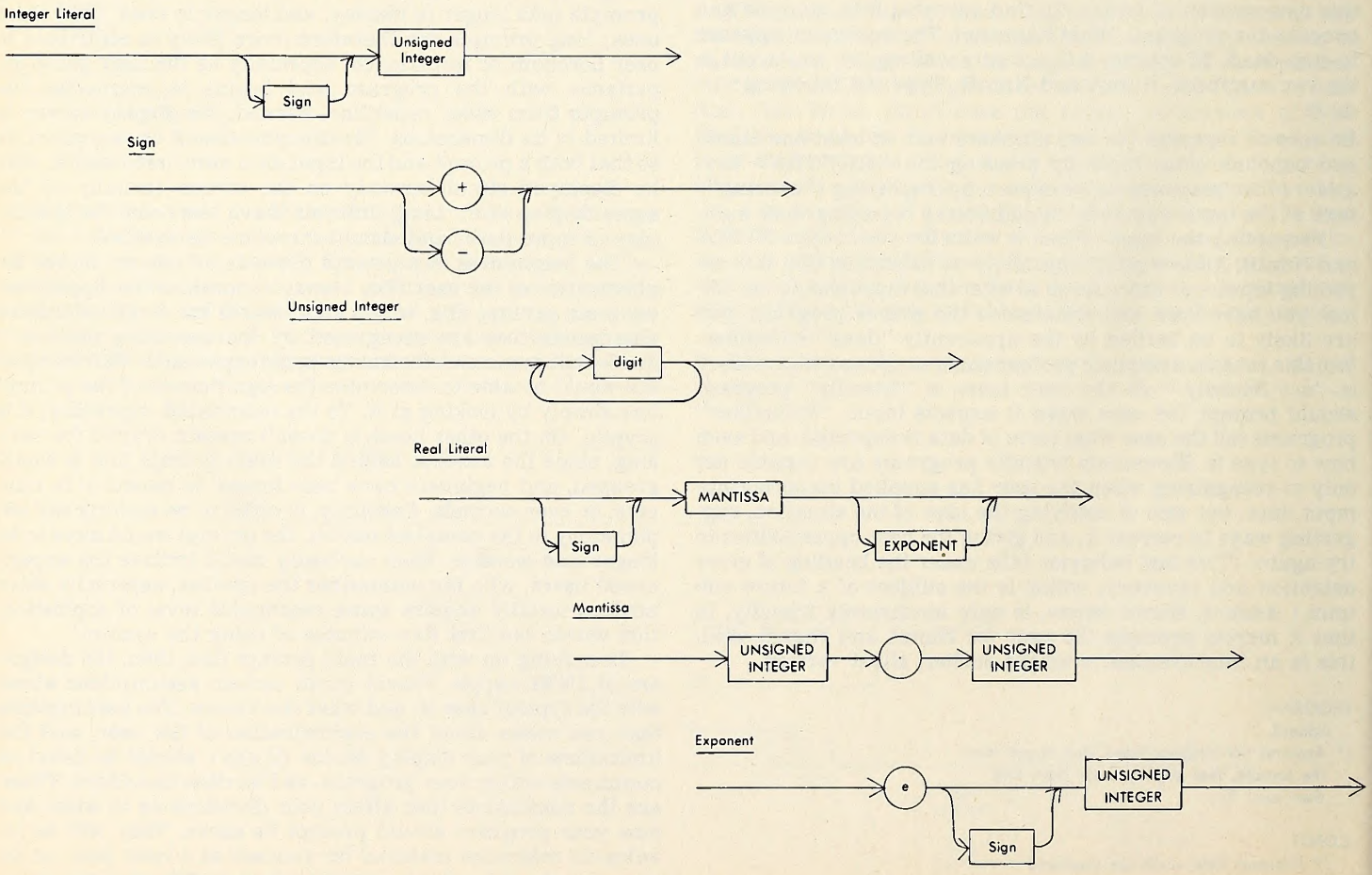


FIGURE 2

We've already written 80% of your next program.

Imagine writing better, faster business programs in 20% of the time. Now you can, with this revolutionary programming tool. Let's face it...the great majority of programming is pure busywork. Every instance of creativity may require hours of generating code. Not anymore. The Tool™ takes care of the busywork, letting you concentrate on creating.

Design attractive entry screens, validate, format, and store user input, with virtually no programming. The Tool does it for you. What used to take days — even weeks — now takes only minutes. And the result is better, faster, more professional programs.

The Tool provides an entry screen generator, a database manager, and a report formatter, each with very sophisticated capabilities.

ENTRY SCREENS:

The Tool allows you to make attractive, useful entry screens a standard part of your programs. It handles all screen editing, field formatting, and entry validation for you.

Character Validations:
alpha / numeric / blank / yes or no / signed values / password / Radix 50 / decimal point / dollars and cents

Field validations and formats:
range / valid entry / valid date / not in range / not valid entry / defaults / equal / not equal / basic check / left justify / right justify / leading zeros / invisible field / dollars and cents / 100th of cent

DATABASE:

The Tool Operating System (TOS) allows files to span multiple disk drives. Whether you have four 20-megabyte drives or 8 floppy drives and 254 diskettes, TOS can use the storage space as one large file or several — it's up to you.

The database manager of The Tool offers: fast read and write / read and write directly to variables / over 15 million records / sequential and random access / multiple disk drives / partial and masked key lookup / Corvus or floppy support / record length as large as memory

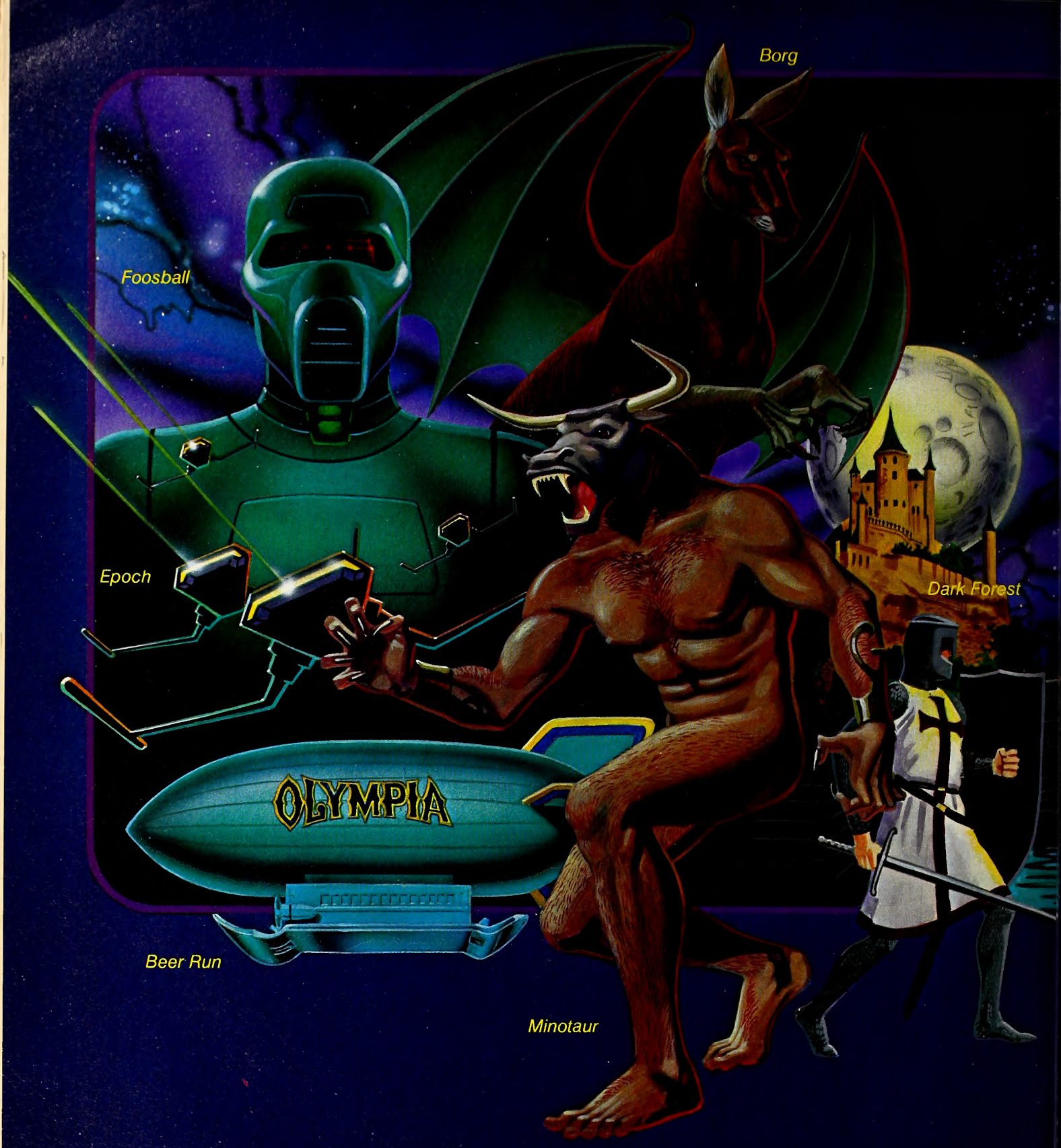
REPORTING: easy report definition / define reports directly on screen / multiple column widths / calculations at print time / totals / paging / subtotals / page numbering / headers / title

- The Tool uses less memory than old conventional programming methods.
- The Tool's many features can be used from BASIC, giving you ultimate flexibility.
- The Tool is already tested, saving you testing and debugging time.

Using The Tool is actually like having an experienced Apple II™ programmer do most of the work for you.

High Technology Software Products, Inc., P.O. Box S-14665, 2201 N.E. 63rd, Oklahoma City, Oklahoma 73113
(405) 478-2105

High Technology Software



WE'RE EXCITING, WE'RE

FUN . . . WE'RE SIRIUS™!

Sirius Software, Inc. 10364 Rockingham Drive, Sacramento, CA 95827. Dealer inquiries invited (916) 366-1195.

on to other business. This process is called "echoing the input," and it is the computer's way of letting you know that your commands and data are being received loud and clear. When you press the return key, Pascal "echoes" this by advancing the screen's cursor to the beginning of the next display line. Thus, when you conclude data input by pressing return, subsequent output from the computer must start "cleanly," on a new line. In this case, the letter *z* concluded your input. Being only a letter, *z* has no magic, line-advancing properties, so the cursor remained in the character position that followed *z*. Since *Adder3* issued no calls to WriteLn prior to reporting the sum of the two input numbers, the final output was bound to run together with the input data display.

Blanks Separate Numeric Input Data. When providing numeric data to a program that uses Read (and/or ReadLn) to collect its input, you should insure that at least one blank separates each datum from its predecessor, and you should not use other characters for this purpose. The blank is the Pascal system's natural separator. Because it is not a part of the syntax for either Integer or Real literals, it marks the end of an input literal, just as *p, z, #,* or any other "inappropriate" character does. But blanks are special, in that Read will skip over them while it is trying to find the first "legal" character in an input literal. In other words, any number of blanks may precede a numeric input datum; Pascal will ignore these "leading blanks." In using *Adder3,* suppose you type the following, in response to the program's request for input:

```
17 299
```

As usual, you conclude your input by pressing the return key. Let's examine what *Adder3* does with your input. First, it receives a sequence of digits (17) that corresponds to an Integer literal. Then it gets a blank, which marks the end of the datum; the value collected so far, 17, is assigned to NumA. Pascal assumes that the character which identifies the end of the first datum also begins the next. But, since this character is a blank, it is ignored (as *p, z,* or *#,* for example, would not be). If any other blanks stood between the first and second data, they would also be ignored. Finally, Pascal begins to get more digits, and is able to build up the Integer value of 299, which is assigned to NumB as soon as you press the return key.

Return is Treated as Blank. The return key is also accorded special treatment by the Read procedure; it is always translated into a blank, and so may be considered as equivalent to the blank, for purposes of this discussion. The equivalence between return and the blank implies that an acceptable response to *Adder3* is to type one Integer literal, then press the return key, then type another literal, and press the return key again. Try it.

It works! It's completely a matter of your own personal taste as to whether you use return, blank, or both to separate numeric data, or to terminate numeric input. My rule of thumb is to group related data on a single input line whenever I can, using one or more blanks to separate each datum from its predecessor, and return to end the line.

Although we've given only Integer parameters to Read so far, Pascal's acquisition of Real input data is similar; it skips leading blanks, expects the user to observe the syntax for a Real literal (which, as shown in figure 2, is slightly more complicated than that for an Integer), and terminates the data collection upon the receipt of a character that does not conform to the syntax. If no usable value has been built before the input is terminated, Pascal issues an error message, just as it does in the Integer case.

The Difference Between Read and ReadLn. Remember that WriteLn may be called with or without parameters, and that the call

```
WriteLn(A, B, C);
```

is equivalent to the compound statement

```
BEGIN Write(A, B, C); WriteLn END;
```

IMAGE PRINTER SERIES

Sensible Software is proud to introduce our new series of high resolution screen dumps. IMAGE PRINTERS provide a simple way to transfer high resolution graphic images onto paper. Each program in the series has unique features that give you full control of the printing. Some of the included options are:

- Full control over the area of the HIRES screen to be printed. You graphically pick the area for the utmost ease and accuracy.
- One-step printout of the picture with the ability to pause or abort the printing at any time.
- Menu-driven. All options are invoked with single keystrokes. IMAGE PRINTERS are extremely easy to use.
- Multiple image sizes, 6 different sizes for letter quality printers, 4 sizes for other printers.
- Creation of an inverse (negative) image for reverse printing.
- The ability to save the compressed and inverse images to disk.
- One time configuring for your printer and interface card. Why answer all those questions about your printer each time you want to print a picture?
- The images may be printed anywhere on the page.
- IMAGE PRINTERS support most popular interface cards, such as cards from Apple, California Computer Systems, Epson, and Mountain Computer. (The SSM AIO Serial Card and user-written 'driver' routines may be used with the letter quality printers.)

There are three separate versions of IMAGE PRINTERS, each one tailored to take full advantage of a different printer.

IMAGE PRINTER—LETTER QUALITY. For all popular letter quality printers (Diablo, NEC, Qume, etc.)

IMAGE PRINTER—EPSON. For the popular Epson MX-70, MX-80 and MX-100.

IMAGE PRINTER—NEC PC-8023A. For the NEC dot-matrix printer.

All versions are available for $40.00

Please specify version desired.

THE BUG

A sophisticated, yet easy to use diagnostic aid for getting "the bugs" out of your assembly language programs.

If you are a novice just getting started with assembly-language programming, you will also find The BUG helpful in developing your understanding of how the Apple's 6502 internal processor operates. The many display options of The BUG will permit you to try out your assembly-language programs at the speed that is most comfortable for you. The BUG will also make it easy for you to see the effect of your program on the Apple as it executes.

If you are a professional programmer, you will find that The BUG can improve your efficiency by reducing the time you spend identifying and solving complex, assembly-language programming errors. You will particularly appreciate the fact that The BUG offers the easiest to use and most extensive breakpointing capability of any "debugger" available for the Apple. Up to 13 different breakpoints can be specified to halt program execution when either: 1) a particular program location is reached, 2) one of the 6502 registers reaches a specified value, or 3) one of the bits in the 6502 status register reaches a specified value.

Another key feature of The BUG that serious programmers will appreciate is the ability to AUTOMATICALLY run lower-level subroutines at FULL SPEED. You no longer have to keep debugging the portions of your program that you already have working.

This is not the least expensive "debugger" program for the Apple, but we challenge you to find more capability for less money!

The BUG is supplied with a 40+ page user guide and is designed for use with DOS 3.3 on either the Apple II or Apple II Plus computer. . . . . only **$50.00**

BUILD USING

. . . . provides an easy to use print-using routine plus similar functions for strings. Creating charts, reports and general screen formatting becomes a simple task. BUILD USING is written entirely in machine language and provides a simple means of avoiding garbage collection (those unnecessary delays that slow down your programs). With BUILD USING, you can choose how many digits should be displayed to right and left of the decimal point, and you can even fill the leading positions with the character of your choice. For example, you can print the number '157.23' as '157.2', or '0000157.230', or '* * * * * $157. AND 23/100 DOLLARS', or hundreds of other ways (including exponential formats). Working with strings is just as easy; it's a snap to convert names from 'John" and 'Doe' to 'Doe, J.'. Also included are three levels of error trapping, so you can trap and correct numbers or strings that cannot fit in your specified format.

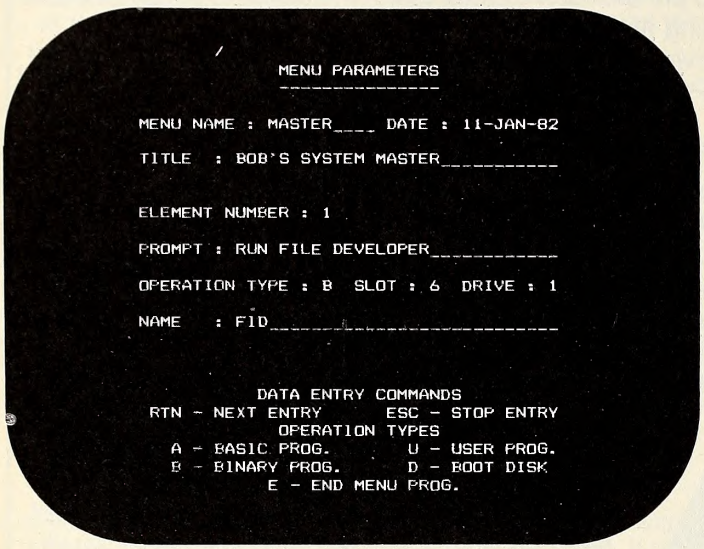
Utilities like BUILD USING are usually difficult to use because they must be located in one memory location (usually between DOS and the DOS file buffers); they cannot be used with your favorite editor or other special routines. BUILD USING does not have this limitation, as it can be easily located in many different memory locations: 1) the "normal" between DOS and DOS file buffers, 2) at HIMEM, 3) APPENDED to your Applesoft program, or 4) anywhere else in memory. Appending BUILD USING to your program is as simple as EXECing a TEXT file. BUILD USING uses the "CALL" command thereby leaving the ampersand vector free for your own use.

BUILD USING requires Applesoft in ROM (Language cards are find), DOS 3.3 and a minimum of 32K . . . only **$30.00**

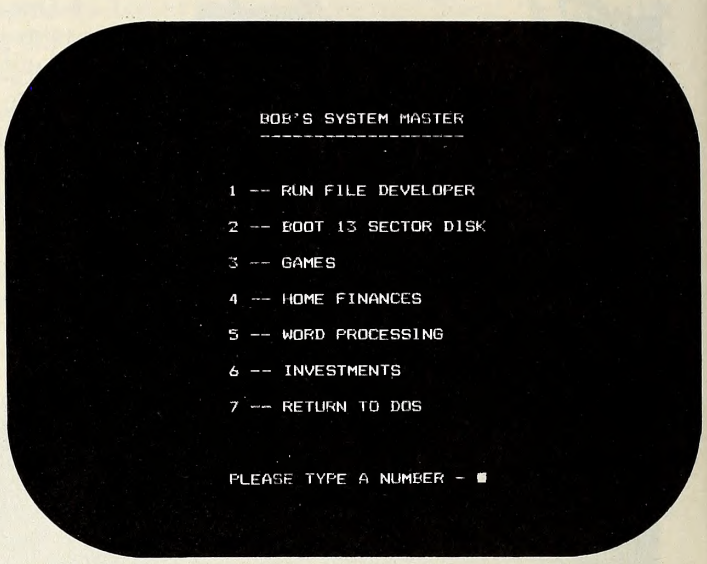
Sensible Software
6619 Perham Drive Dept. M
West Bloomfield, Michigan
48033 • (313) 399-8877
Visa and Mastercard Welcome
Please add $1.25 postage and handling per diskette

MAKE IT EASY ON YOURSELF.

Read how the unique, inexpensive MENU GENERATOR™ can organize all your program disks and save hours of non-productive work.



Just fill out a simple form like this . . . . . .



To create professional-looking menus like this.

Picture this:

You sit down at your Apple II and boot your own custom system disk.

A "menu" of numbered options appears. Perhaps one says "RUN FILE DEVELOPER." Another says "BOOT 13 SECTOR DISK." Other options you've defined might be "GAMES" and "HOME FINANCES."

Say you select "GAMES"— just by entering a single digit.

Immediately, another menu appears listing the games you own. If the one you select is on a different disk, the computer requests you to insert that disk and press a key to run it.

If this sounds like a convenient way to use your computer system, you're right.

If it also sounds like a lot of work to program, you're right again.

Until now.

Introducing the MENU GENERATOR.

Now creating your own custom menus is as simple as filling out a form on your computer screen.

Just run the MENU GENERATOR, and tell it what you want each menu option to say, and what action to take when that option is selected. You can RUN or BRUN a program, boot another disk, execute user—written BASIC statements, or perform any valid DOS operation.

MENU GENERATOR will compile your entries into a neat, attractively-formatted screen menu, *then automatically write a documented, bug-free Applesoft program to generate and process the menu.*

You can list, copy, and modify this program if you want to add your own touches. A single disk will hold data for up to 150 different menus.

Make it easy on others, too!

If you write programs that other people are going to use, you'll find that a few good menus are worth 100 pages of instructions. You'll be thankful, too, when you put the program aside for a couple of months and then try to use it again!

Pays for itself the first time you use it.

Using the MENU GENERATOR should save at least 95% of the time you'd spend writing your own menu program. At just $39.95, can you afford *not* to have this great labor saver working for you?

MENU GENERATOR comes complete with program disk, backup disk, and 40-page manual. It runs on a 48k Apple II Plus, under DOS 3.3.

Ask your dealer for MENU GENERATOR, or order direct from Crane Software for $39.95 (shipping included). Payable by VISA, Master Card, check, or COD. California residents add 6% sales tax.

Give your Apple a lift.

Crane Software, Inc.

Crane Software Inc.
16835 Algonquin, Suite 611
Huntington Beach, CA 92649
(714) 846-8005

DEALER INQUIRIES INVITED

Similarly, you need not include a parameter list in a call to ReadLn. If parameters are included, as in

```
ReadLn(A, B, C);
```

the call is identical to

```
BEGIN Read(A, B, C); ReadLn END;
```

Since you may deduce the behavior of ReadLn, called with parameters, by combining the behavior of Read with that of ReadLn, as called with no parameters, I will only spend time here discussing the "no-parameter" version of ReadLn.

Calling WriteLn with no parameters forces the display cursor to the beginning of the next display line. ReadLn, as you may expect, acts similarly, in forcing Pascal to ignore input until the beginning of the next input line. By convention, Apple Pascal recognizes a press of the return key as marking the end of a line of input data. Although Pascal translates the character generated by the return key into a blank, there is a special, built-in function (EOLN, which we'll examine next time) that permits a program to distinguish between regular blanks, and those that are generated by pressing return. ReadLn makes use of this function in fulfilling its sole purpose of accepting and discarding input characters until the return key has been pressed.

If you replace the call to Read in *Adder3* with a parameterized call to ReadLn, you will find that either a blank or return is sufficient to separate the input value for NumA from that for NumB, just as it was in the previous version of the program. However, only return is accepted by the new *Adder3* as terminating NumB. The effect of ReadLn is to require that the user press return after the last item in the parameter list (if any), and before program execution may continue.

Suppose you replace the "ReadLn (NumA, NumB);" in your latest version of *Adder3* with the two calls "ReadLn (NumA) ; ReadLn (NumB) ;". No longer will it be possible to enter both input numbers on the same line. If you try, the first call to ReadLn will acquire NumA, then discard all characters between NumA and the end of the line—including NumB— as if they were garbage! You will have to reenter NumB on the following line. (By the way, for the sake of aesthetics, I would put out an appropriate prompt prior to each of the two calls to ReadLn.)

As far as numeric input is concerned, the choice of whether to use Read or ReadLn in your programs is entirely up to you. It's a tricky business, trying to decide which is more appropriate in any given situation. In future excursions down the Path, I will try to present examples, in which both options are exercised, along with the rationales behind my particular choices. For data types other than numeric, the decision between Read and ReadLn is (usually) much more easily made, and you'll see examples of such situations soon.

More to Come. By now, I hope it's apparent to you that input is a very meaty subject. Consequently, I will be devoting the next several columns to explaining it as fully as I can. This time, we've only skimmed over Apple Pascal's most rudimentary input facilities; even so, that should be enough to occupy you in the month between this issue and the next. For homework, modify any of our old programs to accept input from the keyboard. This should help you to become familiar with the use of Read and ReadLn, as applied to the collection of numeric data. You'll need to be comfortable with their behavior in order to follow along as I dissect them and show you some of their inner workings. We'll also look at Files, the basic means of input and output in the Apple Pascal system, along with some of the primitive operations that Pascal provides for manipulating them (character by character input, for example). ◨

Be a Part of Our Success

SDS MOVES AHEAD

We at SDS have built our reputation by offering the highest quality utility software available, combined with personalized service and after-sale technical support. Our total involvement in the utility software area is further reflected in our president Roger Wagner's widely read monthly column, "Assembly Lines". Our thorough knowledge of the field, and our sense of commitment to all our customers have led to tremendous success in 1981.

So SDS is growing - from the creative vision of a single person, to one of the shining stars of the software industry. Whether you're an end user, dealer, or author, we want you to make SDS part of your success with the Apple II.

To make that happen, we're now offering a wider range of utility and other software, more customer service, better author benefits. The service is still personal and the products are still the finest anywhere - but now we can offer you more . . .

SDS southwestern data systems™

MORE NEW PROGRAMS

ROUTINE MACHINE – A terrific way to put machine language in your Applesoft programs. Simply choose from a pre-written library of subroutines. No knowledge of machine language is required.

AMPERSOFT PROGRAM LIBRARY – A set of additional machine language subroutines for use with Routine Machine.

MERLIN – A powerful, full-featured 6502 macro assembler.

ASCII EXPRESS: THE PROFESSIONAL A professional version of our best selling communication package that transfers files of ANY size or type. Supports all 80 column boards, interactive macros, and much more.

P-TERM: THE PROFESSIONAL – A Pascal version of Bill Blue's highly acclaimed communication software.

DOUBLETIME PRINTER – Increase your Apple's productivity by simultaneously printing text or binary files from one program while you run other programs.

UNIVERSAL GRAPHICS – The ultimate Hi-Res printing utility to transform your creations into permanent records on ANY graphic printer. Includes picture cropping, inverse inking, horizontal or vertical printouts, and a range of magnifications.

MORE DEALER SUPPORT

SDS is committed to the continuing support of the Apple dealer. As part of our new program, SDS offers:

- Informative point-of-sale materials
- Comprehensive national advertising
- Head Start Program to insure priority delivery of all new products
- Special dealer discount on in-store display items
- Comprehensive line of wide application software products
- Full-time technical support staff

SDS
southwestern
data systems™

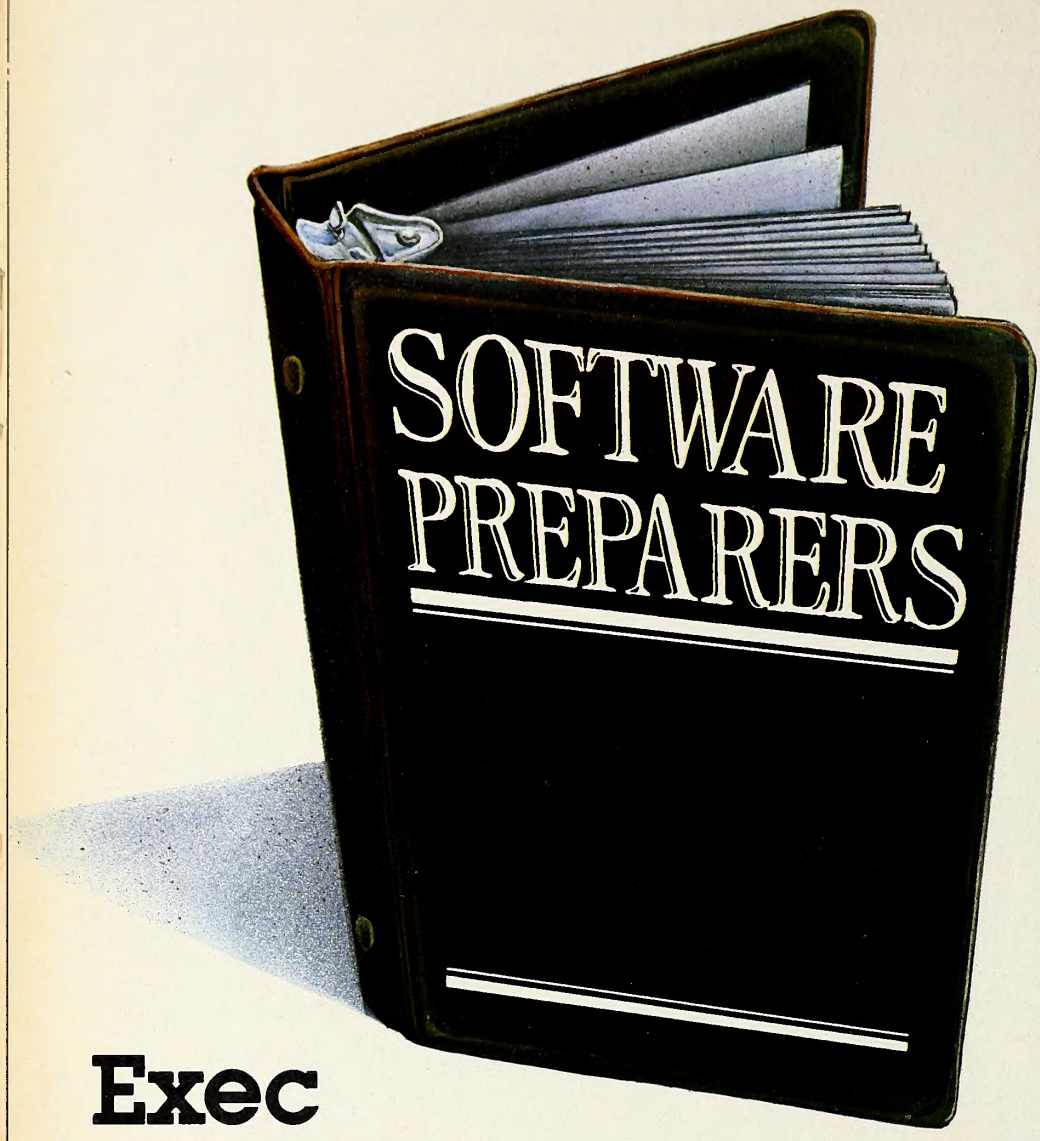
P.O. Box 582S Santee, CA 92071
Tel. 714-562-3670

MORE AUTHOR BENEFITS

SDS is an industry pioneer in innovative software for the Apple II. When you join the SDS team, you get the benefits of a professional support staff experienced in providing all the help you need to market your program. Other benefits include:

- Technical programming assistance
- Unique copy protection with limited backups
- Successful marketing strategies
- Assistance in writing the manual
- Professional product artwork
- Quality advertising
- Superior packaging
- National distribution
- Highest royalties paid monthly
- Customer service support

Here's the chance for you to market your program with the finest publisher in the software industry. Let Southwestern Data Systems' reputation and proven track record go to work for you. Be a part of our success – call today.

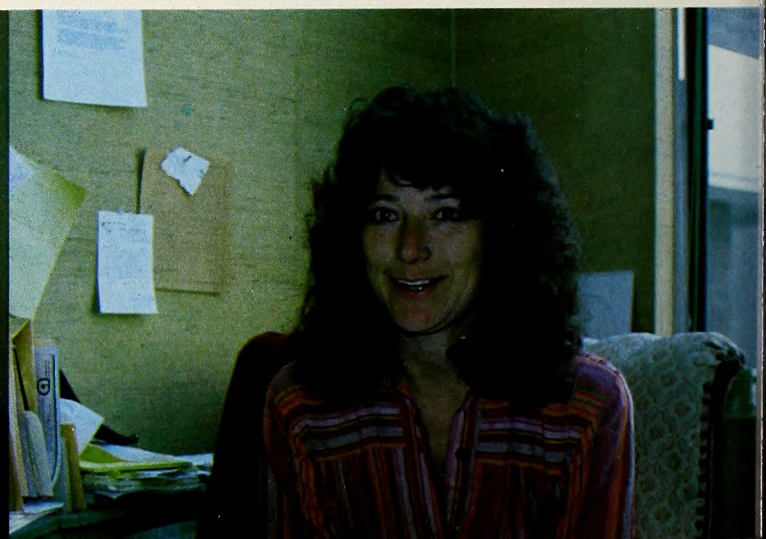


Exec

BY DAVID HUNTER

HSS

Howard Software Services



Jim Howard has always wanted to be an architect. The independence that goes with running a small business initially attracted a young Jim Howard to the idea of starting his own architecture firm. As the years passed, Howard left the possible career of architect for other things, but the spirit of his dream remains.

Girard Avenue curves crazily along the shore in La Jolla, California, ten or so miles north of San Diego. On one steep curve there is an odd blue and white building that overlooks the Pacific Ocean. With a wooden, lighthouselike, New England look, this structure has some decidedly unusual architectural features. The building's architect let his imagination soar and created a labor of love that is far removed from the blandness that characterizes much of modern architecture.

Howard Software recently moved into three offices on the third floor and one ground floor office in this building. The architect fashioned his own living space with a spectacular view and Howard Software is in the process of moving the marketing and accounting departments into this area. Walking through the two-story, studiolike space confirms the feeling that the designer must have been a little mad. The building definitely has the personal touch and it seems like the perfect habitat for the business of mild mannered James Howard.

Howard may not be an architect but he does know about a certain kind of madness; the kind that sweeps the nation once a year when for two or three months all hell breaks loose. Roaring mightily, the IRS gnashes its teeth and all good citizens cringe with fear. A warlike madness grips the country. Howard makes a great deal of his living off this madness.

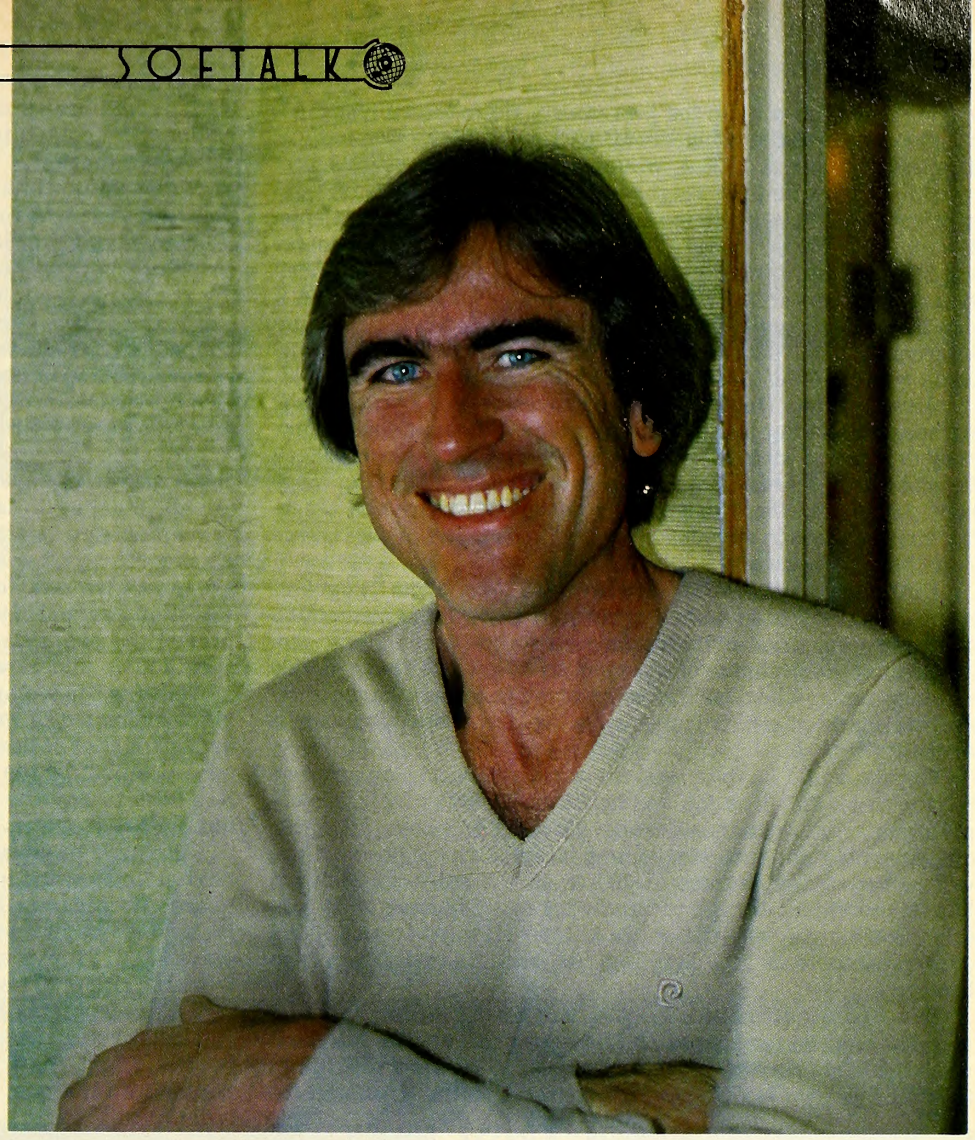
Excessive Success. Wars are won by

using well-thought-out tactics and strategies, but, like the British Army officer says in *The Bridge on the River Kwai*: "There's always the unexpected." The demand for *Tax Preparer*, by far HowardSoft's most popular program, nearly overwhelmed the company's resources this January. In one month, they shipped four thousand copies, the projected stock for the entire tax season.

"A lot of things happen by accident and we were unprepared for it." Jim Howard knew the danger that he faced if he couldn't ship product on time and keep up with the demand. "We were in for big problems with our reputation. People want tax programs right away." Heroically sacrificing physical health and peace of mind, Howard and his hard-working staff of fourteen managed to stay ahead of the avalanche. An unexpected disaster turned into an unexpected victory.

Life is a veil of unusual things, strange buildings, and unexpected occurrences. For instance, one is flabbergasted to discover that there are actually natives of Los Angeles. Well-known as a Mecca for sun starved easterners, Los Angeles is a place to go to, not come from. Defying the odds, two of Howard Software's top personnel are natives of Mickey Mouse's home town.

To Rise Above These Humble Beginnings. Jim Howard was born and raised in Los Angeles. For most of his early years he lived in the area near Venice and Western. Howard graduated from UCLA in 1964 with a bachelor's degree in



Abave: faunder and chief progrommer Jim Haward. Facing page: Left, executive director of morketing Dionne Prittie, Right, administrative caardinator Moril Sowell. Below: Top raw, Caral Garcia, Charlie Mack, John Coffee, Paul Hatamiya, Telv Ressurectian, Chris Tallefsan, Dan Brotemarkle, Bottam raw, Moril Sawell, Jim Haward, Dianne Prittie, and Vivion Pongle.

engineering and a taste for jazz music. He received his Ph.D. in electrical engineering from UCLA in 1969. Economics, Howard's minor at UCLA, would come in handy years later.

Howard worked for Hughes Aircraft from 1964 to 1967 designing radar and participating in several research projects. Sometime in 1976, Hughes asked Howard to return to school for his MBA. They wanted him to take over a management position. Doing research has a certain amount of freedom and it can be exciting and challenging, but management is pretty uninteresting stuff. Howard quit Hughes in 1977 to look for something more suited to his goals and lifestyle.

After leaving Hughes, Howard investigated small R & D companies looking for the kind of job he had at Hughes before they decided to kick him upstairs. Eventually he found Mark Resources in Marina del Rey, a twelve-man consulting firm. Howard worked there contentediy designing radars and teaching short classes in electronics until March 1981 when he quit to run Howard Software full-time.

Until three years ago, Howard had next to no programming experience and he has never had any formal instruction in computers. Once or twice he was forced to do some programming in the course of a project on a Hewlett-Packard 2100 minicomputer. The desire to be involved in all the phases of research made learning programming for these projects a necessity.

Apples would have to wait for the eighties as Howard was busy with his full-time job and hobbies that didn't include computers. For more than a decade, Howard has been doing his own investing and financial management. Call it a hobby if you will: after taking courses at UCLA on the subjects, Howard has felt confident enough to handle his own finances. This confidence built up over the years helped Howard to make the considerable leap into starting his own financial software business.

The Big Day. Howard bought an Apple in late 1979 thinking it would be good for his son and daughter to learn on. Maybe it was curiosity or maybe it was fate that led Howard to attempt to write his own tax-preparing program. Initially written for his personal use, Howard's program ended up a pretty useful device. He decided to try marketing it in the small but growing Apple market.

Working full-time during the day and moonlighting at night, that first year was tough, but Howard managed to survive. He shipped enough product to have a legitimate business and wrote off the Apple on his taxes as a business expense. Working out of his home in the L.A. neighborhood of Westchester, Howard shipped fifty tax preparers, with instructions paperclipped to the disk, in time for the 1980 tax deadline. Pleasantly surprised with the results, Howard began working on his own real estate program and for the next few months he didn't take the future of HowardSoft too seriously.

After the repeated urging of users and dealers, Howard decided maybe there was something in this software business after all. Starting in late 1980, Howard incorporated many improvements and additions into the 1981 *Tax Preparer*. After gathering research on how to market a software program, Howard got high quality binders to house the documentation and instruction manual. He also increased the advertising and improved the channels of distribution. The resulting sales were spectacular and Howard had a real business going.

Quitting his well-paying, full-time job in March 1981 was a big decision for Howard. A conservative person at heart, Howard spent a lot of time determining the wisdom of leaving a secure position for the uncertainty of heading his own business. Needless to say, he made the plunge and now he's glad he did.

Although he admits to always being afraid of taking the next step, Howard has had a pretty hectic life in the last few years. After leaving Mark Resources he moved down to La Jolla, a place he had visited many times previously and had always liked. "It was a dream come true to live down here, away from the smog." Dreams are never foolproof though, and Howard spent his first three months in dreamland packing boxes and answering phones.

The Right Staff. In June 1981, Howard realized that he couldn't do it alone. He went looking for someone to organize

SVA MAKING APPLES GROW

RIGHT PRICE – RIGHT DELIVERY

SVA originoted 8" floppy disk controllers for the Apple in 1979 bosed on 8 yeors of microcomputer consulting experience.

Since then we hove odded four new memory systems ond enhonced our system ond opplicotion softwore to bring you the utmost in performonce ond convenience.

SVA products odhere strictly to industry computer stondords.

SVA mokes your doto reodoble on other computers ond insures moximum reliobility ond tronsportobility by recording oll doto using stondord IBM formots.

Standord Apple DOS, CP/M ond Poscol ore used to bring you moximum flexibility, compotibility, ond continued support.

SVA hos delivered 2000 cords ond systems on time with the opplicotions ond ossistonce to support them!

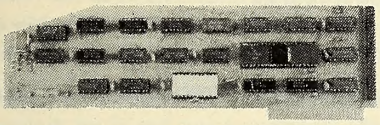
Relioble? You bet!

Every product is operoted for o full 7 doy burn in period ond retested before leoving our door. Top quolity components like Shugort floppy drives - the stondord of reliobility - ore used exclusively.

Support? Count on it!

Our opplicotion development group helps you instoll stondord DOS, CP/M or Poscol opplicotions. Coll the SVA Hotline for o list of opplicotion softwore thot runs on the MEGABYTER systems.

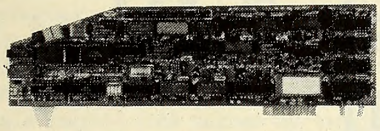
Look to SVA for o growing line of Apple Memory System products . . . SVA meons business.



DISK 2+2

8" Flappy Disk Cantraller
Single density - Single/double sided
Twice the Byte

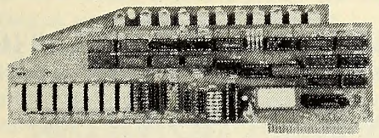
Intelligent cantraller daubles the disk starage capacity af the Apple. Existing CP/M, DOS ar Pascal saftware aperates cancurrently with mini disks. Laad and run CP/M and Pascal disks ar transpart/capy data disks af any camputer using the IBM 3740 farmat.



ZVX4 MEGABYTER

8" Flappy disk contraller
DUAL density - Single/dauble sided
Faur Times the Byte

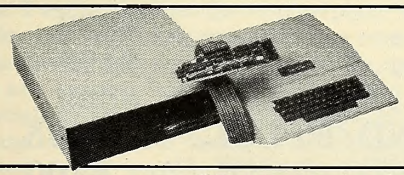
All the features af Disk 2+2 plus IBM System 34 dauble density recarding. Autamatic density switching. Cantral up ta 4 mega-bytes af anline data. Increase data security and reliability. File yaur tax return an 8" flappy disk. Excellent far hard disk backup.



APP-L-CACHE

256k RAM Memary Card
Memary with disk emulatar ROM
Mini disk cast - Hard disk perfarmance

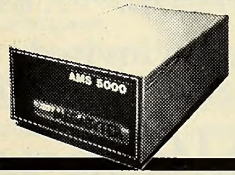
Operate as a standard 16k memary card and stare/retrieve data at 3-50 times mini disk speed. Eliminates the need far an extra mini disk. Uses standard DOS, CP/M and Pascal disk cammands. Access large data bases withaut mini disk delays.



AMS 8000 MEGABYTER SYSTEM

Dual 8" Flappy Disk Memary System
1/2 ta 4 Mega-bytes data standard starage
Remavable Online Memary

Complete single ar dauble density memary system far applicatians requiring large memary. Data base, Accaunting, Ward processing. Key ta disk, and DOS, CP/M ar Pascal pragram develapment. SVA has the suppart saftware far these and ather applicatians.



AMS 5000 MEGABYTER SYSTEM

5" Winchester hard disk system
5 to 20 Mega bytes af anline memary
Hard disk starage - SVA suppart

The latest in high speed - high density anline memary systems. Laad up yaur DOS, CP/M, ar Pascal applicatians and watch the brain pawer af the Apple graw. Cambine with an SVA 8" flappy system far data transpartability and backup.

MEGABYTER SOFTWARE
Dedicated ta standards and support
Application Hatline assistance
(714) 452-0101

Doto Tronsporter: Move doto between computers with DOS to IBM, CP/M to IBM, CP/M to DEC.

Generol Business: Support for: The Apple Controller*, Accounting II Plus*, and others.

Word processing: Magic Wond*, Apple writer*, Word Stor*, Poscol Editor.

System Softwore: Peorl opplicotion development, Apple DOS, Poscol ond CP/Ms.

Doto Bose: Desktop/Plon, DB Master, dBase II, CCA Doto monoger.

Disk Copier: Copy ony 8" IBM 3740 or System 34 formot disk.

*Trodemorks CP/M - Digital Reseorch DB Master - Stone Wore; dBase II Ashton-Tote.

SVA

SORRENTO VALLEY ASSOCIATES
11722 Sarrenta Valley Raad
San Diega, CA 92121-1084
(714) 452-0101 TWX 910-335-2047

the business and help with the daily grunge work. One of the first individuals he interviewed was Dianne Prittie, another displaced native of Los Angeles. Here it was a case of the potential employee badgering the employer. Dianne took one look at Howard's messy office/apartment and thought, this is the job for me. The morning after the interview she knocked on Howard's door and said: "I want this job."

Prittie had an interesting and varied life before she came to work for Howard Software. For three years in the mid-sixties she was tour coordinator for the Fifth Dimension rock group. It was crazy and fun; she never knew what to expect next. After the Fifth Dimension, Prittie managed a female artist for a year. In the early seventies she ran an antique store on Sunset Boulevard.

Then it was a stint as a real estate agent for a company in Beverly Hills. "I realized after a while that the real estate business was too consuming in every way. You have to work seven days a week and work nights. You have no private or personal life." Buying homes and property can be a traumatic experience for buyer and agent alike; Prittie found the whole thing all too stressful.

Good Vibes. As if on some cosmic wavelength with her future boss, Prittie moved down to La Jolla to escape the hectic lifestyle of Los Angeles. Like many people, then and now, she thought computers would be a good thing to look into and that is how she came to find Howard Software.

Originally hired on as a general, part-time flunky, Prittie has worked full-time since the second week on the job. Starting as administrative assistant, she is now executive director of marketing in another case of the unexpected. Howard had no idea that she would do so well with advertising and handling dealers. Hard working, intelligent, and industrious, Pritti has helped Howard Software maintain its well-respected spot in the Apple market.

Which brings us to the Howard Software advertising campaign, which should be familiar to *Softalk* readers who've received the last four or five issues. This series of ads featuring a beautiful young woman offering Howard software has garnered some negative response from some folks. Prittie is very concerned about this problem because it seems to be working at cross-purposes to her original concept of the campaign. An avowed feminist, Prittie never thought the ads were cheap or exploitative.

It is Prittie's desire to make an image for Howard that will be instantly recognizable. This is not limited to any single person or kind of person, and there will be different people showing up in future ads. Still, there will be human beings (of both sexes) in the ads, because Prittie believes this is the best way to sell Howard Software. "People buy from people. There is no reason why our ads must be only technical and serious."

Howard Software is not looking for the institutionalized look of big business. Jim Howard has a loose office dress code that specifically forbids ties. In their advertisements they want something to show the quality of their programs but at the same time keep it personal. "Accountants, attorneys, real estate brokers—they're just like me," Prittie says. "Technical data doesn't mean much; people want to know what the program does." Prittie is very conscious of HowardSoft's image, though sometimes it is possible to get caught up in the mechanics of a task and temporarily lose touch with the results.

Maril Sowell is an eminently likable woman who has taken on the administrative coordinator position. Previously a counselor at a rape crisis center in San Diego, Maril felt her future there to be limited. Looking for something with greater opportunity in a time of economic recession, she decided to investigate the booming field of microcomputers.

Thankfully, Sowell was used to handling a multitude of tasks at the crisis center. She had her hands full as soon as she joined Howard Software. Jim and Dianne headed up to Comdex in Las Vegas and left Maril to handle the move into the Girard Avenue offices. Displaying talents no one knew she possessed, Sowell took care of all the interior decoration and coordinated the layout of the offices.

Howard is always pleased when he finds someone as hard-working and industrious as Maril Sowell. She is very happy at

HowardSoft and has great plans for making the business tighter and better organized.

Despoiler of Youth. Howard has recently hired a few programmers to help with updating existing programs. Offering who knows what, he has convinced poor innocent college students from the local university to give up their studies and come work for him. Don Brotemarkle has a BA in physics from the University of California at San Diego and he was working on his Masters in abstract physics. That was before he met Howard. Now Brotemarkle is happily programming for Howard and seems to have lost interest in his previous career goals.

Howard claims to hire employees on intuition, recognizing the right person by some intuitive process not usually associated with sound business practices. Paul Hatimaya and Steve Templeton are two more programmers seduced away from the hallowed halls of UCSD. Like Prittie when she started, these fellows may be doing something in a few months they can only guess at now. Howard looks for intelligent and capable individuals who are not afraid of branching out into other things.

Everybody at Howard Software works more than forty hours a week if the extra time is demanded. Hours are flexible and some programmers work at home because computers are scarce during normal working hours. Nobody complains much because they have a good attitude about what they're doing.

Howard Software has three main programs to offer at this point — *Creative Financing, Tax Preparer,* and *Real Estate Analyzer*—but more may be on the way. Jim Howard thinks the time may be right to release a tax planning program. In the Howard tradition it would be cheap in price but sophisticated in performance.

These two factors—low pricing and a high amount of sophistication—have made Howard the top name in financial software for the Apple today. There are programs available that match the sophistication found in *Tax Preparer*, but they usually run around a thousand dollars. Similarly, few programs as inexpensive as *Tax Preparer* have the sophisticated features included in Howard's package.

Although there have been some encroachments into Howard's market, he is not too concerned. The head start HowardSoft has over the competition should keep them ahead as long as the programs keep getting updated and improved.

Making sure Howard Software's product gets there in one piece and familiarizing dealers with the features of that product is another of Dianne Prittie's areas of expertise. Prittie is planning a lengthy trek across the length and breadth of California to visit dealers and solidify the relationship that Howard Software considers so important.

Selling only through dealers has helped the customer, and HowardSoft is helped in turn by keeping close to user and dealer feedback. Believing that a better product sells better, Howard is constantly updating and expanding the scope of his programs. Feedback from the marketplace is very much responsible for the improvements that are implemented.

The desire to meet the needs of the user doesn't end here. Plans are afoot to hold local seminars in the San Diego area, sponsored by HowardSoft for business professionals. Accountants and real estate brokers will benefit from personal instruction on the ways and means of using *Tax Preparer* and *Real Estate Analyzer.*

One thing that caught Howard by surprise is the demand for his products by professionals. He figures that the majority of sales for *Tax Preparer* and *Real Estate Analyzer* are attributable to serious business people. About half of *Tax Preparer*'s sales are accounted for by what Howard calls the "heavy professionals" who process up to five hundred tax forms a year. Apparently the alternatives for professionals can get fairly expensive. They have to send out the work to large computer firms, who traditionally charge arms and legs.

Individuals account for a third of the sales of *Tax Preparer,* and most of these occur in the months of March and April. Better late than never, as the saying goes. After exhausting his stock in January, Howard is set for the rush of orders from individuals which will hopefully not be quite so intense. One reason that some individual users may not be rushing out and buying Howard's program is the price. Although it's cheap for a professional or a company, most individuals are used to sending out their taxes to a preparer for seventy to one hundred dollars. A hundred and fifty dollars for a program might seem too much to pay.

With HowardSoft's current update policy, buying a *Tax Preparer* for a single person or family may seem a little more reasonable. For a fee of about thirty dollars, you can exchange last year's program for the new one with the current tax laws. Over the space of a few years, Howard's package will pay for itself in savings over the cost of sending out your taxes each year to the tune of a hundred bucks.

Doing One Thing Well. Howard plans to stay with products that are unique in the marketplace. He wants to explore his particular expertise—financial software. There will be no games or word processors mysteriously emerging from Girard Avenue.

Howard likes to do everything, but at the same time not labor too long on one project. "I'm enjoying what I'm doing and I hope that everyone around me does the same." The sole owner of Howard Software, Jim Howard is enjoying the prospect of incorporating sometime this year.

Prittie is very happy with her job and feels the future bodes well. Dianne enjoys talking to people because she learns so much. "I felt intimidated at first, not being a programmer or someone familiar with the industry. But there are many people in the same boat. I've made so many friends and that makes it worthwhile."

Without a doubt, Howard is on the right track. An architect of programs as opposed to buildings, he has a small business that he intends to keep small. Carving a niche in the Apple landscape, Howard Software may be small but it is gaining a big reputation. Small is beautiful for some things, but respect is something you can never amass to excess. The folks at Howard Software have earned theirs and they plan to keep on earning it in the years ahead. ◨

HARD TALK

BY JEFFREY MAZUR

Apple deserves a lot of credit for the forethought that went into the design of the Apple II. Aside from the eight peripheral slots, one of Apple's best innovations is its game I/O connector. The game I/O provides a dozen analog and digital interface lines. Primarily designed for hand controllers to permit ball-and-paddle type games, this port has proven valuable for many other purposes. A wealth of paddle and joystick type controllers and other peripherals interface through the game I/O port.

Many of the game-related peripherals were evaluated by a panel of avid game players; the results of their evaluations follow.

Paddles. The basic gaming accessory is a single paddle or a pair of them. A paddle usually consists of a variable resistance control (the knob you turn back and forth) and a simple switch closure (the button). These two devices can provide sufficient information for almost all games.

In evaluating our test paddles, we divided each unit into four parts: case, buttons, control knobs, and cable. In judging the case, we considered both appearance and comfort. Comfort was rated according to how easy it would be to hold the paddles for an extended length of time. This was a reflection on the size, weight, and shape of the paddle case. Button and paddle controls were not considered in this rating.

We then proceeded to evaluate the buttons both for their mechanical feel and their apparent electrical reliability. The latter specification was quite subjective, but is, nonetheless, very important, since low-quality buttons can be worn out very quickly under heavy use. The position and feel of the button's plunger was also rated, using the program *Raster Blaster* as a "test course." This program was chosen because it uses only the two push buttons to play. If the buttons are not easy to work, this game will bring on sore fingers or thumbs very quickly.

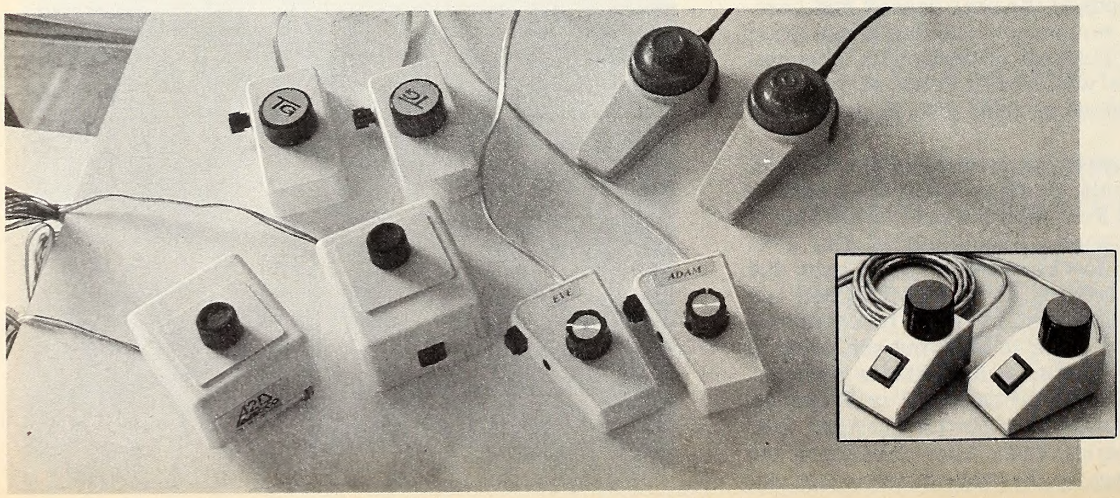
Another consideration on button ratings was whether any form of feedback was incorporated to verify that the button had been pressed. Some switches employ *tactile feedback* which provides a physical impulse to the finger as the button is pressed. *Audible feedback* is also possible and consists of a clicking sound to indicate switch closure.

In each case, the paddle control itself was basically evaluated on its size and shape; these qualities determined its ease of operation. Sirius Software's *Autobahn* was used to test paddle-only operation, and *Space Eggs* was used for evaluating paddle/button combination use. Each unit was also disassembled to determine the quality of the components used. Finally, the length, flexibility, and ruggedness of the cable and dip connector were considered.

Adam and Eve. The Adam and Eve paddles from Tech Designs received the highest overall rating of all paddles tested. The trapezoidal-shaped case is made of high-impact plastic and is extremely comfortable to hold in either hand. The identity of PDL(0) and PDL(1) is denoted by the Adam and Eve labels on each unit. High quality parts are used throughout, including a heavy-duty, square push button with tactile and soft audible feedback. The switch showed very good action in the *Raster Blaster* test, giving positive response and causing little fatigue.

The paddle knob on the Adam and Eve paddles is about one-inch in diameter and easy to maneuver. These paddles were also the only ones to have a fine-tuning trimmer control to match the paddles' range to any given computer or game program. This trimmer proved very useful in the *Autobahn* and *Space Eggs* tests. Like many other games, these programs respond to only the first half of the paddle's rotation (that is, from 7 o'clock to 12 o'clock). Turning a paddle knob any further clockwise has no effect on the game piece, as the knob is already at its rightmost position. This can by annoying if you overshoot the limiting point and then have to back it off slowly counter-clockwise until the piece begins to move again. With proper adjustment of the trimmers on these paddles, the motion of the game pieces can be spread out across the entire range of the paddle control.

Exiting the back of each paddle is a very flexible, five-foot long cable, similar to the type used by the telephone company for connecting the new modular phones; this should give some indication as to its reliability. The two paddle cables terminate in an oversize dip connector that makes the cable extremely easy to insert and remove. Also mounted on the cable is a strain relief to isolate the connector from excessive tugging. All in all, these paddles seem to be made of exceptionally high-quality components, well designed to be used comfort-



Poddles, clockwise from upper left: TG Poddles, TG Products; Hand Controllers, Keybaord Compony; (inset) Pro Poddles, Computer Works/ Roinbaw Computing; Adam ond Eve, Tech Designs; Model 2002, A2D Company.

ably for long periods of time, and built to last. With a six-month guarantee, the Adam and Eve paddles sell for $39.95. Also available is the Adam paddle, a single unit with two switches, for $29.95.

Apple Computer Paddles. Paddles came with the Apple computer prior to 1981. Complying with the new FCC regulations for computers at that time, Apple made several design changes to reduce interference with other RF devices such as radios and television sets. One of the major problems in this regard was the game paddles, so Apple decided simply to remove them from the package. They are now sold separately for $29.95.

Apple has had several versions of their paddles in terms of component arrangement. The apparently current type has a rectangular-shaped casing with the switch on the right side and a four-foot cable.

The major problem with the paddles is that they employ a very inexpensive switch that can't take much vigorous use. Also, the plunger is very small and, with heavy game playing, can make quite a (sore) impression on the fingers. Another problem with the newer version is that the placement of the switch is less favorable for right-handed users. All our right-handed testers found it most comfortable to hold the paddle in the left hand so they could operate the paddle knob with their right hand. With the switch on the right, it had to be pressed with one of the fingers, but given a choice, users would have preferred to operate buttons with their thumb. Holding the paddle in the right hand made it more comfortable to operate the switch, but, this required using the left hand for the paddle, resulting in greater strain; holding the paddle upside-down worked for the button but sacrificed a good hold.

A2D Company Model 2002. The Model 2002 game paddles consist of a rather large case (3.5 inches by 3 inches by 2 inches) with a small (less than 1-inch diameter) knob on top and a square switch on the back. These paddles were the least comfortable to hold and fairly awkward to play with. The quality of the parts, however, was quite good. In fact, the Model 2002 switches received the highest rating in the *Raster Blaster* test. They offer both tactile and a soft audible feedback and should probably live up to the manufacturer's claims of five million presses.

Another plus for the A2D paddles is the extra long eight-foot cable (no strain relief is provided, so it might be wise to tie the cable in a knot about four inches from the connector). There was nothing wrong with the operation of these paddles; their overall size and the placement of the switch account for the low overall rating. No identification between the paddles is provided, so you'll probably want to add some markings of your own. Price of the Model 2002 paddles is $34.95.

Pro Paddles. The Pro Paddles are manufactured by Computer Works and distributed by Rainbow Computing. Although it is not apparent from their advertisements, these paddles are the smallest we've seen. Measuring only 2.5 inches by 1.5 inches by 1.5 inches, they have a large 1-inch paddle knob which dwarfs the front of the unit. Although they felt comfortable to hold, they were almost too small, and sometimes became annoying to operate (especially when the fingers of the right hand, operating the paddle control, would rub against the left hand which was holding the unit). The placement of the switch on the back was also less than optimum, requiring the use of the (usually cramped) index finger to operate.

Identification of the paddles is provided by small numbers "0" and "1" engraved into the switch plungers. These plungers also had quite a bit of "play" (sideways movement), which was noted as somewhat distracting. Also unusual for paddles was the cable coming out from the front of the unit. The cable is very flexible and shielded to reduce possible interference with radios, TVs, and other devices. At less than four feet in length, however, it was considered a little too short. Price of the Pro Paddles is $49.95.

TG Products Game Controllers. Getting back to a more normal size, the TG paddles fit comfortably in the hand. The two paddles are identified by a red logo on one knob and black one on the other. The knobs are almost 1½ inches in diameter and easy to control. The buttons are on the left side, with large round plungers. Unfortunately, these switches do not appear to be extremely rugged, provide no feedback, and consistently provided less action in the *Raster Blaster* test. They seemed to require more force to press and therefore fatigued the players faster. Actually, the force required to close the switch is not greater, but with the lack of tactile feedback, our players felt compelled to press the button harder until it bottomed out against a firm spring.

The five-foot, flat ribbon cable and the connector received average ratings. Price of the TG paddles is $39.95.

The Keyboard Company Hand Controllers. These paddles have several unique features and were rated second overall. They are made of durable plastic in precise Apple colors. The case matches the Apple's tan color perfectly, while the paddle knob is the exact shade of grey found on the Apple keyboard. The only new color is the bright orange switch on the right side of the paddle. The placement of this switch was the only shortcoming found in these paddles, although it was certainly much easier to operate than the buttons on the Apple paddles.

If you are left-handed or if you don't mind operating the switch with your fingers, these paddles have a lot to offer. First, there's the extra large plunger that activates the switch. Both tactile and moderately loud audible feedback are produced when the switch is pressed. The paddles have a volcano-shaped knob that allows you to grab them along either the two-inch outside diameter or the one-inch inner knob. Molded in the center of each knob is the identifying 0 or 1.

Continuing on the positive side, there's the five-foot shielded cable that includes a unique strain relief that slides into one of the slots on the back of the Apple. Also included is a ground connection to the Apple's case (assuming you have one of the newer FCC Class B approved Apples). In fact, these were the only paddles tested that included a notice of certification by the FCC. The Keyboard Company Hand Controllers sell for $29.95.

Joysticks. When single-axis paddles cannot provide the two-

Joysticks, left to right: Keyboard Company; TG Products; A2D Company.

dimensional control necessary for some programs, a joystick is used. Basically, a joystick is nothing more than two paddles controlled by one shaft. One control senses movement in the vertical direction; the other in the horizontal. Some joysticks have springs that return the shaft to the center position. Joysticks are also used with graphics programs to create images on the screen. We tested the following joysticks with such a program, checking each stick's feel and accuracy. Most joysticks also have switch controls which were evaluated as was done previously with the paddles.

A2D Joystick Model 2001. Like their paddles, the A2D joystick is a bit too large to hold comfortably in the average hand. The switches are again on the back which also was somewhat awkward. Since it is more likely that a joystick may be resting on a table when operated rather than being held in the hand, this was not considered as much of a disadvantage in their joystick as in their paddles.

From a strictly hardware viewpoint, however, this unit contains top quality components. The joystick employs an "open gimbal design" such as is found on radio-controlled model airplanes. It has a self-centering action but this can be defeated by removing two springs. This is accomplished by removing first the bottom of the case and then the back of the joystick assembly (four screws each). The springs are then carefully lifted out, along with a small plastic cam that falls loose. These parts should be stored in a safe place (taped to the inside of the case, for instance), in the likely event that you may want to replace them.

With the springs removed, the joystick has a very smooth precise action. The two mechanical trimming devices are useful for setting the center points when the self-centering action is employed; with the springs removed, they act somewhat to center the active portion of the stick's movement. A small amount of overtravel occurs at the extreme positions of the stick (the computer reaches its limit before the physical stop on the stick), but this is not objectionable and the stick's output is very linear. The model 2001 connects to the Apple via a seven-foot round cable. Price is $44.95.

TG Joystick. The TG Joystick is also sef-centering, with two trimmer levers to calibrate the center position. This feature can be disabled by removing eight screws and two springs. These springs are very easy to reach for removal or installation. As for precision, thejoystick moves quite smoothly and is matched almost perfectly to the Apple's range. There was virtually no overtravel.

This unit also contains two large buttons just to the left of the joystick. These switches are identical to the ones found on the TG Paddles and are therefore subject to the same critique. A four-foot ribbon cable connects the unit to the Apple; this cable carries all sixteen pins of the game port into the joystick box, thereby making it possible to customize the interface if desired. The TG Joystick goes for $59.95.

The Keyboard Company Joystick II. Looking as if it were made by Apple Computer themselves, this joystick contains two switches. One is a small square pushbutton for switch 0. The other is a toggle switch that controls input #1. This switch has three positions. When pushed to the left, the switch latches on; when pushed to the right, the switch makes momentary contact and then springs back to the middle position. This may have some advantage for certain programs, but with our *Raster Blaster* test it was clearly a drawback. The joystick operation was slightly less precise than that of the previous units, but this was not deemed to be too significant. The Keyboard Company certainly deserves credit for styling their products to be a perfect match for the Apple computer. The joystick sells for $49.95

Other Game Port Devices. The convenience of the game I/O port has not been overlooked by other nongame related devices. There are light pens, copy protection devices, shift key modifications, and even modems that connect through the game port. Because this port contains four digital outputs, three digital and four analog inputs, plus power and a utility strobe, an enormous range of products can be interfaced through it.

Symtec BSR X-10 Control. If you have considered using the Apple to control lights and appliances within your home, this product may interest you. If you connect the small ultrasonic transducer (that is, speaker) to the game port, you make it possible for the computer to send command signals to the BSR X-10 series command console. The X-10 system is a widely available line of products designed specifically for remote control of AC-powered devices. The Symtec interface comes with software on cassette that allows the Apple to duplicate the command functions available on the BSR ultrasonic remote (that is, turn on, off, dim, or brighten one of sixteen remote devices, or turn all on and off). The addition of a clock/calendar card could make the Apple a powerful controller for the home.

Memory Expansion for Apple®

The company that brought you the first 32K RAM board for Apple II® and Apple II+® now offers:

VC-EXPAND/80™

NEW!

80 column VisiCalc® display on an Apple II !!

Now in addition to greatly expanding your workspace you can add 80 column capability to Personal Software's16 sector VisiCalc®. Works with Videx 80 column card. Previous owners of VC-EXPAND™ can upgrade to VC-EXPAND/80™ for $25.

ONLY **$125**

VC-EXPAND™

MEMORY EXPANSION FOR VisiCalc®

Expand memory available to Personal Software's 16 sector VisiCalc®. Add 32K, 64K, or even 128K to your present workspace (even if you already have a 16K card in use!) with this program plus one or more Saturn boards. Simple operation.

ONLY **$100**

■ 128K RAM

ALL FOR ONLY

$599

Our newest product. Fully compatible with Saturn's 32K RAM board, 16K RAM cards and language card.

Includes 5 comprehensive software packages:

1. MOVEDOS (relocates DOS)
2. RAMEXPAND (for Applesoft®, Integer®)
3. PSEUDO-DISK for DOS 3.3 or 3.2
4. PSEUDO-DISK for CP/M®
5. PSEUDO-DISK for PASCAL

■ 64K RAM

$425

A medium range memory expansion board which can be upgraded to 128K at a later date. (Upgrade kit sold for $175) Includes all 5 software packages offered with the 128K board.

■ 32K RAM

STILL ONLY

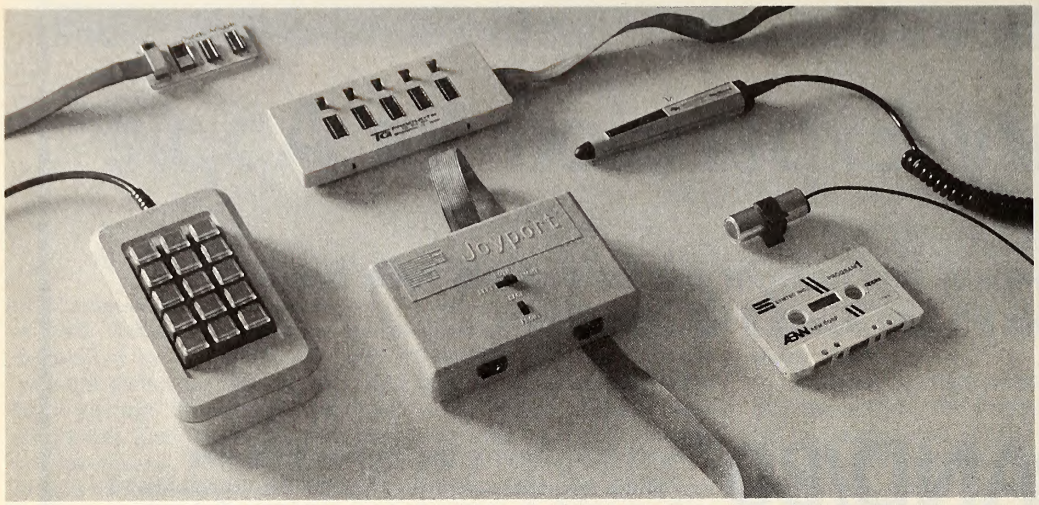
$239

The old favorite for Apple users. Includes our first 3 software packages (above) with CP/M® and PASCAL pseudo-disks now offered as options ($39 each).

SATURN SYSTEMS INC

(313) 973-8422

P.O. Box 8050, Ann Arbor, MI 48107



Special game part devices, left to right, top: Poddle Adopple, Southern Colifornio Research Group; Select-A-Part, TG Products; BorWond, Advonced Business Technology. Bottom: SoftKey, Advonced Business Technology; Joyport, Sirius Saftwore; BSR-X10 ond progrom cossette, Symtec.

The next step would be to add appropriate analog and digital inputs such as temperature, precipitation, switch closures, and so on to make the system truly intelligent. This would mean that you could do such things as having the sprinkling system automatically skip a cycle if it has been raining, or design an efficient lighting system for your home. This device sells for $49.95.

ABT Softkey. The Softkey from Advanced Business Technology is an auxiliary keypad containing fifteen user-definable switches. They are housed in a sturdy Noryl injection-molded plastic case that is styled and colored to match the Apple. Each key has a removable plastic top under which a small identification label can be placed. A pre-printed sheet of popular labels is also included.

Operation of the Softkey requires a machine-language driver routine that is linked into the Apple through the normal input vector. Thus, output from the Softkey is handled just as if it were typed on the keyboard. Each key can be assigned the equivalent of up to eleven normal keystrokes (for example, one key could be programmed to type *catalog* so that pressing this single key would cause the directory to be displayed).

Along with the driver is a table of key definitions that determines what character or string of characters will be "typed" for each Softkey. These definitions can be created using the supplied program or one of several standard sets can be used. For example, complete instructions are provided on how to modify Apple's word processing program, *Apple Writer*. After you've made the appropriate patches (done by simply typing *exec tedmod*), the supplied key labels can be inserted. Now all of the cursor movement and various other commands can be accessed at either keyboard. However, functions that normally require one or more *escape* and/or control keystrokes can now be activated by pressing a single button on the Softkey. Figure 1 shows the layout and definition of the keys when used with *Apple Writer*. The only drawback encountered with this modification is that it may not be compatible with other patches such as for a lower-case input device. The Softkey sells for $150. Pascal software is also available for $50.

ABT BarWand. Another interesting device from Advanced Business Technology is their BarWand. About the size of a large pen, this wand is used to read various types of printed digital bar codes, such as the UPC labels found on most products. Inside the wand are a small LED light source and a light sensor. As the wand is moved over a label, light reflected from the light and dark bars is registered by the computer. By comparing the reflected pattern against a known code, you can identify an item.

Many different codes can be read by the BarWand with the appropriate software; included in the package are routines for reading UPC, ABT's LabelCode IV, and Paperbyte codes. This last code is used for distributing programs in a machine-readable printed form. This makes it feasible, for instance, to include program listings in a magazine article.

When using the BarWand, it is important to move the wand at a relatively constant speed. This takes a little practice, but soon it becomes easy to read most labels with one pass. If a label is read incorrectly, it will not register; valid data is usually signaled by a beep from the Apple speaker. Variations in the scanning speed and direction (most codes can be read in either direction) are compensated for by the software driver routine. Furthermore, a switch control on the side of the wand that is activated with the thumb allows you to activate the wand only when needed so that extraneous signals are not produced when the wand is not scanning.

Various other packages are available to work with the BarWand. Some of these allow the wand to read other codes such as LabelCode V, Code 39, Codabar, and Two-of-Five. These codes are widely used in industry, each with its own advantages. ABT also has software for printing out these codes on many of the popular graphics printers. All of the read and print software is available for use with either Basic or Pascal. The BarWand sells for $195. Optional read or print packages are $75 each.

Paddle-Adapple. With all of the paddles, joysticks, light pens, protection keys, and other devices to select from, it can become quite tedious to plug each device in and out as needed. This is especially true if you keep disk drives, a monitor, or anything else on top of your computer. Fortunately, several devices on the market now make this task much easier. One such device is the Paddle-Adapple from Southern California Research Group. This small device attaches to the outside of the Apple with double-sided adhesive foam. A short ribbon cable connects the unit to the game I/O socket in the computer.

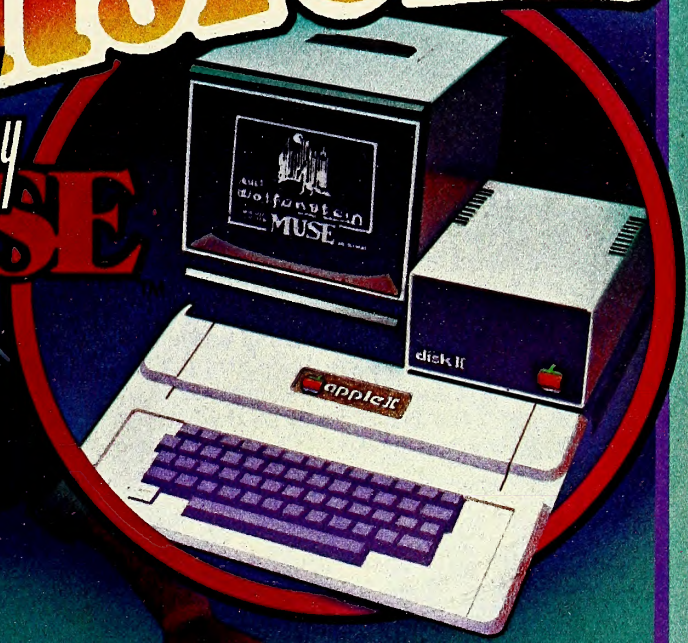
The Paddle-Adapple has two sockets that accommodate a pair of game I/O accessories. A switch can be used to select between devices. The real power of this unit is that it has a configuration socket that provides a great deal of flexibility in setting up various devices. For example, it is possible to reverse the X and Y axis of a joystick simply by moving two jumpers. Another unique feature of this device is that it includes a cable which can be plugged into the cassette jack to add one more switch input to the three available at the game I/O socket. This makes it possible to use four game paddles, each with its own switch. Of course, the software being used must be written to take advantage of the extra switch input. The only complaint we had was that the connecting ribbon cable was too short to allow us to mount the unit toward the front of the computer. At

FBEG	^ 12	CAPS
ESC	UP	MEM
LF	+	RT
<–	DN	–>
MENU	!12	FEND

Figure 1. Softkey layout when used with Apple Writer.

Castle Wolfenstein™

by MUSE



THE MOST INTERACTIVE GAME EVER!

The wildest, most addictive adventure game to date–a genuine merger of fantasy and real-time arcade elements.

"A virtuoso feat of programming..." –*New York Magazine*

For the Apple II or II Plus, 48K. $29.95.
By Silas Warner.

™MUSE

347 N. Charles Street
Baltimore, MD 21201 (301) 659-7212

Call or write for information and the name of your nearest MUSE dealer. Apple is a TM of Apple Computer Corp.

WALDREP

$29.95, the Paddle-Adapple makes an excellent investment for the serious game player.

Select-a-Port. If you have more than two game port devices, you may be interested to know about Select-a-Port, a five-socket expansion device from TG Products. The long, thin case comes with little "hangers" that enable you to mount it on the side of the computer. Or it can just sit on the table, resting on its four rubber feet. The connecting cable is long enough that you can place the device wherever you want.

Above each socket is a switch to control which devices are active. This switching is accomplished using diodes to connect or isolate each device from the computer's control lines. Any number of sockets can be used simultaneously, with two exceptions. The first socket is always active, and is in fact completely in parallel with the Apple port (the switch above this socket is not used). This arrangement ensures compatibility with special devices such as copy protection "keys" that might not function properly through the diode switching. Of course, no more than one paddle control should be connected at any time to a given paddle input. The other exception is that the paddle controls of socket #2 are connected to the third and fourth paddle inputs of the computer. This, of course, allows four-player games using two sets of paddles or dual joysticks (only two switch inputs can be used, however). This still leaves three completely independent sockets for your other devices. Unless you are writing an article on game I/O devices, this many should be more than adequate. The Select-a-Port was invaluable in running our tests on the various paddles and joysticks; it retails for $59.95.

Next Month. As you can see, there's a lot of hardware to choose for your game I/O. There probably are several other paddles and joysticks available in addition to those reviewed here, and more devices are likely to appear in the future. Some new items have also recently appeared to eliminate the problems of using the Dip connectors that are too fragile for constant plugging in and out.

To add even further to the possibilities, there are two new products that will greatly expand the usefulness of the game I/O port. One of these is the Joyport from Sirius Software. This unit allows selection between one or two Apple-type paddle/joysticks or up to two Atari-type joysticks. Atari joysticks are not true analog controls, but instead provide switch inputs whenever a lever is moved forward, back, left, or right. This requires additional hardware (in the Joyport) as well as software to convert this information to analog motion.

Another exciting new product is the Freedom wireless joystick. As its name implies, this joystick gives you the freedom to move around with your control unhindered by the usual connecting cable. The Freedom and Joyport will be described in detail next month. ◨■

Advanced Business Technology, Inc., 12333 Saratoga-Sunnyvale Road, Saratoga, CA 95070; (408) 446-2013. Apple Computer, 10260 Bandley Drive, Cupertino, CA 95014; (408) 973-3429. A2D Company, P.O. Box 6471, Greenville, SC 29606. Rainbow Computing, (Distributors of Computer Works' Pro-Paddle), 19517 Business Center Drive, Northridge, CA 91324; (213) 349-5560. Tech Designs, 3638 Grosvenor Drive, Ellicott City, MD 21403. TG Products, P.O. Box 2981, Richardson, TX 75080; (214) 424-8568. The Keyboard Company, 7151 Patterson Drive, Garden Grove, CA 92641; (714) 891-5851. Sirius, 10364 Rockingham Drive, Sacramento, CA 95827; (916) 920-1195. Symtec Inc., P.O. Box 462, Farmington, MI 48024; (313) 272-2950.

Editor's Note: Jeff Mazur praises the Adam and Eve paddles highly. His company also distributes them. But in this case, evaluation and praise preceded distribution.

While Mazur was writing this article, several people contacted him to express interest in purchasing the Adam and Eve paddles. When he learned that the paddles were distributed almost exclusively in the East, Mazur arranged to distribute the product through his company. His decision was based on his conviction, derived during his researching of this article, that the Adam and Eve paddles were an excellent product. MCT

We have ways...

to make it talk!

Give your computer the magic of speech with an ECHO™ Speech Synthesizer!

There are four new models to choose from: the ECHO 80 (TRS-80 models I & III), ECHO 100 (S-100), the ECHO GP (general-purpose, serial/parallel) and the ECHO SBC (single board computer). These join the popular ECHO][(Apple).

ECHO Speech Synthesizers use LPC synthesis pioneered by Texas Instruments, combined with a phoneme-based operating system to provide any vocabulary you desire while using a minimum of your computer's memory.

The TEXTALKER Speech Generator, standard with all ECHOs except the SBC, translates normal English text to speech automatically.

Simple commands allow you to select from:

- 63 different pitch levels
- Words spoken monotonically or with intonation
- Fast or slow speech output
- Entire words pronounced or spelled letter by letter
- Different volumes
- Spoken punctuation if desired

Speech applications are virtually unlimited, ranging from education to games to aiding the handicapped. The flexibility and low price of ECHO Speech Synthesizers make them the logical choice when adding speech to your system.

For more information see your dealer or contact:

SEC **STREET ELECTRONICS CORPORATION**
3152 E. La Palma Avenue, Suite D
Anaheim, CA 92806 Telephone: 714/632-9950

Electric Duet

insoft®

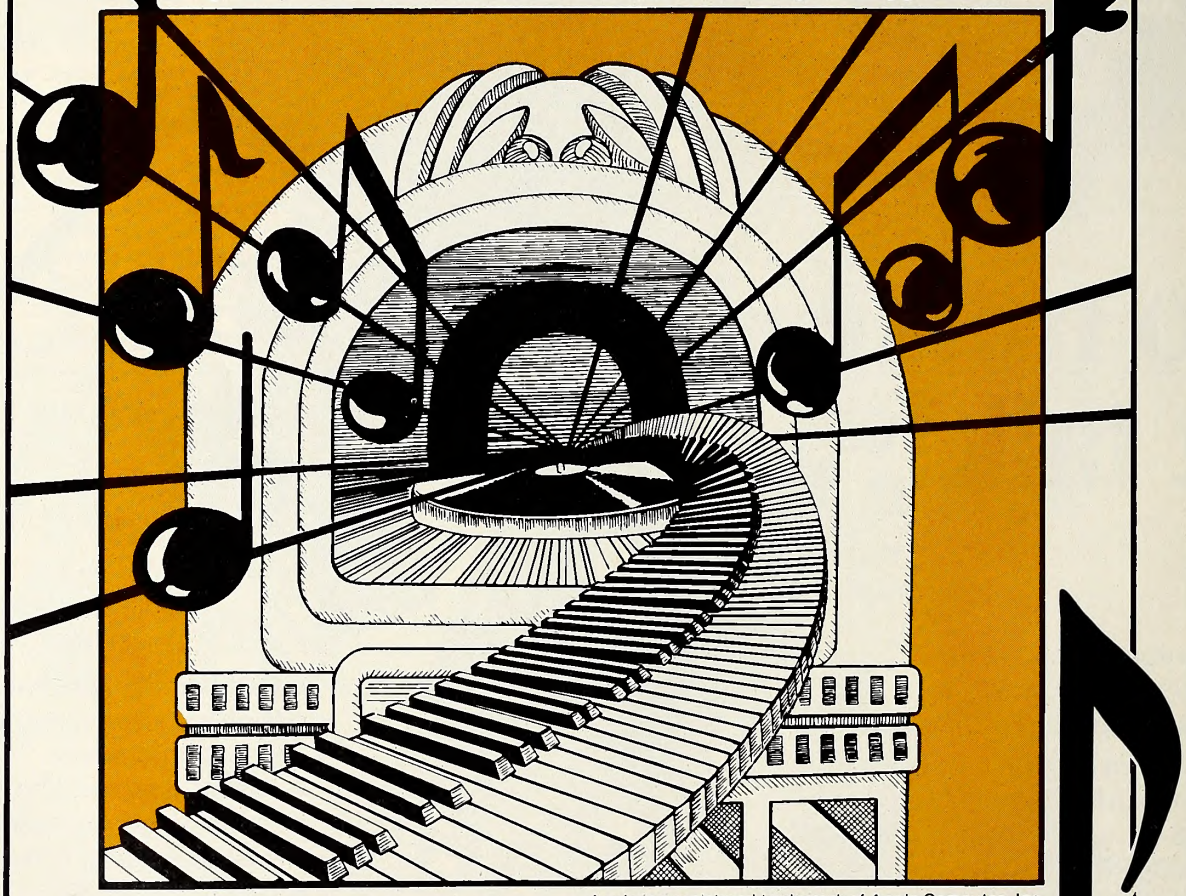
Two-Voice Music Synthesizer

The only two-voice music synthesizer
for the Apple computer that does not
require additional hardware.

Only $29.95

by Paul Lutus

See and hear it at your Apple dealer.



Apple is a registered trademark of Apple Computer, Inc.

Once you listen to the Electric Duet,
you won't buy a "hardware" music system.
Your Apple can now play two simultaneous notes.
You can select from our included music library or create your own.
The Electric Duet features a music editor with easy piano entry...
transposes both pitch and durations...
and plays through the Apple speaker or tape output.
Plus
Music and sound effects you create with the Electric Duet
can be saved into your own programs.

Distributed by:
SOFTSEL
8295 S. La Cienega Blvd.
Inglewood, CA 90301
In CA (800) 262-4242
(800) 421-5770

INSOFT/ 10175 S.W. Barbur Blvd. / Suite 202B / Portland, OR 97219 / (503) 244-4181

GO AHEAD AND INTERRUPT YOUR APPLE

Part 2 of an Interrupted Article

BY DAN FISCHER AND MORGAN P. CAFFREY

Hello—you're back. Finish that book? Well, then, let's get down to business and hope we can get a thing or two accomplished without any more interruption.

Last month we introduced the concept of a microcomputer interrupt. We covered some basics of machine-language code operation and how the processor knows it has been interrupted. We described the four types of interrupts that commonly occur.

We covered some important topics such as registers (processor or status, accumulator, x, y, stack pointer), and how they must be protected during interrupt processing. To complete the story, we will cover how Apple software is used to process the interrupt event and some problems we found and overcame.

But first, if we may interrupt just once, let's consider some of the design decisions that went into the development of the Apple II.

Who could have foreseen in 1977 or 1978 the uses to which microcomputers would be put in the next four years? We doubt that the creators of the Apple ever dreamed in the early days that there would be 400,000 Apple IIs in the world by 1982. The Apple II has remained virtually unchanged since its introduction, yet the limits to the range of its application do not appear to have been approached. The plug-in cards and software that have been developed for it make the Apple II adaptable both to the business environment and to the highly technical control application environment.

The Apple II is a marvel of modern technology and engineering insight. If it were being built now, the Apple II would be designed differently. This is not meant as criticism. The idea of making a computer that is an open book to competing manufacturers is a model of foresight. In hindsight, however, several design decisions were made then that have the effect of making certain things somewhat difficult now.

Do you remember when personal computers were called "hobby computers"? If you were designing a hobby computer, you'd use different criteria than if you were designing a business computer. That explains how the Apple II happens to have a forty-column upper-case display; its designers didn't anticipate all the business uses to which the Apple II would be put. The limitations on the screen are the most obvious, but there are other design decisions that arose from the hobby computer mindset.

A number of functions that occur within a computer require precise timing. One method of fulfilling the timing requirements is to incorporate hardware timers that interrupt the computer at rigidly defined intervals. This involves hardware that is not provided in the Apple II; it also involves fairly sophisticated software.

A less elegant solution to the timing problems involves using the microprocessor itself as a timer. If the microprocessor executes a loop a known number of times, the exact timing of that loop can be determined. With one known time, a limited *clock* can be constructed in software. Here's an example of one such clock.

```
LDA #100 ; set timer
STA TIMER
LOOP DEC TIMER ; count times thru the loop
BNE LOOP ; if not zero keep counting
```

This routine will delay the program precisely 806 memory cycles (788 microseconds). The Apple uses timers like this to put data on disks, read the game paddles, and, in the case of the Apple Serial Interface Card, to serialize the bits of memory cell going out of communications line (to a printer or modem). Timers like this work; they are versatile and they are cheap.

When a Cheap Solution Is Expensive. Software timer programs are cheap if you have nothing else to do with your microprocessor while you wait for the timer to "time out." They

work as long as they don't get interrupted.

Aha! You noticed. Using the processor for a timer is incompatible with having interrupts operating in the system. The interrupt will do its job—and disfigure timing loops—without the timing loop knowing that the wrong amount of time has actually elapsed.

Because of such software timers being used with DOS, NMI interrupts cannot be used in the Apple II while disks are operating. Showing foresight, the system has activated the IRQ mask, however, allowing the interrupts to be ignored while time-critical tasks are being done.

Since we know of no serious software that doesn't use disks, the rest of this article will be restricted to IRQ interrupts.

The conflict between IRQ interrupts and software timers can be resolved by setting the IRQ disable bit in the P register (the SEI instruction in an assembly language program) before entering the timing loop. When the timer runs out, the assembly program may reenable IRQ interrupts by a CLI instruction. In fact, the routine in DOS that reads and writes to the disk does just this to protect itself from IRQs.

Using interrupts with the game I/O paddle and *high-speed serial card code doesn't work.* The Monitor routine that reads the game paddles does not properly disable IRQ. Apple's Serial Interface Card protects its timers from interrupts but errs in its restoration method, so an interrupt with the high speed serial card cannot work.

In a nutshell, the results of these design limitations are:

1. The Apple II disks can operate with IRQ interrupts active but will mask them out while accessing the disks with time-critical loops.

2. The game paddle routine in the Monitor will give erroneous results if interrupts are active.

3. The Apple Serial Interface Card cannot be used when interrupts are active.

The F8 ROM Problem. There's one more design problem that must be handled before interrupts can be run with DOS active. Earlier, we described in general how the Apple determines whether it has an IRQ or a BRK. It saves the A register to gain a working register and then checks to see if the BRK bit is set in the P register. The entire F8 problem revolves around *where* the A register is stored.

How the Monitor and DOS Share Page Zero. Another small digression is required here. Page zero, consisting of the first 256 memory locations in the Apple, has special significance for the 6502.

Locations $45 through $49 are used by the Monitor to store the A, X, Y, P, and S registers during a break or at other times. Special routines are included in the Monitor to store all microprocessor values in these memory locations and retrieve them later.

Location $45 is also used as a multipurpose Monitor work area, called *A5H*. Some routines in DOS that are *not* protected from IRQ interrupts use location $45. This is the crux of the problem. Sometimes the contents of location $45 are very important to the proper operation of the active program. DOS uses the location for *bsave* and *bload* operations. If an interrupt occurs at this time, the wrong memory area can be stored to the disk. Other errors can occur that destroy existing files stored on the disk in the drive.

Location $45 is where the Monitor stores the A register while it determines whether it has a BRK or an IRQ. As noted above, this can be a deadly flaw unless DOS commands and some Monitor routines are protected from IRQs.

When we wrote *Doubletime Printer* we realized that it would not be a very useful piece of software if we required users to modify all their DOS programs to protect location $45. Instead we decided to eliminate the location $45 conflict by modifying the Monitor ROM. The only software that would have to be modified would be interrupt handlers—which we saw as a minimal problem because very few Apple users are already implementing interrupt-driven applications.

Let's look at what the IRQ routine does in the Apple Monitor and then at the way we changed it to permit wider use of interrupts.

If you can, refer to page 143 of the *Apple II Reference Manual*. In the middle of the page is a routine named IRQ. The first thing it does is store the A register in location $45 ("ACC"). The next two instructions pull the P register from the stack and restore the stack. (The P register had been put on the stack automatically by the 6502 when it encountered the BRK or IRQ.)

The next three instructions move the BRK flag to the position showing the minus condition. The next instruction tests this bit and transfers control to the break routine if there was a break (minus condition). If no break occurred, it must have been an IRQ, so the next instruction jumps to the routine whose address is stored in page three at location $3FE,3FF. At this point, the user's interrupt routine takes over.

The interrupt handling routine can restore the system status flag by loading A with the contents of location $45 and doing an RTI (return from interrupt). The RTI restores the P register and the program counter. The only thing lost is the original content of location $45. *But that can be fatal.* It can even destroy disk files.

If there's no user IRQ routine, the indirect jump to $3FE transfers control to the Monitor. It is the equivalent of a CALL-155. The stack still has the P register and the PC from the interrupt. The Monitor doesn't know what to do with these, so the program that was interrupted is probably irretrievably bombed. This is rarely a problem because few programs will enable IRQs. Fewer still will do so without first setting the page three vectors.

The Doubletime Interrupt Vector. Consider what can be done to protect location $45. The Apple Monitor decides whether there is a BRK or IRQ before passing control to the user routine: The sequence of operation can be modified to pass control to the user routine *before* making the determination about IRQ or BRK.

With the doubletime ROM in place, the IRQ routine starts out with the indirect jump to the location stored in $3FE. It is left to the user to decide if it is a BRK or an IRQ. At the time the user gets control, all the registers are still set as they were at the time of the interrupt, except the P and PC registers. Location $45 has not been altered.

New Responsibilities. In addition to the register-saving responsibilities that the user program has always had with interrupt processing, the interrupt handler program must now also determine whether it is processing a BRK or an IRQ. Further, it must save and restore the A register. The routines must not restore the A register from location $45, but rather from whatever location was used to save them. The rest of the IRQ routine of the autostart Monitor is essentially unchanged. If the page three IRQ vector $3FE,3FF points to $FA43, the doubletime Monitor will make the determination between IRQ and BRK. It will process a BRK with the same code that the autostart ROM uses. For an IRQ, it will jump to the Monitor.

DOS Patcher. Along with *Doubletime Printer* and the enhanced F8 ROM, we have included our program DOS Patcher on the *Doubletime* disk. It has been placed permanently in the public domain and is included in the San Francisco Apple Core Disk of the month. The most important purpose of this set of utilities is to make changes to the DOS on the disk that you're booting to handle the changed ROM appropriately. This utility will quickly alter any unprotected DOS to point the IRQ vector to $FA43. This will handle IRQ and BRK just like the autostart ROM with no user interrupt routine.

What will happen if we experience an IRQ when the doubletime ROM is installed, there is no user routine at $3FE, and DOS has not been patched? The same thing that would happen if we got an IRQ with the autostart ROM with no user routine: we will land in the Monitor (with a beep). What will happen if we get a BRK? The same thing—we will land in the Monitor. Unless DOS is altered, a BRK will be processed exactly like an IRQ. No damage will be done to the existing program, although the program will stop.

Tips For Interrupt Programmers.

1. Interrupt programs must be in assembly language.
2. Interrupt programs must be located in memory that will not be used by the foreground program.
3. The interrupt vector at $3FE,3FF must be set before interrupts are enabled.
4. The interrupt program must restore all registers to the condition they were in before the interrupt.

Before we close, we'd like to suggest some things that can be done with interrupts: data acquisition (the outside world provides the interrupt when the data is ready), spooling (a periodic interrupt to move a character to the pointer), or your application that we haven't thought of. There's lots of spare time available in the Apple.

Concentration at Last—Maybe. Who hasn't wished for a weekend cabin in the country with no utilities, no access to worry, no—well—no interruptions? And it's possible; you can get away from interruptions if you really want to.

So too with your Apple. When the immediate processing ends, you can turn off the interrupt routine and leave your Apple in peace.

I've said enough, and you've got to go. Your phone is ringing. ■

Technical writer, programmer, and consultant Morgan Caffrey got tired of twiddling his thumbs while his Apple printed out his work. So he started fiddling instead of twiddling, and now his Apple prints while it plays. To see that your Apple got this broadening opportunity too, Caffrey became partners with Dan Fischer.

From managing "the world's largest commercial message switching computer center" to teaching folks to interrupt one of the world's smallest computers may sound like a strange career progression to some, but to Dan Fischer, anything to do with the Apple is better than not. Formerly a systems analyst for Univac, Fischer's love affair with his Apple brought him enthusiastically out of self-imposed early retirement.

OPEN DISCUSSION

from page 22

were delighted to see a review of clock/calendar cards in the September issue of *Softalk*. However, we could not help but feel that the article did great injustice in the comparisons of the clocks reviewed. Part of this injustice is revealed in the author's misconception that "some real-time clocks also do other things for you. Added features include built-in interfaces for printers and other accessories and. . . ." There are several multifunction boards available for the Apple II, but it should be made clear that these boards combine several *different* functions on one circuit card, not several *clock* features. Unfortunately, each of the individual functions tends to be compromised by the attempt to get in as much as possible for a reasonable cost. This is analogous to the all-in-one stereo system. It may be a bargain, but you can always get better performance and more flexibility with separate components.

What amazed us the most was the way Mr. Kaapke evaluated the individual clock boards. After expounding the many uses for a real-time clock, he still found the Mountain Computer CPS card "first and fanciest." Yet, he gives little information on the clock portion of this card. Its lack of interrupt generation is only mentioned in passing. The fact that it may not work with many programs written for the company's own Apple Clock without modification is not noted. Nor is the lack of year and day of week information in the emulation mode. Automatic time and date stamping of files? No. Pascal support? Not included. Use it in an Apple III? Forget it!

Next, the Thunderclock Plus was reviewed. Again, a lot of hoopla about BSR control but little information on the clock. Does it keep track of the year? No. Pascal support? 1.1 version only. Automatic Filer update is done with the *exec* function and therefore cannot be used from within a program. Apple Clock compatibility is very good but slot-finding routines will require modification to find the Thunderclock.

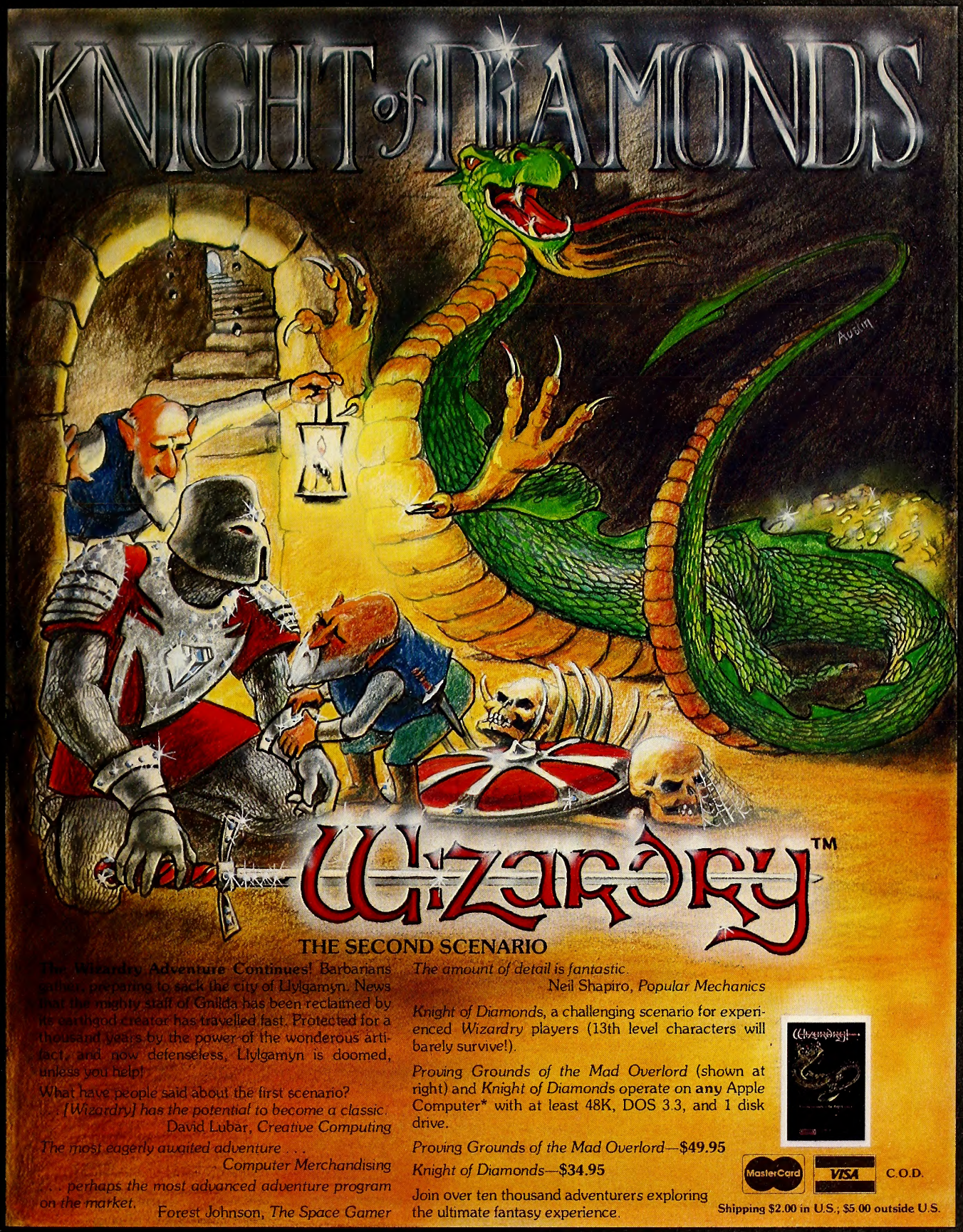
Finally, the author comes to the Superclock II and it seems he can't find much to say about it—its *only* a clock! It has "no X-10 accessory" and "only two operational modes." He then counters, "But the Superclock has more interrupt intervals then the Thunderclock." While this statement is factually correct, it totally misses the point. The important advantage of the Superclock's interrupt capabilities is in the wide selection of interrupt rates. These span from less than one microsecond (1,024 per second) to one hour. It is very useful to have one second interrupts, for example. But the Thunderware's interrupt frequencies only go down to a minimum of sixty-four times a second. This places a much greater overhead on the foreground program, causing it to slow down or possibly miss interrupts.

Practically unmentioned are some of the Superclock's other features:

Rechargeable Ni-Cd battery back-up
Day of week and year support
Full Pascal support, including Filer update
Millisecond timing routines

All these functions are directly concerned with the capabilities of a real-time clock, which is what the article was supposedly written about—not "other functions." Certainly, one should take into account these other functions; but be aware of what you may be giving up to get them.

We brought the Superclock II to the market only after we had completed all the software to make it an easy to use, functional system. We were the first to offer automatic dating of files for DOS 3.2 and 3.3. Our Pascal software was thoroughly researched so that automatic updating of the Filer and any diskette could be accomplished directly and un-



The Wizardry Adventure Continues! Barbarians gather, preparing to sack the city of Llylgamyn. News that the mighty staff of Gnilda has been reclaimed by its earthgod creator has travelled fast. Protected for a thousand years by the power of the wonderous artifact, and now defenseless, Llylgamyn is doomed, unless you help!

What have people said about the first scenario?

...[Wizardry] has the potential to become a classic.
David Lubar, *Creative Computing*

The most eagerly awaited adventure . . .
Computer Merchandising

. . . perhaps the most advanced adventure program on the market.
Forest Johnson, *The Space Gamer*

The amount of detail is fantastic.
Neil Shapiro, *Popular Mechanics*

Knight of Diamonds, a challenging scenario for experienced *Wizardry* players (13th level characters will barely survive!).

Proving Grounds of the Mad Overlord (shown at right) and *Knight of Diamonds* operate on **any** Apple Computer* with at least 48K, DOS 3.3, and 1 disk drive.

Proving Grounds of the Mad Overlord—**$49.95**

Knight of Diamonds—**$34.95**

Join over ten thousand adventurers exploring the ultimate fantasy experience.

MasterCard VISA C.O.D.

Shipping $2.00 in U.S.; $5.00 outside U.S.

NOW AVAILABLE AT YOUR LOCAL COMPUTER AND HOBBY STORE

*APPLE IS A REGISTERED TRADEMARK OF APPLE COMPUTER, INC.

Sir-tech **SOFTWARE, INC.**
6 MAIN STREET, OGDENSBURG, N.Y. 13669
(315) 393-6633

der program control. We have always believed in a "systems approach" and continue to support our clock with useful utility programs, such as our *Time-Clock II*. We will continue to expand and enhance the Superclock II as much as possible.

Thank you for the opportunity to express our views and beliefs.
West Side Electronics, Inc.,
Chatsworth, CA

The Good, the Gentle, the Important

In accordance with the January issue, I am letting you know. I do not want to miss this issue or any issue in the future.

Since I have taken the time to go this far, I will just take a few words to compliment you on the publication. I have watched with marvel as the expansion process takes place.

The departmental sections are of particular interest to me. Wagner's column has been the shining light, encouraging me to do some simple Assembly/Basic interface programming. I completely agree with his philosophy of using Basic except when the speed or special situations require the use of Assembly Language programming. I think I understand the USR function but would like to see Roger address it some day.

You can count on my check for the magazine when the new distribution policy is instigated. Between *Softalk* and *Nibble* you get the best of both worlds.

Some time ago I recall a reader who was interested in a "deaf terminal." I have a deaf son and very recently purcased Novation's *Apple Cat II* with the deaf options. Bill Collier is the project engineer at Novation in Tarzana, California. He seems to be a receptive individual as I have discussed several improvements to the "Deaf Software" with him. The program does work in its present form but lacks many features of *Comware II* which could and should be included. You may want to evaluate it. I am sure that there are plenty of Apple users that would be interested in this capability. I do not know of another modem/software package that allows communication within the deaf network.

Thanks for your ear. Keep the thing going.
Richard G. Orman, Rio Del Mar, CA

Friendly Apple People in Japan

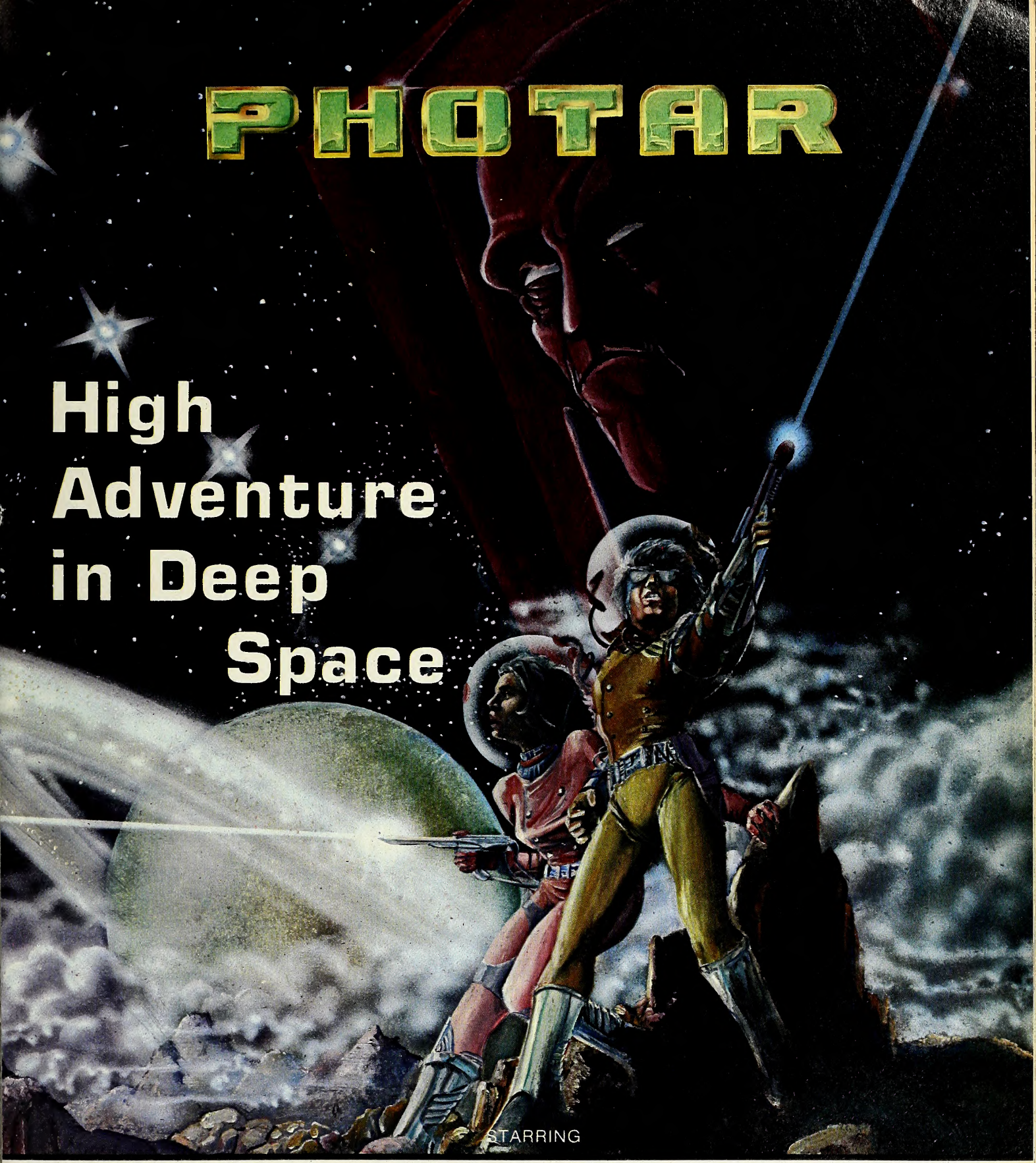
In September of last year I had occasion to pack my trusty suitcase and equally trusty Apple for a five-month research hiatus to the land of the rising sun and Honda automobiles. Knowing I was venturing into unknown computing waters I braced myself by lashing *VisiCalc* and *WordStar* to the main mast and set off on what I expected to be a journey fraught with dangers; after all, I'm constantly reminded of the coming Japanese computer invasion and here I was sallying forth into lands (and CPUs) unknown.

Fear not!! Fellow travelers, a warm welcome, a cold beer, and a hot computing time in old Tokyo can be had as I soon found out.

First, the Akihabara district of that fair city can only be described as a hacker's dream, containing as it does nearly twenty blocks of electronica to satisfy even the most rabid midnight wire-wrapper. More important, it was here I was introduced to the two gentlemen who proved to be my "open sesame" to the world of Japan and Apples, Shigeharu Ikeda, owner of Compucraft store, and Dr. Steven Bellamy, a British writer currently living over there.

Both gentlemen took a lot of time and trouble to make me (and incidentally any other fellow traveler) very welcome. In particular, Dr. Bellamy made special effort to show me around and introduce me to quite a few Apple users; his knowledge of computers and computing was outstanding and I considered it a privilege to have met him.

So as I bid farewell—a warm sake in hand—I would like to take the chance through your fine magazine to say a big thanks to all the Apple-people over here and to tell other interested voyagers that the land of Shogun is alive, well, and computing—Apple style!
Tony D. Artwell, Tokyo, Japan



PHOTAR ■ THE RINGS OF SATURN ■ PULSARS ■ KILOSTARS ■ BLACK STARS

IN

The Ultimate Experience in Outer Space Adventure

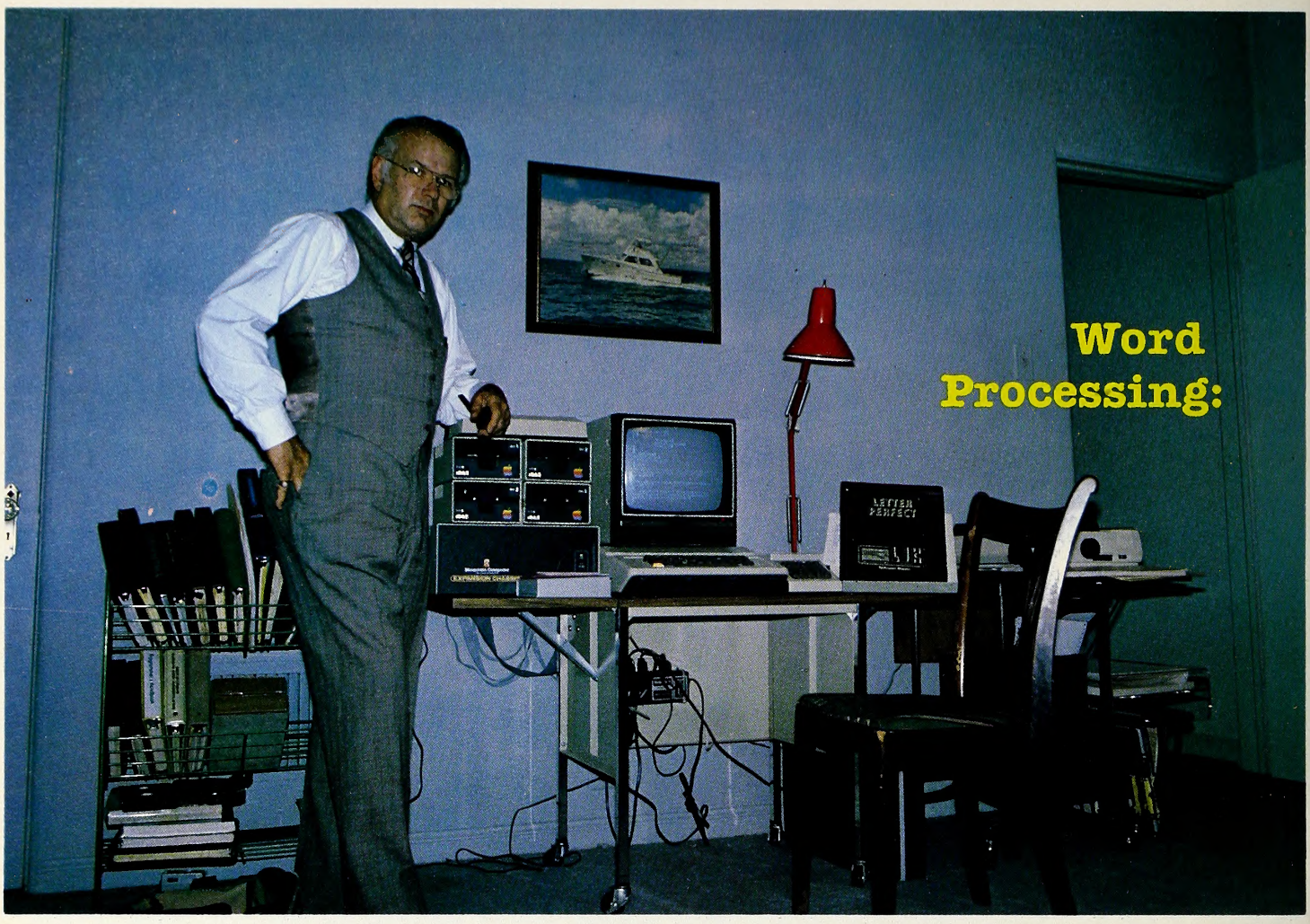
The Ultimate Experience in Outer Space Adventure
The Ultimate Experience in Outer Space Adventure

Produced by SOFTAPE An Artsci Company

Screenplay by SAB

Distributed through Softape, Softsel, and other computer stores world-wide

10432 BURBANK BOULEVARD ■ NORTH HOLLYWOOD, CALIFORNIA 91601 ■ 213/985-5763



When Porter Loring Writes, He Knows It's Letter Perfect

BY JONATHAN MILLER

Mild-mannered Porter Loring Jr. is joking, of course, when he talks about divorce, but the man from San Antonio does have a little marital problem. For over a year now he's been having an affair, an insidious infatuation that swallows evenings and Saturdays whole. Fortunately for Porter, his wife knows the identity of the other woman. Her husband's been smitten by an Apple.

The Apple a day that is keeping Porter away occupies a second-story bedroom in the Loring's San Antonio home. He apologizes for the bare appearance of the room, recently vacated by his youngest child, as he introduces a visitor to the object of his desire, an Apple II Plus—so faithful, so smart, sitting there along the west wall with his Epson printer, four disk drives, and a library of programs.

"I don't think there's anything that comes close to comparing with an Apple as far as value, flexibility, and ability," he boasts. "The Apple's never been wrong, though I've been wrong ever since."

Loring's laughing and looking very much the doting father, all crinkly with pride. "It's like talking about your children," he says. "Like when it's broken, you feel your children are sick."

A small picture of a powerboat, the only decoration on the wall above, recalls an earlier Loring passion. When his four children were growing up, the family would drive the one hundred fifty miles south to Corpus Christi on weekends to ply the Texas gulf and try their luck in marlin and sailfish tournaments. It was great sport, but the real fun for Loring was not the cruising and the fishing; it was the tinkering.

"It was working on the boat. If I had a problem it was necessary to get in there and fix it. No manufacturer has ever built anything the way I want it, so I've got to alter it. As soon as I had added everything to that boat—put in every kind of radar—I lost interest."

The Start of Something Big. As the kids grew older, the weekend trips fewer, Porter Loring was slowly but surely becoming a man without a hobby. For a while, he made the most of pipe collect-

ing, but his tinkerer's nature begged another escape and his family urged him to find one—soon.

"And I said, 'You're telling me to go and fall in love. Those things happen on their own—you don't decide you're going to go out and fall in love.' "

Then it happened. Radio Shack came out with a little pocket computer and Loring was "bitten by the bug, the venom permeating my body." He was fascinated, all right, but the machine was little more than a sophisticated calculator and he had to have more. He had to have an Apple. Not only for itself, not only for its business potential in data management and word processing, but because it filled another, more basic need.

"I live in a real intangible world," says Porter Loring, funeral director. "My problems are by and large intangible. They're people and emotional problems and I find a great escape dealing with tangible things."

Man is the only creature on this planet that ritually disposes of its dead. Death, the cardinal fact of life, has preoccupied the world's great religions and a species that refuses to accept it as the definitive end of life. It is, in short, a very, very serious business.

In the course of a year, Porter Loring Mortuary will handle twelve hundred funerals, roughly 20 percent of the city's business. The firm was founded by Loring's father, a former railroad worker, some sixty years ago when this historic southwestern cowtown ranked as the largest city in Texas. (Today it places third behind Houston and Dallas and owes much in economic vitality to the presence of seventeen area military installations.)

Strength of Tradition. Perhaps it was inevitable with the handle of "junior" following his name, but Porter never formally addressed the question of whether to enter the family business: he just did.

"For my generation and my father's generation things that were family things were accepted as family things," he explains. "I just never made a decision."

Not that he has regrets. Life has been good to him. Loring is active in state and national funeral director associations and he's a prominent civic figure—past president of more organizations than he cares to reel off and the man recently tapped to raise $1.5 million for a new public radio station transmitter. And then there's his work.

"My personality is such that I feel kind of rewarded by the strokes you get in this business," he says. "If you're open with people, if you're honest with them and are really sincere about doing what they want, they can tell."

The business has its rewards, but it also has its drawbacks. You're always concerned about going over the line, about saying something that will upset the delicate balance of emotions when a family faces this major crisis—accepting the death of a loved one; accepting death.

That's problem enough, but of course undertakers—funeral directors, in contemporary parlance—have others. Media problems: an image, in literature and the popular press, that vascillates between a cadaverous figure in stovepipe hat to an unctuous charlatan bilking sobbing, destitute widows. It's another matter about which Loring feels passionate.

A Delicate Balance. "We're an industry of response," he pleads. "We're here to respond to people's needs. We're not here to guide them or tell them what to do. We really don't manipulate people or try to overcharge or take advantage of the situation. Frankly, we bend over backward not to, almost to the point of not being any help at all."

Consider the purchase of the casket as satirized in *The Loved One*, the movie send-up of the three-billion-dollar-a-year funeral industry. It's the classic stereotype: the grieving, guilt-ridden relative, putty in the hands of the oily salesman (played by Liberace). Now consider Loring's approach. When families come in to select a casket among the thirty on display, Loring leaves them to make their own decisions.

"Obviously they're not going to be coerced into anything that will be contrary to what they think's appropriate," he says. "Then, on the other hand, they've never bought a casket before. They don't know anything about them and there's nobody to ask questions of. In other words, when it gets to making decisions in areas involving the expenditure of dollars, we just back off. We're somewhat afraid of being misinterpreted as trying to sell a bill of goods, so it's almost an overreaction."

But there's an eminently practical reason for this approach. Many times, says Loring, a family will be accompanied by a "hostile," a person whose role is to see that they're not taken advantage of. When Loring removes himself from the selection process, suddenly this family advocate finds he has no function.

"So all he can do is be our advocate and when they come out, this guy's attitude has changed an entire 180 degrees. He's become your confidant. It may sound like manipulation but it's not. It's just that if you leave people to do what they want to do, you don't have to worry about it. You just have to know when to be quiet and let people answer their own questions."

On funeral matters generally, says Loring, people behave a lot like fish: they school together. They find an area that

BUSINESS

"The Friendliest System For Your Apple*"

The Complete Mailing System for incredibly dynamic (and easy) file maintenance. Customized labels, 1 to 4 across, duplicates, multi-level COUNT/SORTs, range sorts, partial field sorts; Formatted 80-column reports, and MUCH MORE

ULTRA PLOT: Bar Charts, Stacked Bars, Line or Filled Line Graphs, Scatter Charts, Pie Charts, Hi-low, and our UNIQUE Map Charts. Includes Statistics. Save, edit, and superimpose any combination of graphs and charts.

The Interface. This allows you to use your Mailing System data files to create ULTRA PLOT data or picture files: average, search, sort and count; Merge, reformat, subfile creation, and any-field-index utilities included.

Includes 8 diskfuls, 5-part manual, Muffinable & Backup-able: $175 plus $3.75 post. & handling. (Requires 1 or 2 disk drives.)

UTILITIES

HI-RES SECRETS: Finally a graphics system that really teaches you about Apple* HI-RES and how to use it! THE SECRET IS OUT!!

Vector, Hplot & Text File Shapes, ANIMATION including Logical Shift, XDRAW, Page-Flipping; Machine Language Animation; Fonts, Sounds, Scrolling, Music; Color Animation, YTABLE; COLOR PALETTE & COLOR-FILL PROGRAMS

Includes 4 disks, 270-page manual: $125 plus $3.75 post. & handling. (Demo Disk: $10 plus $2 postage). Please specify DOS 3.3 or 3.2.

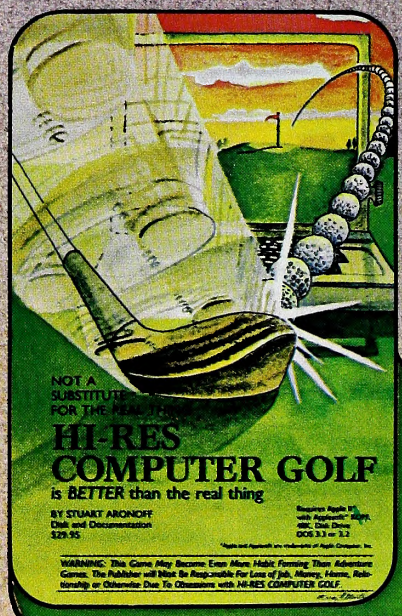
EDUCATION

Z.E.S. - THE ULTIMATE IN COMPUTER AIDED INSTRUCTION IS FOR APPLE* COMPUTERS ONLY!

THE MOST ELEGANT AUTHORING SYSTEM IN THE KNOWN WORLD. The system consists of 4 diskfuls: Program, Graphics, Module, and Module Format/Copy. Requires NO PROGRAMMING knowledge and is completely menu driven.

Includes 130-page manual, 4 diskfuls: $250 plus $3.75 post. & handling. (Demo disk: $10 plus $2 postage).

GAMES



HI-RES COMPUTER GOLF:

- 5 18-hole courses from Beginner to Champion
- A golf bag full of clubs to choose from
- 1 to 4 players
- Amazing Dynamic Club-Swing Control
- Contoured Greens/Multi-screen Fairways/Wind Gauge
- Obstacles: sand traps, roughs, water, trees, high winds
- Computerized Scorecard
- Mach. Lang. Sounds and Full Hi-Res Color
- Yelling "FORE!" is optional **$29.95**

Wasting time calculating grades?? STOP! Use our **A.E.N. Grading System**. Uses number grades or any of 3 types of letter grade systems. Reports of single students, categories, students or classes combined. Disk & documentation $70.

Does **SENTENCE DIAGRAMMING** have your students confused? Use the only one of its kind. Includes Parts of Speech, Usages. 3 levels, 60 sentences. Disk and documentation $24.95.

The **WORD SCRAMBLER & SUPER SPELLER** makes learning spelling fun! 3 files included. Add your own word files. Use for any level. Only $19.95.

The CREATIVITY TOOL BOX: Drawing, poetry and music, Action Sounds plus utilities. Tools and aids for programming or just for fun! Includes 88-page manual, 3 diskfuls: $44.95 plus $2 post. & handling.

© Evil Mastermind

Write for free literature and ASK YOUR DEALER for our complete line of software products for your Apple*.

All disks require Apple II* with Applesoft* ROM, 48K, disk drive, DOS 3.3 or 3.2. (Please specify.)

*Apple & *Applesoft are registered trademarks of Apple Computer, Inc.

VISA* VISA/Mastercard accepted MasterCard

AVANT-GARDE CREATIONS
P.O. Box 30160 Eugene, OR 97403
(503) 345-3043

fills their needs; in San Antonio that translates into funerals averaging $1,700, give or take $200.

Many supersensitive funeral directors would probably blame muckraking author Jessica Mitford (*The American Way of Death*) for the industry's image problems. Loring, surprisingly, takes a different tack. He actually thinks the book, which accused the industry of calculated overcharges, produced some unintended benefits.

"Here was this sensational book and suddenly death was not a dirty word anymore. People started talking about death, dying, and bereavement at cocktail parties and dinner parties and suddenly researchers were discovering a whole new field to look into. I think history could record that her book probably created the beginning of the most healthy approach to death. Funeral directors were saying, 'Let me look at myself. Am I real? Is what I'm doing for real?' "

Planning Ahead. Mitford's book dovetailed with the sixties impulse for greater personal freedom and expression of feelings to produce another anomaly: prearranged funerals, otherwise known as "pre-need." Today, the business of handling funeral arrangements before the day of reckoning is growing by leaps and bounds, says Loring, a past president of the National Funeral Directors Association. He averages 1.5 families a day, he reports, and "that's without aggressively soliciting the business."

It's a highly customized service, he explains. Some people will want to prepay their funerals and work out the arrangements to the minutest detail. Others will elect to drop in and file a card simply indicating general preferences. In states like Texas, says Loring, where an individual can prepay at today's prices, there's also an economic incentive. Prepayment is largely a middle-class phenomenon, says Loring, and particularly strong in California and Florida, states with large immigrant, single, and elderly constituencies.

"I think people feel that at some time in their life they must put things in proper order," observes Loring. "Either they want to relieve their children of the decision and financial obligation or they want to make sure that their wishes are expressed."

Loring is a great believer in self-expression, however crazy it may seem. He cites the example of a dying woman who asked that her coffin be fashioned from a tree growing in her front yard; of a lumberman who wanted his service to be said amid the trees he loved; and of the grandchildren of a family who honored the memory of the deceased by raising champagne glasses in a spontaneous graveside tribute.

More often than not, the individual touch will be subtler—a family-made arrangement of flowers for the casket taken from the garden of the deceased or

a few heartfelt words from employers and friends.

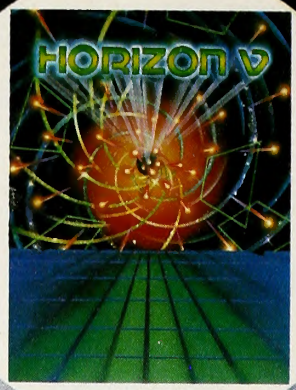
"Those are the kinds of things that wipe you out emotionally in my business," says Loring. "When you see people varying from the norm, you see they do it with great feeling and commitment. A person may be ninety-three and have had the most beautiful life in the world; their death is really a victory and a release. Their death can be acknowledged in many ways, as a Christian would see, often by celebration, which is really a celebration of a successful life. When you see people express themselves in real gut-level expressions of devotion and admiration, which is hard for the layman to do, it just shoots the legs out from under you."

The Right Words. When you live in an emotionally charged environment, words loom large. You have to be extremely careful not only in the words you write, says Loring, but in what those words say. And those, he adds, may not be the same thing.

"If I'm writing a speech or an article or even a policy for people, I have to have the flexibility and editability that is available on a word processor," he explains. "It allows me to change an article from A to Z or just to change the whole tone of a sentence."

Loring is a satisfied user of LJK's *Letter Perfect* word processing system. It isn't his first processor, but it is his favorite, thanks to its format flexibility and friendly, menu-driven features. *Apple Writer*, he says, never felt as logical or comfortable, and *Pie* was not as easy as

NASIR GEBELLI PRESENTS:



HORIZON V

NASIR outdoes himself again . . . this time with three-dimensional effects in a simulated space battle that rivals the best of arcade machines. From the open plains of alien planetoids to the twisting vortex of time, this game is destined to be the measure of three-dimensional simulation. Requires 48K Apple II or II+ with disk drive.

RUSSKI DUCK

Introducing a new concept in computer games . . . the real-time adventure. As a secret agent working for the DIA, you must recover the stolen LX missile plans before foreign agents can smuggle them out of the country hidden in a duck. By Eric Knopp and Alan Merrell, requires 48K Apple II or II+ with disk drive.

FIREBIRD

An exciting new arcade game by the master of Apple animation: NASIR. PIGGO, the fearless firefighter, battles blazes left in the fiery wake of the incredible FIREBIRD. Requires 48K Apple II or II+ with disk drive.

HOLLYWOOD

GEBELLI SOFTWARE INC.

Gebelli Software Inc.,
1771 Tribute Rd., Suite A, Sacramento, CA 95815

its name might imply. And as for service, LJK can't be beat.

"Not only does LJK help you with your problems, they'll go another mile and send you a new disk," says Loring.

Loring employs *Letter Perfect* in three areas: to compose letters, to write speech notes and articles, and to design forms he uses in conjunction with his *DB Master* data management system. He could dictate letters to a secretary, he explains, but he finds something is always lost in the translation. The Loring solution: rough it out on the word processor, then give the secretary the printout for retyping.

That may seem to confound the advantages of word processing, but not from Loring's standpoint. His is a personalized business and computerized correspondence and billing statements just don't fit.

"In fact," he says, "I had a hard time going to electric typewriters because I wanted the holes in the paper to show this was done by hand—that this is a personalized statement and that you're not part of a process."

For similar reasons, Loring rejected another state-of-the-art innovation proffered by a software salesman: a computerized telephone sales pitch featuring the disembodied voice of Arthur Godfrey.

"It was very sophisticated," says Loring, "and very scary."

As a prominent civic figure and active member in state and national funeral directors associations, Loring is often asked to give speeches and write articles. It's a chore made less onerous by *Letter Perfect*. A recent association project: forecasting the funeral service of the twenty-first century. The result? Laughs Loring, "It turned out we didn't know."

Perhaps his favorite use for *Letter Perfect* is in the design of forms, including one for a pilot data management program he's devised to track business trends. After six months of data gathering, Loring says he has yet to discern any observable pattern—"Women seem to do pretty much the same as men, the Baptists pretty much the same as Catholics"—but the exercise has justified itself, at least to him.

"Well, my secretary could have done this, but I had a lot of fun designing this simple little form. It was very easy to do, but, hey, I did something."

Loring would like to do more with this program—for that matter, he'd like to bone up on machine language—but he's not sure he's qualified.

"A good statistician needs to know two things—what information to get and what to ask of it. I've got a lot of information, but I'm not sure I'm smart enough to know what to ask for."

New Worlds to Conquer. Whatever comes of this project, others are waiting in the wings. He's considering bringing his computer into the office environment, hiring a bookkeeper who isn't frightened to death by the sight of an Apple, and developing another program to track, if not control, his manpower needs. Personnel tends to be a problem area, he explains, because not enough qualified people are attracted to the work, which suffers the additional burden of irregular hours and minimal pay.

Finding new business applications for his Apple, of course, is a fine way to justify his investment in hardware and software—"I've bought everything you can buy for the damn thing," he jokes—but he knows it's mostly a glorious hobby and a fine excuse for having fun. He can sit in front of his CRT, pipe in hand, looking pensive and professorial, or amaze incredulous friends who drop by from time to time to have Porter "do something," as if his Apple were a performing circus animal.

In matters computer, Loring tends to be self-effacing.

"I'm a super example of a little bit of learning being a dangerous thing," he says. "I've never learned the axiom, 'If it works don't fix it.' "

The wags down at the computer shop say there ought to be a lock on his Apple to keep him out of it. "But that's part of the fun of it," protests Loring. Like putting chips in backward, misconnecting wires and committing that number-one no-no: changing a board with the power on.

"If you're really honest with these guys and admit you goofed," he explains, "they're a lot more responsive and you learn a lot more. You don't learn a lot blaming it on somebody else."

The Quality of Life. The fun of computers and the sobering reality of death—they're both an education. After a lifetime in the funeral business, Loring is constantly reminded of his mortality, of the accelerating passage of time—the shorter it seems it has been and will be.

"But it increases certain values you have," he reflects. "It's made my family and my time mean more to me. Life must really have quality to it because it's all you have."

The interview is winding down, but Porter Loring, fifty-three, has one parting observation he'd like to share about the binary world.

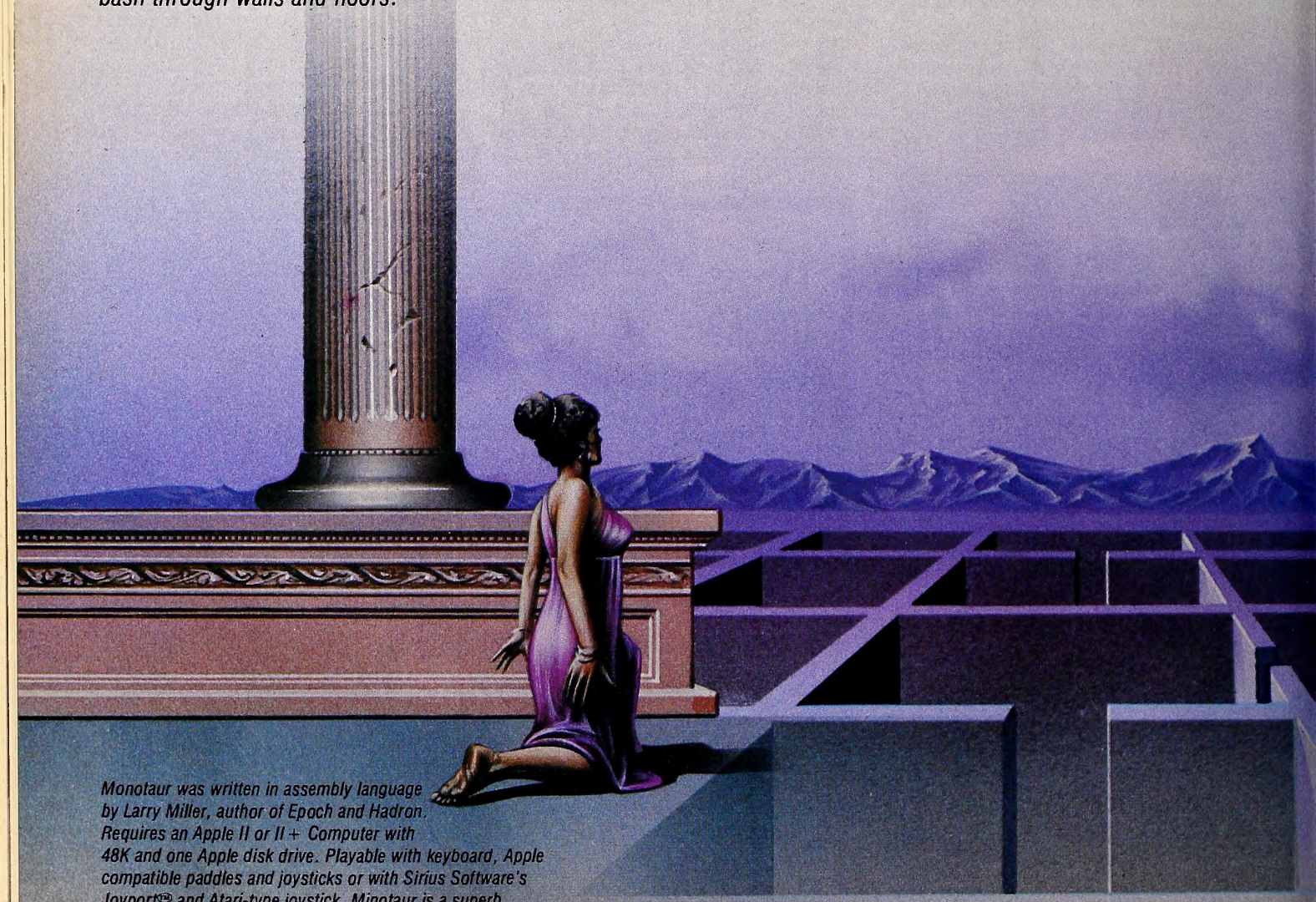
"Computers are built on the yes/no. When you think of every decision we make in life, everything you put into your brain, ultimately it resolves down to a yes or a no. The compromises even end up being yeses and noes. You experience something or you don't; you're exposed to something or you're not. Essentially, it's all the same kind of process." ❑■

Flash back with us to Ancient Crete . . .

Remember when Poseidon gave Minos, the King of Crete, a bull to sacrifice . . . Well, when Minos kept it for himself, Poseidon retaliated by making Minos' wife fall in love with the bull. The offspring was a rather disagreeable blend of man and bull — the Minotaur.

Minos decided to have Daedalus build a complex labyrinth to hide the beast. Minotaur was kept amused with a regular feeding of twelve youths and twelve maidens. You can imagine how popular this was with the Cretans. Enter our hero, Theseus (disguised as a mild-mannered victim) who secretly hoped to kill the Minotaur. The King's daughter (naturally) fell in love with Theseus and gave him hints (from Daedalus) on how to beat the beast. Of course, Theseus killed the Minotaur, got the girl and lived happily ever after (that is until his wife fell in love with his son, but that's another story).

Sirius now offers you the chance to re-live this classic story in a modern setting. Take Theseus' place in the labyrinth and you will travel through 32 different three-dimensional mazes. Armed with a laser sword, you will encounter a myriad of fanciful and frightful creatures. In our version of the story, the Minotaur keeps company with Flying Snakes, Charon, Dragons, Ghosts, Spiders, Scorpions and other monsters too horrible to mention. You'll start out with 99 lives, find Wings to fly above the maze, a Virgin to grant you extra lives, an Elixir to drink which allows your spirit to roam free, a magical Skull, and a Club to bash through walls and floors.

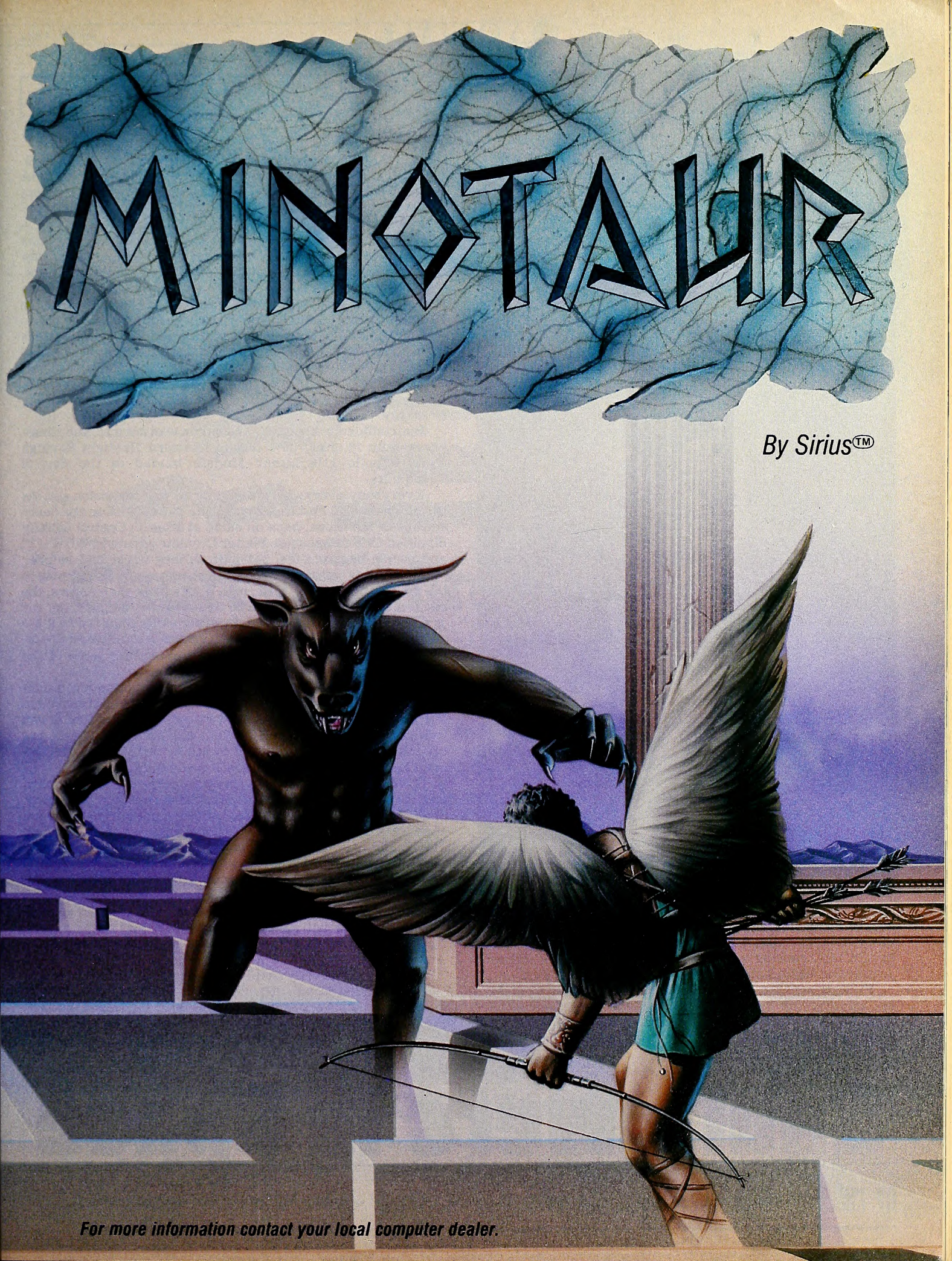


Monotaur was written in assembly language by Larry Miller, author of Epoch and Hadron. Requires an Apple II or II+ Computer with 48K and one Apple disk drive. Playable with keyboard, Apple compatible paddles and joysticks or with Sirius Software's Joyport™ and Atari-type joystick. Minotaur is a superb blend of arcade action, adventure game strategy, and great graphics.

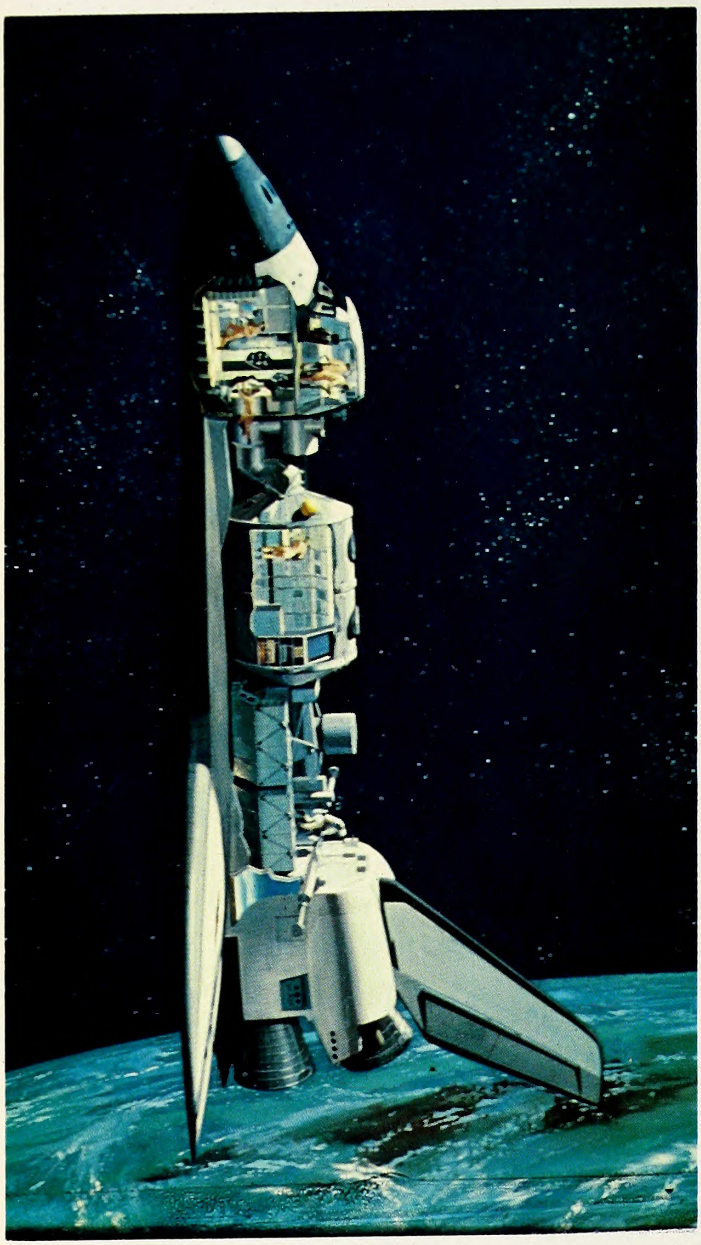
Sirius Software, Inc., 10364 Rockingham Drive, Sacramento, CA 95827 (916) 366-1195.

MINOTAUR

By Sirius™



For more information contact your local computer dealer.



Sunflowers in Space

BY MELISSA MILICH

The moon was within reach. Rocketing from Russian soil to three hundred miles high in space, Sputnik orbited the earth first in 1957, causing wonder and a little bit of jealousy on the part of the United States.

Firstest with the Mostest. In a technological fervor to catch up, some of the first American rockets were fired off with monkeys and chimpanzees in the driver's seat. The race in space was on, but the Russians held onto their lead, sending their first cosmonaut, Yuri Gagarin, into orbit in 1961. Alan B. Shepard, Jr., followed, barely a month behind, in the USA spacecraft Mercury-Redstone 3, which went up and came down without orbiting, reaching 115 miles at its apex and then dropping with a big splash in the Atlantic Ocean.

Six months later, before Cape Canaveral had had a chance to cool down, Mercury-Atlas 6 carrying John Glenn sped into orbit and circled the earth three times during a five-hour flight.

Work stopped in American cities and towns. Children were let out of school. Strangers began talking to one another in the street. The United States was finally ahead of Russia.

The Space Age brought its victories and its tragedies. Both nations lost men: The Apollo 1 cabin fire took the lives of astronauts Virgil "Gus" Grissom, Edward White, and Roger Chaffee in January 1967. Three months later a Russian spacecraft made a crash landing killing its pilot, Colonel Vladimir Komarov.

But confidence in space exploration was not lost, and, finally, on July 16, 1969, at 4:19 p.m. Eastern Daylight Savings Time, Apollo 11's Lunar Module landed in the Sea of Tranquility.

Everybody everywhere was in front of a television set. As the hatch opened and Neil Armstrong put his foot on the dusty surface of the moon, back on earth at Mission Control usually dignified CBS newscaster Walter Cronkite shouted "WOW!!!" and people began to feel patriotic as they never had before.

Succeeding missions brought land rovers and golf games to the moon, and, little by little, as it goes with human nature, the excitement began to wear off. Today, most people take for granted the scientific and engineering skills required to put a person on the moon, and it may take the sophisticated Spacelab and the reusable space shuttle to recapture some of the thrill.

Some NASA officials are cautiously predicting communities in space within the decade. Rendering of civilian space hotels and metro transport between planets are no longer drawn from the imagination of H. G. Wells and Arthur C. Clarke but are taken from plans produced by space engineers at McDonnell Douglas and other aerospace companies.

Computers, of course, are playing a major role in the space program. Large mainframes were there from the beginning, and, during more recent years, microcomputers have begun appearing in various research laboratories and offices around NASA. But only one micro is scheduled to go into outer space.

Although its flight is not scheduled till mid-1983, the Apple II Plus will be on board Spacelab when it makes its premiere voyage in September of that year. Spacelab is a joint effort of NASA and the European Space Agency, and mission specialists on the crew will be conducting more than thirty experiments in the biological, physical, and life sciences during its seven-day orbital flight. One of these research projects, called HEFLEX, will directly involve the Apple.

Honeysuckle Knows. The HEFLEX experiment should answer a question that has been nagging scientists since the time of Darwin: How do plants know which way is up?

It's a good question. Cover a seedling with dirt and no light

Chart your financial future with MicroFinesse™

In this fast-paced business world, the best way to stay competitive is to be able to see your financial alternatives clearly and make decisions fast.

That's why P-E Consulting Group Ltd. developed MicroFinesse, a complete package providing professional financial forecasting PLUS full high-resolution color graphics support, including pie charts, histograms and graphs, for the financial projections you create.

Now this evolutionary financial resource planning tool eliminates the time-consuming reprogramming required for financial model consolidation or expansion. Micro-Finesse's menu-driven command format can also generate up to 15 user-defined reports per model, with visuals, without reprogramming.

Previously available only for main-frame applications, the MicroFinesse program can now be purchased at leading Dealers nation-wide. MicroFinesse is specifically designed for the 48K Apple II® with the Apple Language Card.

So when your variables are many and your time is limited, take a good look at the financial artistry of MicroFinesse.

MicroFinesse

Distributed by: OSBORNE/McGraw-Hill
630 Bancroft Way, Berkeley, CA 94710, (415) 548-2805

Apple II is a registered trademark of Apple Computer, Inc.


The Apple® Polishers
The most comprehensive Apple Catalog ever offered. Over 200 pages of Apple Computer Systems, Displays/Monitors, Printers, Software, PC Cards, Games, Books and Accessories! Proven products carefully selected for their quality and cost effectiveness by the Apple experts at Computer Town. To get your catalog send $9.95 refundable with your first order, to Computer Town today.
Computer Town
307 S. Broadway • Rte. 28 • Salem, N.H. 03079
CALL 800-258-6378 IN N.H. CALL 603-893-8812
TWX 710-366-0507 MICRO COURIER 603-893-4361
AMERICAN EXPRESS Cards Welcome
MasterCard
VISA
NO SALES TAX
Computer Town

can pass through to guide its growth. Yet, given enough water and oxygen, the roots of that seedling will push farther beneath the soil and its sprouts will eventually break through to the surface. A plant knows which way is up.

Plants grow up but not in a linear fashion. The roots, branches, tendrils, and other parts of most plants grow in a spiral pattern, somewhat like the rising stripes of a barber pole. This growth process, called circumnutation, was first scrutinized by Charles Darwin during the latter part of the nineteenth century.

Darwin's theories about circumnutation are somewhat akin to his theory of evolution. In *The Power of Movement in Plants,* published in 1880, Darwin propounded that the phenomena of circumnutation is endogenously motivated—a driving force or a behavioral trait hidden within the plant makes it grow the way it does.

The plant power theory seems only slightly less controversial than some of Darwin's other opinions. A number of scientists religiously adhere to an alternate view; the theory that plant growth is gravity-dependent, that gravity pulls the plant up and around in the corkscrew pattern. The differing opinions about the cause of nutation have caused a "friendly dispute" within the scientific community about the cause of nutation, the Darwinian-endogenists versus the gravity-dependents.

About ten years ago, a group of scientists on each side of the controversy decided that the matter could be resolved once and for all by conducting a plant growth experiment in a weightless, zero-gravity environment. One of two things could happen: nutation would either proceed as usual or the plant would forget which way is up. When the experiment's results were in, the scientists of one group would have to modify their thinking.

A gravity-free laboratory can be simulated on earth, but scientists are not sure exactly how close this comes to the actual environment in space. They suspect it's fairly imperfect.

Along came the plans for Spacelab, which was designed for research projects requiring a zero-gravity atmosphere. Experiments conducted aboard Spacelab could conceivably start a long string of breakthroughs in medicine, physics, and biology, since scientists will have the opportunity to do research in an environment that's not restricted by the earth's gravitational forces.

Flower Power. When the call came for proposals for scientific experiments that could be run aboard Spacelab, Dr. Allan Brown and his sunflowers were among the first to apply.

Brown is a research biologist at the University of Pennsylvania and is on the gravitational side of the great plant growth debate. He's hoping for an answer with the experiment he calls HEFLEX, named after *Helianthus annuus,* the dwarf sunflower which will be used as the test subject.

In project HEFLEX (HELianthus Flight Experiment) several sunflower plants in various stages of growth will be videotaped in the weightless environment of Spacelab's orbital flight. When Spacelab returns to earth, still photographs will be made from the videotape and analyzed frame-by-frame by a group of biologists to determine what effect the zero-gravity had on the sunflowers.

The live sunflower seedlings will be some of the last items loaded into Spacelab before it blasts off, and that's where Brown will have to bid his plants a temporary farewell. Since still very limited numbers of people can go along on an actual flight, Brown will be leaving his experiment in the hands of the three astronauts from NASA and the European Space Agency who are scheduled to fly with Spacelab in 1983. These crew members, who are referred to as "payload specialists" by NASA, will be handling each of the research projects on the orbital flight, including the sunflower experiment.

Sunflower Basics. The payload specialists have already undergone at least two in-depth training sessions at the University of Pennsylvania in order to completely understand the mechanics and biology of Project HEFLEX.

"Plants really are interesting things," says Brown. "You'd be surprised."

That was the first thing the astronauts learned; the procedures of the experiment came next.

A population of *Helianthus annuus* seedlings will be planted sequentially over a six-day interval. Since nutation becomes pronounced in seedlings 100 to 130 hours old, some of the plant specimens will be started before the launch so they'll be ready to go on camera by the first day of the flight.

The sunflowers will arrive on Spacelab in a plant carry-on container, which looks something like a large suitcase. The older plants will go directly into a photographic chamber, while the younger seedlings will go into a centrifuge where they will spin in a one-gravity centripetal force, to simulate the gravity of earth. Payload specialists will plant the seedlings and add water to initiate germination. Each time a group of eight plants reaches the proper age, the payload specialist will select the four best for camera surveillance.

Choosing the plants will probably be the most difficult part for the payload specialists, especially since the Apple will be running the rest of the sunflower experiment. The Apple will take the role of a "dedicated microprocessor," responsible for regulating and monitoring centrifuge speed and temperature surrounding the seedlings and for controlling the photography that will take place at ten-minute intervals recording the nutating or nonnutating plants. The computer programming required was contracted out to a private company in Pennsylvania, and the routine but time-consuming duties the Apple performs will free up time for the astronauts.

But They Can't Play Snoggle. Astronaut flight time is precious in terms of American dollars, and NASA wants to automate whatever it can. Every minute counts. According to Brown, every human being aboard the upcoming flight has his time scheduled to within plus or minus six minutes. Thanks to the Apple, the payload specialists won't be tied up with routine tasks, which will enable them to go on to the other forty some-odd experiments aboard Spacelab or perhaps to steal a mo-

ment to enjoy the scene outside the viewing ports.

The Apple II Plus selected for the 1983 flight looks slightly different from earthbound models since only the boards will be used. They have already been stored away at Johnson Space Center and are waiting for the countdown to go to work. Brown, however, still has a lot of work to do before then. A group of sunflowers just returned from the March space shuttle flight as part of a pre-HEFLEX engineering test to determine the optimal soil moisture for a zero-gravity environment.

And even though Brown and his colleagues will be left behind when Spacelab finally takes off, they will be extremely busy at Mission Control monitoring the experiment from the ground. With all the botanists waiting for the results of the experiment, Brown wants to make sure nothing goes wrong.

Because It's There. The HEFLEX experiment probably won't generate enough excitement or publicity to cause strangers to stop each other in the street and comment on "those amazing sunflowers," but Brown and his colleagues have been waiting a long time to solve this mystery. And what will its significance be beyond proving a point? Says Brown:

"There's no immediate practical application. We're not doing this to see if we can grow sunflowers for the astronauts to eat in space. And we're not trying to change the price of sunflowers on earth.

"But it will give us one more new piece of information to work with as far as space research goes. Further experiments down the line could employ our results for an agricultural space application. Right now, it's just a basic research question."

The sunflowers will share their quarters aboard Spacelab with about thirty-five other basic research projects, everything from an electron beam generator to blood sample studies. The beam generator will eject a stream of electrons from Spacelab that will charge the atmosphere around the orbiter and see if it perhaps creates an artificial aurora. In a series of life science studies, the astronauts will draw fresh blood from each other's veins to measure the effects of weightlessness on such factors as hemoglobin concentration.

The value of space research cannot be disputed, especially when you look back at discoveries made during early space exploration missions and their earthly applications. Fluorocarbon propellants in spray cans were banned in this country after Mariner 2 and Mariner 10 experiments showed scientists their danger to the protective ozone layer covering the earth.

That may have prevented a future disaster. Meteorological studies and Voyager photographs from the earth's upper atmosphere have also led scientists to a better understanding of global weather patterns, hurricanes, and solar energy.

NASA astronaut Owen Garriott, who will be aboard Spacelab 1, points out that the space program has been a major impetus in the development of computers. The microminiaturization of electronic circuitry was driven by the requirements of spaceflight. And from the results of solid state circuitry technology came the silicon chip, and hence the desktop computer.

Where will it lead us? Garriott predicts there will be permanent space stations on the moon, probably within the decade, to which scientists will shuttle back and forth from the earth. Your local airport may soon become your local spaceport.

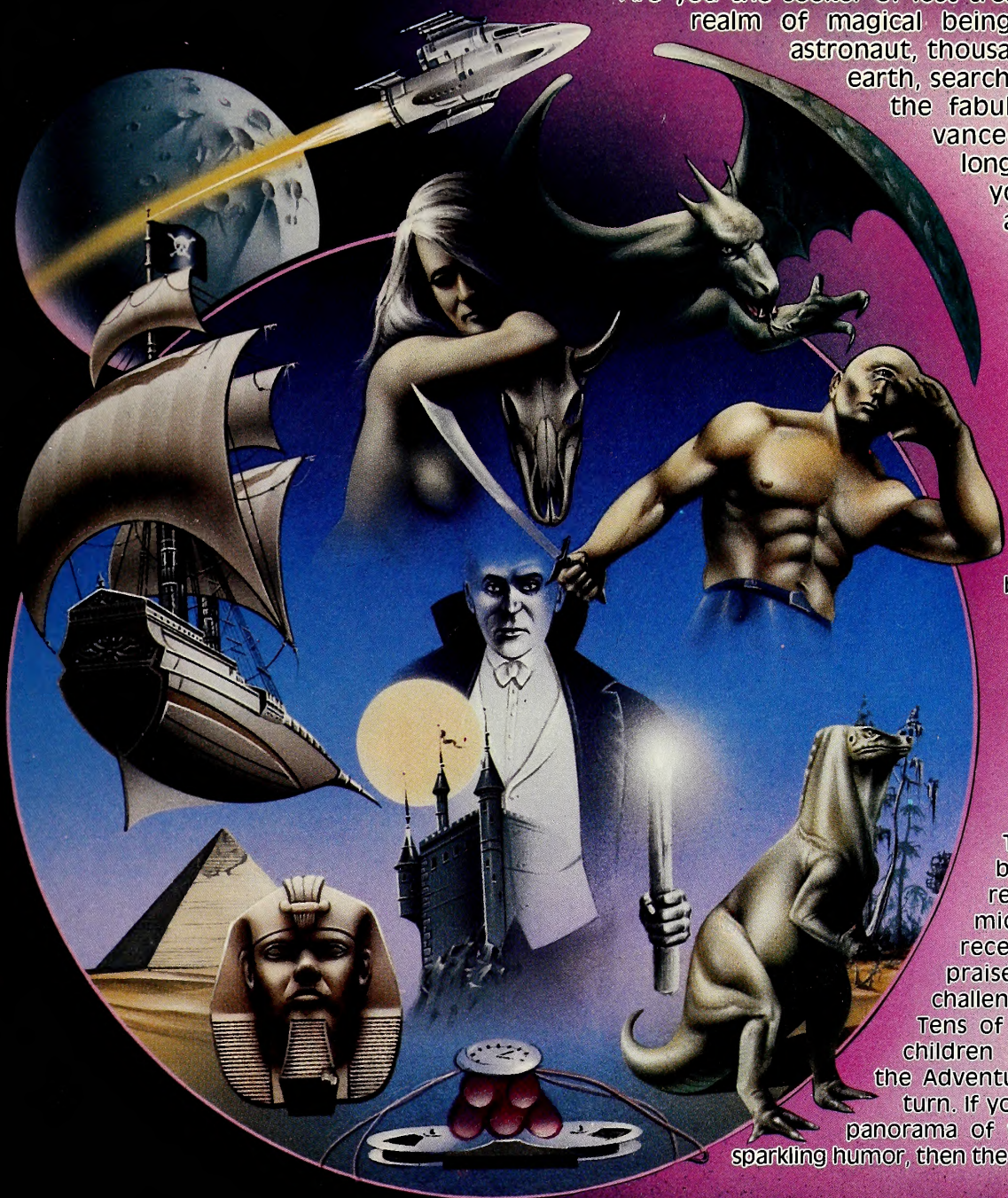
Across the Universe. The sunflower experiment is only one small step in the entire Spacelab project which in turn is only a small part of the entire space exploration phenomenon.

As long as NASA manages to wrestle money away from the national budget, space exploration will move on. No period in history has fostered so many discoveries since Magellan and Columbus and their colleagues set sail across wide unknown oceans just for the sake of exploration. And it shouldn't stop now.

Most certainly there are planets to leave footprints on, possible life to discover, galaxies to travel, and new civilizations to build.

There are miles to go before we sleep. ■

WHO ARE YOU THIS TIME?



Are you the seeker of lost treasures in an enchanted realm of magical beings? Perhaps you're an astronaut, thousands of light years from earth, searching the galaxy's rim for the fabulous treasures and advanced technologies of a long-dead civilization. Maybe you're the plunderer of ancient pyramids in a maddeningly dangerous land of crumbling ruins, and trackless desert wastes.

Imagine.

You wish to soar to other worlds, to behold wonders never beheld by mortal eyes, to dream unrestrained to the furthest limits of your imagination.

Imagine now that you can.

The ADVENTURE SERIES by Scott Adams has been reviewed by **every** major microcomputing magazine, receiving only the highest praise for its mind-puzzling challenge & refreshing originality. Tens of thousands of adults and children have matched wits with the Adventure Master. Now, it's your turn. If you crave the challenge and panorama of the exotic, touched with sparkling humor, then the Adventure Series is for you.

With the incredible Adventure Series by Scott Adams, YOU ARE WHOMEVER YOU WANT TO BE, WHENEVER YOU WANT TO BE!

	SINGLE TAPE	12 PACK TAPE	SINGLE DISK	TRIPLE DISK	12 PACK DISK
APPLE	$19.95 24K	$129.95† 24K	$29.95* 48K	$69.95* 48K	—
ATARI	$19.95 24K	$129.95† 24K	—	$39.95 40K	$129.95† 48K
CP/M (Z-80)	—	—	—	—	$129.95†
TRS-80 MOD 1&3	$19.95 16K	$129.95† 16K	—	$39.95 32K	$129.95† 32K
TRS-80 MOD 2	—	—	—	—	$129.95

PRICES SUBJECT TO CHANGE WITHOUT NOTICE

Adventure Hint Book (All 12)	$ 7.95
Adventure Hint Sheet (Single)	1.00
Commodore VIC (Cartridge)	39.95
T.I. 99/4 (Tape or Disk)	29.95
Requires Adventure Command Module	

*Scott Adams' Graphic Adventure Series Apple Disks Full Color Hi Res Graphics. †12 Adventure Hint Book Incl.

SHIPPING & HANDLING ARE EXTRA

Adventure INTERNATIONAL
A DIVISION OF SCOTT ADAMS, INC.

BOX 3435, LONGWOOD FL 32750
(305) 862-6917 (QUESTIONS)
ORDER FROM YOUR FAVORITE DEALER or CALL TOLL FREE (800) 327-7172
(ORDERS ONLY PLEASE A.I. #119)

FUNTASTIC!

DATAMOST

9748 Cozycroft Ave.,
Chatsworth, CA 91311
(213) 709-1202

NEWSPEAK

☐ **Higher Education.** Clarke College, Graduate Division (Dubuque, IA 52001; 319-588-6331) will this summer become one of the few institutions of higher learning in the country to offer the Master of Arts degree for computer applications in education. Clarke, the first small college in the United States to offer a computer science program, added the computer applications concentration to its education major after concluding a study that indicated teachers are generally unprepared to use computer technology in the classroom and would take the course. Degree candidates will take nine to twelve hours of education courses, fifteen to eighteen hours of computer science courses, and three to six hours of electives. The college's facilities include seventeen Apples and an IBM 4331 mainframe.

☐ **Velocipedenal Interface.** March 7, 1982, marked the Third Annual Old Road Ten Mile Time Trial of the North Hollywood Wheelmen, a California bicycle club that used to speed over the roads of the Malibu Canyon area until the road conditions deteriorated to such a degree that the club went looking for a new course. They found it in Valencia, under the shadow of Six Flags Magic Mountain.

It also marked the first Wheelmen race timed by the Apple, provided courtesy of club treasurer Steve Drucker. Drucker, a math teacher who got his Apple in 1980 "to play games and keep track of grades," eventually hit upon the idea of combining his micro hobby with his other vocation, bicycle racing, to provide on-site race results of the kind furnished to regional marathons by Race Central of Rialto, California ("Race Against the Apple," *Softalk*, November, 1981). Drucker's system is not quite as elaborate; at this point it involves spotters along the course with stop watches, entering the 142 racers' starting and finishing times on cards with their name, age, gender, and club affiliation; four bits of info per card. After the race, the cards were taken to Drucker and his waiting Apple at a nearby house and keyed in. The final results were produced two hours after the finish, an improvement over the manually tabulated results of all previous meets, which usually took three to four days of card shuffling to find out who finished when. (The special prize for fastest time went to John Stevenson, 21:43.)

For future runnings of the Old Road trial, Drucker hopes to have the Apple in a van on the course site for even faster results. Though in his scorekeeping capacity Drucker is not as much an active participant as he once was, his enthusiasm is undiminished. "You're riding as an individual against the clock; you can try for a personal best, but it's primarily a great event to see where your fitness level is."

The Wheelmen, founded in 1946, have twenty-three national championships to their credit, plus two Pan-American gold medals and a world competition silver medal, and have participated on U.S. Olympic teams.

☐ **A Smarter Card.** A laminated plastic card with a memory chip, developed by a French subsidiary of Honeywell, is being tested by the Department of Agriculture as a possible replacement for food stamps. Known as an "intelligent" card, it would store the data on a household's monthly food allotment, which would be entered at the counter of a grocery store every time the card is used, and registered at the local welfare office. The Department hopes the card may reduce the problems of fraud and waste in the 11.3 billion dollar food stamp program.

☐ **A Law of the Land.** Pierce Brooks, the former Eugene, Oregon, police chief who was investigating detective in the "Onion Field" kidnap-murder case, is seeking a source of funding for the Violent Criminal Apprehension Program, a national computer center that would link evidence in random motiveless slayings in an effort to prevent "serial murders," that proportion of homicides in the United States that are committed by one person or a group of persons who repeat the crime in different police jurisdictions. Data on cases involving missing children with evidence of abduction, a disappearance with evidence of foul play, children who escape from kidnappers, and homicides that appear random and motiveless and include ritual aspects would be entered into a VI-CAP computer and analyzed for similarities. The program would then notify agencies of patterns in slayings and put police officials in different parts of the country in contact with each other in the hope of developing leads or suspects. All police agencies must cooperate for the program to be successful.

☐ **Court Order.** Charged with the November 23, 1980 killing of Xavier Guillen, manager of a Cudahay, California apartment building, and the subsequent slaying of a fellow jail inmate, Michael Contreras is being tried in Los Angeles Superior Court with the aid of a computer terminal. Contreras claims he cannot hear, read lips, or understand deaf sign

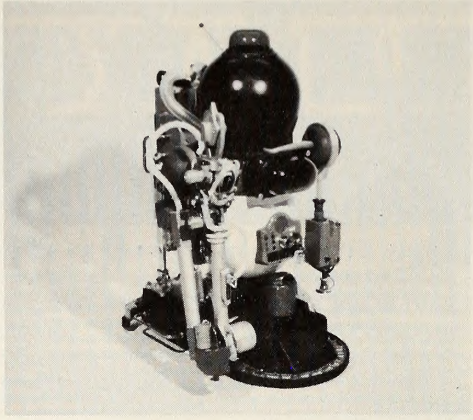
language.

Testimony taken in shorthand form by the court reporter on an electronic stenographic machine is transferred to the computer terminal, which matches it to words in its vocabulary, using phonetic spelling for words it doesn't know. Court reporter Jere Shank programmed the system with sixty-eight thousand words at the beginning of the trial and has since added three hundred more, mostly street slang. Both Contreras and Judge Ricardo Torres are able to read the testimony on separate terminals.

The computer system was developed by Xscribe of San Diego, and includes the ability to print out a transcript.

☐ **One Disk, One DOS.** The subgroup of the Institute of Electrical and Electronic Engineers involved with microprocessor operating system interfaces, headed by Jack Cowan of Intel, has completed a first draft of voluntary industry standards for microcomputer operating systems. It will be published before the end of the year.

☐ **It Doesn't Do Windows.** The readers of *Robotics Age* recently chose Charles Balmer's design as the winner in the magazine's homebuilt robot picture contest. Balmer's creation, Avatar, is an autonomous mobile robot with sensors and a system purpose of finding a feeding station to recharge its batteries, the first stage of development in realizing the robot servant concept popularized in science fiction. Its 64K on-board computer doubles as a software development work station when the robot is immobile and connected to a disk drive, CRT terminal, and printer.



Avotor, the 64K winner of the Robotics Age homebuilt robot contest; photographed by its designer, Chorles Bolmer.

Balmer, a resident of Urbana, Ohio, works as a microcomputer system design and engineering specialist, and performed his experimental work on Avatar over the last four years in his spare time.

Make Me a Match. What do a lock of George Washington's hair, John Wayne's yacht, and the 1928 Porter used in "My Mother the Car" have in common?

These items and many others are catalogued in a computer marketplace for the ultrarich. Run by a company called The Investment Matchmaker, the service links buyers and sellers of real estate, classic cars, businesses, and the exotic. Investment Matchmaker was started five years ago, soon after company president James Hall acquired his own computer. Now the company is housed in a plush office, three blocks above the Sunset Strip, in Hollywood.

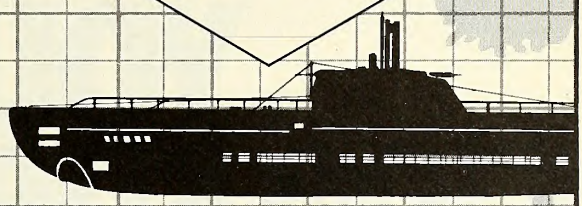
The service caters to clients in the United States, Canada, and Mexico, with 50 percent of its business coming from the entertainment communities in New York and Los Angeles. The company currently has two thousand five hundred sellers and five thousand buyers in its computer banks, and adds approximately forty clients each week.

At a cost of $400 to $1,000, Investment Matchmaker will provide the seller of goods with a computer printout of prospective buyers who are interested in what's being offered. Buyers pay a $45 fee for the service of being linked up with the goods.

Items recently handled include a forty thousand-year-old mastodon tusk going for $30,000, and a conch belt worn by Elvis Presley being offered by his brother for $60,000. Twenty percent of Investment Matchmaker's business involves items of this sort, while 80 percent of the information they provide covers real estate, classic cars, and businesses. ❑■

2 FROM HORIZON

Fathoms 40™



Time: June, 1943.
Place: North Atlantic.
German Wolf Packs have been driven from their hunting grounds by Allied escort vessels and massive air attacks. But unknown to the Allies is a Nazi super sub completed well ahead of schedule: The Mark XXI. Der Führer has demanded total destruction of enemy merchant shipping — a vital link for continued success of the Allied war effort. Will you, as one of the Kriegsmarine's remaining ace skippers, alter history and save the Third Reich from defeat?

Der Führer will not be disappointed!

FATHOMS 40: $39.95 ppd. Foreign, FPO and APO add 10% for shipping. Please specify Apple or Atari when ordering.

CLOAK & DAGGER™

In the 1930's dramas were slowly unfolding in capital cities throughout the world. From dimly lit hotel rooms to highly sophisticated Intelligence Headquarters, men and women were playing a game of cat and mouse for scraps of information essential to the security of their country. CLOAK & DAGGER pits you against computer, or other players, as head of an Intelligence Agency trying desperately to stay one step ahead of its rivals. Moving agents from capital to capital, ferreting information, and offering bribes are fundamental parts of each player's turn . . . and your survival!

CLOAK & DAGGER $29.95 ppd. Foreign, FPO and APO add 10% for shipping. Please specify Apple or Atari when ordering.

For Apple II*, DOS 3.3, or Atari 800* 48K with disk drive. Joysticks required. Games on Dysan** diskettes. Visa & MasterCard accepted. Dealer Inquiries Welcome, contact your local dealer for full details.

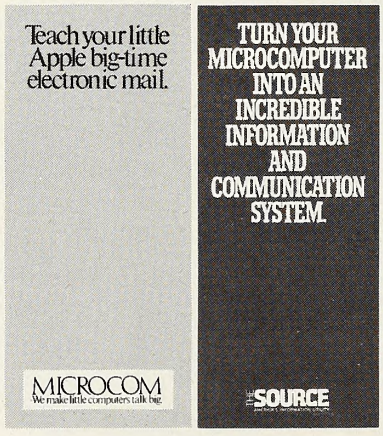
HORIZON SIMULATIONS
Horizon Simulations Inc.
7561 Crater Lake Highway / White City / Oregon 97503
(503) 826-4640

*Apple II, DOS 3.3 and Atari 800 are registered trademarks, respectively, of Apple Computer, Inc. and Atari, Inc. **Dysan is a registered trademark of Dysan Corporation.

IF YOU BUY MICRO-COURIER™ BEFORE JULY 1, YOU GET A $100 SUBSCRIPTION TO THE SOURCE℠ FREE.

Micro-Courier is *the* electronic mail software for Apple™ computers. It lets you send anything in your Apple computer to any other Apple, in any other office, anywhere in the world. It's fast, 100% error-free, and inexpensive. You can send and receive charts, graphs, VisiCalc™ reports and entire programs.

Micro-Courier also takes you right to THE SOURCE. You can access one of the largest data bases in the world, including news, travel service, commodity updates, educational programs, electronic games and more.



And if you act now, you can get a free $100 subscription to THE SOURCE with your Micro-Courier package.

See your Apple dealer for complete details.

MICROCOM
We make little computers talk big.

1400A Providence Highway
Norwood, MA 02062

Apple is a trademark of Apple Computer Inc. VisiCalc is a trademark of VisiCorp., Inc. THE SOURCE is a servicemark of Source Telecomputing Corporation, a subsidiary of The Reader's Digest Association, Inc.

THE BASIC Solution

By Wm. V. R. Smith

Many times, Basic Solution has offered routines that we hoped would help you produce efficient and eye-pleasing output. We've provided methods to position dollar values correctly and to handle most of the overhead involved in creating professional-looking output. Most of these routines from the past work with both the video screen and printed output.

Now that the prices of dot matrix and letter quality printers are coming down from $600 and $1,300, more of you are likely to want to write programs that produce printed material.

If you are like most home programmers, you begin the creation of paper reports on a trial-and-error basis. Once you've written the main program, you print out the first style of your report and then, slowly and laboriously, hand-massage and edit it and each succeeding output until reams of paper are used up and you've achieved the report format you want. This method does get the job done, but it takes many hours and lots of paper to get the desired result.

Some companies have attempted to produce paper grid patterns for programmers to use. Their method has been to write in each grid square the appropriate letter—the letter that should be printed in that space—thereby creating a handwritten X/Y grid format of what formatted reports should look like for programming purposes. Using a handwritten grid report is the best way for the programmer to specify the appearance of output.

The new, inexpensive printers have some very nice features, one of the best of which is the capability to go back and forth between expanded and condensed print. Taking advantage of this requires the programmer to find the proper grid format before beginning report specifications. While most expanded and condensed pitches are uniform from printer to printer, some are not, and these can make it very difficult to find the proper grid paper.

This month's Basic program will take your printer and whatever format you specify and print out a grid format for you, using the exact character size for your printer grid. The program is relatively short and can be changed if you require a shorter or taller grid.

Good format can be very beneficial in producing all types of forms that include mailing labels, especially when you are not quite sure how many blank lines to print between labels or where to position the print head for the zip code or other special information.

Programming efficiently is the key to secure and professional programs. Keep your eye on future Basic Solutions for new programs and concepts that will help you tackle the time obstacle in computer programming.

```
 1  REM ******************************
 2  REM *
 3  REM * OUTPUT FORMATTER
 4  REM *
 5  REM ******************************
10  PRINT CHR$ (4);"PR#1"
20  FOR Y = 0 TO 60
25  PRINT Y;: POKE 36,5
26  IF A = 0 THEN FOR T = 1 TO 9: FOR TT
    = 0 TO 9: PRINT TT;: NEXT TT: NEXT T:A
    = A + 1: GOTO 70
27  A = A + 1: IF A > 4 THEN A = 0
29  PRINT "]";
30  FOR X = 1 TO 8
40  FOR Z = 1 TO 9: PRINT " ";: NEXT Z
50  PRINT X;
60  NEXT X
70  PRINT
80  NEXT Y
90  PRINT CHR$ (4);"PR#0"
```

If you have any questions or have written any routines that you'd like to share with fellow readers, please forward them to Softalk Basic Solutions, 11021 Magnolia Boulevard, North Hollywood, CA 91601.

Unless otherwise noted, all products can be assumed to run on the Apple II, Apple II Plus, and Apple III in the emulator mode and to require 48K and one disk drive. The requirement for ROM Applesoft can be met by RAM Applesoft in a language card.

☐ *The Program Writer* by Bob Schermeister enables the user to write, access, and update a hashed data file. Functions as a standalone multiple database; will write to text files from Business Basic, making the files interactive. Custom data entry form and screens will display calculations between price and quantity ordered. No maximum edits per field, transfer up to two hundred fields, transfer amounts between files, print out from scan. Coming soon from **Vital Information** (7899 Mastin Drive, Overland Park, KS 66204; 800-255-5119).

☐ *QuikFolio* is a brief portfolio evaluator serving as an introduction to a software series for the small investor from **The Computing Investor** (29 Estancia, Marana, AZ 85238; 602-682-4444). Gives total worth, dollar and percentage gain/loss. Free to anyone sending a mini-floppy and return postage.

☐ **The Computerist's Directory** (Box 405, Forestville, CA 95436; 707-887-1857) has published its spring, 1982 edition. White pages include clubs, user groups, bulletin boards, glossary, and alphabetical index. Yellow pages list computer accessories and supplies, consultants and programmers, hardware and peripheral manufacturers, software applications, publications, retail stores, and mail order houses. Also lists of databases, information providers, and data communications networks. $7.50; one-year subscription $10.

☐ **The Learning Circuit** (2083 Westwood Boulevard, Los Angeles, CA 90025; 213-475-8528) offers Apple computer courses to children and adults, ranging from computer games to advanced Basic and graphics techniques. Seminars for special interest groups; facilities available for personal and business use on an hourly basis. Self-tutoring programs in English, mathematics, and science.

☐ The 1982 edition of the *Tax Preparer* by **Howard Software** (8008 Girard Avenue, Suite 310, La Jolla, CA 92037; 714-454-0121) prepares and prints out Form 1040 and nineteen others. Expanded features; new features for 1982 include easy editing of itemized lists, year-to-year transfer of data, automatic computation of ACRS depreciation deductions, and full handling of all minimum, maximum, and income averaging tax alternatives. $150.

☐ **Ken Cohn** (23767 Calle Azorin, Mission Viejo, CA 92692; 714-770-8843) has developed *Deadbolt*, a hardware device for protecting software. Plugs into game port. Deadbolt, $10; encryption, $200.

☐ *MusicMaster*, a series of interactive programs in basic musicianship from **Syntauri** (3506 Waverly Street, Palo Alto, CA 94306; 415-494-1017), uses the *alphaSyntauri* keyboard to teach music students mastery of ear training and theory fundamentals. Beginner module now available, featuring scales, intervals, and triads. Student record-keeping and performance analysis, teacher management system, and default settings. Requires AGO keyboard, interface, software, and two foot pedals; audio processing hardware, and audio system. $150. ☐ *Draw Waves* is a new sound design tool allowing design of arbitrary sounds beyond the conventional saw/sine/square/triangle waves technology. Graphical representations of waveforms are drawn on the screen using an analog controller, freehand, point by point; or with lines between points, or by specifying vectors using x/y coordinates to obtain arbitrary waveforms. Requires *alphaSyntauri* digital system. $29.95. ☐ The *alphaSyntauri Five* is a full five-octave digital synthesizer with all standard features plus multilayer sound-on-sound recording (bounce), an eight-way programmable keyboard split, and instant exponential or linear envelopes. New performance operating system, heavy-duty foot pedals, and punch-in sequencer. $1,795. ☐ The *alphaSyntauri Plus Four* provides eight-voice polyphony, a single track recording system, and sounds design and analysis programs. Both synthesizers will run the *MusicMaster* programs. $995. Interconnect cable, $35. Current five-octave instrument owners may update their software for $150.

☐ *GSS-CORE* is a computer-independent, two-dimensional graphics subroutine library for programmers and system builders who want to create their own applications using computer graphics. Will draw lines, create objects, add text and color, create graphs, create charts, and analyze data. From **Graphic Software Systems** (Box 673, Wilsonville, OR 97070; 503-682-1606). Single and multiple user systems; $300 to $1,500. ☐ **NewsNet** (945 Haverford Road, Bryn Mawr, PA 19010; 703-790-5500), a new electronic information distribution and retrieval service for the business community, is available for the publication of electronic newsletters. Features headline scanning, keyboard search of back issues, and automatic clipping service, plus UPI news reports and two-way electronic mail between readers and publishers. Now conducting full-scale system tests with "Communications Daily" and seventy newsletters from sixteen publishing groups.

☐ **Creative Software** (6081 Barbados Avenue, Cypress, CA 90630; 714-893-4695) has announced *Data Base III*, a database program for the Apple III featuring user-designed screens with calculated fields, preformatted fields, input with syntax checking, password protection, and a report generator with totaling. 128K required. License fee, $175.

☐ **Interactive Structures** (112 Bala Avenue, Bala Cynwyd, PA 19004; 215-667-1713) is marketing its *PKASO* series of printer interfaces featuring dot graphics with full printer resolution, user-definable characters and symbols, sixteen-level gray scale printing, total screen dump for hi-res, lo-res, or text, and full Apple III operation. Supports various printers. $165.

☐ *The Ghostwriter*, an authoring system from **Cavri Systems** (26 Trumbull Street, New Haven, CT 06511; 203-562-4979), allows non-programmers to create, modify, and update sophisticated interactive video or CAI programs for information management applications. Provides graphics capability, word processor/test editor, unlimited branching, and scoring of responses. On-line manual accessible from any point in lesson development. Requires two disk drives and 48K with 16K card. $3,500. ☐ *Cavri Systems Ink*, a new quarterly newsletter for users of interactive video systems, is available free from the company.

☐ Howdy, farmers (and we call you farmers 'cause who else would be up this early?) A whole flock of agrarian software this month; most of it from **Farmplan** (Box 65, Campell, CA 95009; 408-379-3932). *Dairy Herd Management* is designed to record detailed information about each cow's production and breeding records. Provides reports and comparative information on herds of up to 590 head; herd activity data entered at weekly to monthly intervals; action list, management reports, and individual cow reports. *VisiCalc*, *FarmFiler*, and *Least Cost Ration Formulator* programs included. Requires printer and two disk drives. $950. ☐ *Crop Management* creates crop budgets and cash flows, prints reports of resources required

and used, and maintains records of field applications. Field yield entered after harvest for gross profit/loss statements. Records for one hundred fifty fields, two hundred resources, and fifty user-designed application programs. *VisiCalc* and *FarmFiler* included. Printer and two disk drives required. $950. ☐ *Pig Breeding Management* is designed to keep records while providing management reports, casualty reports, listing of females by status, and herd evaluation information. Configured to keep current records for a herd of sixty boars and six hundred sows with ten litters each. Herd activity data recorded weekly; weaner and suckler deaths reported. Includes *Least Cost Ration Formulator, FarmFiler,* and *VisiCalc.* Printer and two disk drives required. $950.

☐ A collection of worksheets with the career farmer in mind, *AgriCalc* is now available from **McIntosh Software** (2428 1st Avenue N.E., Cedar Rapids, IA 52402; 319-366-6327). Includes depreciation schedules, balance sheet, personal budget, cash flow, interest calculator, loan amortization schedules, vehicle maintenance records, commodities account worksheet, agricultural cost analysis, recipe files, and a checking account with income expense analysis. $125.

☐ A new paper tray from **Praxis** (8327 Potranco Road, San Antonio, TX 78251; 512-684-3231) fits into the corner recessions of the Epson MX-70 and 80, allowing printer and paper to be moved as a unit without unloading. $34.95.

☐ Three new games from the microcomputer division of **Avalon Hill Game Company** (4517 Hartford Road, Baltimore, MD 21214; 301-254-5300). *Guns of Fort Defiance* is a real time arcade/strategy game putting the gamer in command of a Napoleonic artillery crew. Correct ammunition for target, fuse length, direct or rolling fire elevation adjustment, and deflection needed for each shot must be selected. Every attack successfully repelled is followed by a more determined one. $20. ☐ *Computer Stocks and Bonds* is a micro version of the company's investment simulation board game. Up to four players; initial $5,000 allows speculation in shares of nine different corporations or investment in municipal bonds. Simulated wire service teletype relays news flashes of special circumstances affecting individual stocks; graphic bar analyses display price histories of stock; portfolio includes reviews for charting past securities movements. $20. ☐ *Dnieper River Line* is loosely based on the 1943 engagement between Russian and German troops in the southern Ukraine. Computer controlled Soviet units attempt to overrun player's German lines and capture sufficient objectives to assure victory. Each side may draw from fifteen types of units with varying strengths and capabilities. Package includes pawns representing all infantry, artillery, and armored units, and a full-color mounted map board. $25 on multi-programmed cassette; $30 for disk.

☐ *Basic Programs for Scientists and Engineers* by Alan R. Miller has been published by **Sybex** (2344 Sixth Street, Berkeley, CA 94710; 415-848-8233); text phototypeset from a word processor via computer controlled interface to preserve the accuracy of the programs. Applications in computer graphics, statistics, and business management. $14.95.

☐ Users of the productivity tools of **Software Design Associates** (260 Madison Avenue, New York, NY 10016; 212-686-2032) are invited to attend *Productivity in the 80's*, their twentieth user conference, May 17-19, 1982, at the new Hyatt Regency Crystal City in Arlington, VA. Attendance fee includes workshop materials, luncheons, cocktail receptions, and other program activities. $235.

☐ **Strategic Simulations** (465 Fairchild Drive, Suite 108, Mountain View, CA 94043; 415-964-1353) has released two new strategy games. *The Road to Gettysburg* simulates command of an American army in the Civil War during the week-long campaign and the three-day Battle of Gettysburg. Players must deal with misinterpretation of orders, weather, fatigue, and disobedience by their corps commanders, whose abilities are based on historical factors. Includes player aid card, map board, and one hundred counters. Applesoft in ROM. $59.95. ☐ *Pursuit of the Graf Spee* recreates the engagements of the Ger-

LETTER PERFECT

T.M. LJK

WORD PROCESSING

ATARI 400/800 — APPLE II & II+

EASY TO USE — Letter Perfect is a single load easy to use program. It is a menu driven, character orientated processor with the user in mind. FAST machine language operation, ability to send control codes within the body of the program, mnemonics that make sense, and a full printed page of buffer space for text editing are but a few features. Screen Format allows you to preview printed text. Indented margins are allowed. Data Base Merge with **DATA PERFECT** by LJK, form letters, accounting files and mailing labels only with MAIL MERGE/UTILITY by LJK. **FEATURES** — Proportional/Incremental spacing * Right Justification * File Merging * Block movement * Headers * Footers * Print Multiple Copies * Auto Page Numbering * Scroll forward/backward * Search and Replaces * Full cursor control * Underlining * Boldface * Superscripts * Subscripts * Auto page numbering * Insert character/line * Delete character/line * Centering * Horizontal tabs/changeable * Multifunction format line (line spacing — left margin — page width — lines/page — change fonts — top/bot margin adjust) **MUCH MORE! $149.95**

ATARI VERSION 2.0 #2001

Uses proportional font, right justified with Atari 825/Centronics 737, 739 printers. Uses EPSON MX* Series + Graftrax/italicized font. Can mix type fonts on same page; mix boldface and enhanced font in same line with justification. Can be used with 16K Atari/400.*

"Compared to the price of many other word processors, this package is a steal. It does everything the advertisement claims and more. On top of this the software is very easy to use." **A.N.A.L.O.G. MAGAZINE**

APPLE VERSION 5.0 #1001

DOS 3.3 compatible — Use 40 or 80 column interchangeably (Smarterm — ALS; Videoterm-Videx; Full View 80 — Bit 3 Inc.; Vision 80 — Vista; Sup-R-Term — M&R Ent.) Reconfigurable at any time for different video, printer, or interface. **USE HAYES MICROMODEM II*** LCA necessary if no 80 column board, need at least 24 K of memory. Files saved as either Text or Binary. Shift key modification allowed. Data Base Merge compatible with **DATA PERFECT*** by LJK.

"For $150, Letter Perfect offers the type of software that can provide quality word processing on inexpensive microcomputer systems at a competitive price." **INFOWORLD**

DATA PERFECT T.M. LJK

APPLE & ATARI — introductory price

DATA BASE MANAGEMENT — **$99.95**

Complete Data Base System. User orientated for easy and fast operation. 100% Assembly language. Easy to use. You may create your own screen mask for your needs. Searches and Sorts allowed, Configurable to use with any of the 80 column boards of Letter Perfect word processing, or use 40 column Apple video. Lower case supported in 40 column video. Utility enables user to convert standard files to Data Perfect format. Complete report generation capability. **Much More!**

EDIT 6502 T.M. LJK

This is a coresident — two pass **ASSEMBLER, DISASSEMBLER, TEXT EDITOR, and MACHINE LANGUAGE MONITOR.** Editing is both character and line oriented. Disassemblies create editable source files with ability to use predefined labels. Complete control with 41 commands, 5 disassembly modes, 24 monitor commands including step, trace, and read/write disk. Twenty pseudo opcodes, allows linked assemblies, software stacking (single and multiple page) plus complete printer control, i.e. paganation, titles and tab setting. User can move source, object and symbol table anywhere in memory. Feel as if you never left the environment of **BASIC.** Use any of the 80 column boards as supported by **LETTER PERFECT,** Lower Case optional with **LCG.**

LJK DISK UTILITY — APPLE $29.95

This menu driven program allows the user to manipulate a variety of different file types. Binary, Text, and Source files may be easily converted into each other. The program may be used with **APPLESOFT*, VISCALC***, and other programs. These program files may be readily adapted for multiple use including editing with **LETTER PERFECT** word processings.

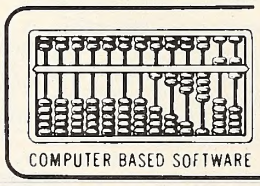
$29.95

MAIL MERGE/UTILITY — ATARI

This menu driven program combined with **LETTER PERFECT** allows user to generate form letters and print mailing labels. With the Atari, you may **CONVERT ATARI DOS FILES,** or Visicalc files compatible for editing with **LETTER PERFECT.** Utility creates Data Base files for Letter Perfect.

LOWER CASE CHARACTER GENERATOR — $24.95

Lower Case Character Generator for the Rev. 7, Apple II or II+ computers. When installed, this Eprom will generate lower case characters to the video screen. Lower case characters set has two dot true descenders. Installation instruction included. Manual includes listing of software for full support and complete instructions for shift key modification. Compatible with **LETTER PERFECT.**



COMPUTER BASED SOFTWARE — LJK ENTERPRISES INC.

LJK ENTERPRISES INC.
P.O. Box 10827
St. Louis, MO 63129
(314) 846-6124

DEALER INQUIRES INVITED

*Trademarks of: Apple Computer — Atari Computer — Epson America — Hayes Microcomputers — Personal Software — Videx — Bit 3 Inc. — M&R Ent. — Advanced Logic Systems — Vista Computers

man pocket battleship in the South Atlantic in 1939, attempting to destroy as many merchant ships as possible while evading Allied warships. Each ship rated for speed and strength; fuel restrictions, visibility and sighting system; tactical combat system accounts for ranging-in of guns, torpedo fire, and specific damage area. Hi-res nineteen-by-nineteen square grid map. Applesoft in ROM. $59.95.

☐ **LPA Tech** (Box 215, Springtown, PA 18081; 215-346-7757) has a package for Unity Mutual General agents, consisting of GPWL rates and values for ages twenty-five to sixty-five, three minimum deposit formats, illustrations for minimum pay, maximum pay, quick pay, and custom pay. Includes qualified and non-qualified flexible premium annuity illustrations. $750.

☐ **The Computerized Shopper** (3545 El Camino Real, Palo Alto, CA 94306; 415-856-7467) announces publication of the first automated cookbook. *Dinner on a Disk* features one hundred screen pages of proven recipes. No data input or technical knowledge required. $14.95.

☐ **S&H Software** (Box 5, Manvel, ND 58256; 701-696-2574) has released a 3.0 version of its *Universal Boot Initializer* for software vendors (see Marketalk Impressions, October 1981). Includes a directory title formatting procedure allowing individualization of each disk with a unique catalog title. Utility disk, two demo disks, and training disk keyed to manual. Either DOS. $49.95. ☐ *Amper-Sort/Merge,* a general purpose disk sort/merge utility for Apple DOS text files, features machine language read, sort, and merge routines. Will sort or merge five input files or pre-sorted files into a single file. $49.95.

☐ An enhanced version of the disk-based *Property Management System* for tracking income and expenses on rental property has been released by **Realty Software** (1116 'F' Eighth Street, Manhattan Beach, CA 90266; 213-372-9419). Will also handle single family homes and condominiums; displays late rents, year-to-date, monthly income and total building income displayed on screen or printed, operating statements for each building printed separately or combined to form a consolidated operating statement for all properties owned. $375.

☐ *Pro-Tech,* an all-steel construction security stand for the Apple II from **Seagull Enterprises** (11 Cove Avenue, Berkley, MA 02780; 617-823-9684), secures two disk drives and provides space for storing software, documentation, and manuals. Compatible with all monitors, does not have to be secured to desk, requires no disassembly or extension cables for disk drives. Computer can be turned on without unlocking. Quantity discounts; free demo with order of twelve. $135.

☐ **Chatsworth Data Corporation** (20710 Lassen Street, Chatsworth, CA 91311; 213-341-9200), manufacturer of optical mark sense card readers, announces a new intelligent interface for its OMR-500 card reader, now allowing it to operate under either CP/M or standard Apple control. Firmware stored on the interface board controls the reading and converts the card data to either ASCII or card image. Card reader can now perform functions using Basic, Fortran, or Cobol. $1,095. Field upgrade kit, $75.

☐ An enhanced version of *Typing Tutor* with a student monitoring feature for use in classroom instruction is being offered by **Microsoft** (400 108th Avenue, N.E., Bellevue, WA 98004; 206-454-1315). Instructors can monitor progress of up to forty-nine students simultaneously, review or remove individual student records. Student identity and access to records can be restricted to instructor. Three preprogrammed tests can be changed or added to. $24.95.

☐ The graphic drivers of **Computer Station** (11610 Page Service Drive, St. Louis, MO 63141; 314-432-7019) is now available in a *Combined Enhanced Graphics Software* package, allowing the user to put Apple hi-res pages on paper. Choice of picture or plot printing, normal or expanded sizes, and three different positions on the page. Compatible with a variety of printers and interface cards. Applesoft. $54.95. ☐ *Flexiterm* is an intelligent terminal package including a full text editor. Supports five interface cards in addition to the *Hayes Micromodem* and eighty columns with a Videx board and lower case using any of the lower case adaptors in forty-column display. Print option. $74.95.

☐ **Computer Camps International** (310 Hartford Turnpike, Suite D, Vernon, CT 06066; 203-871-9227) has added a new session in Connecticut beginning June 13. New courses for 1982 will include Hardware Architect and Assembly Language Programming, APL, Graphics, and Puzzle Solving. ☐ Their version of *Computer Camp for Adults* will be at Banner Lodge, Connecticut, for three weekends and one full week from July 9 through the week of August 22.

☐ **Corvus Systems** (2029 O'Toole Avenue, San Jose, CA 95131; 408-946-7700) is offering an interest floor plan to dealers featuring payment of interest on products for up to sixty days. Payments can be extended on unsold product; 25 percent curtailment due after ninety days, 25 percent more after one hundred fifty days, and the balance after one hundred eighty days. All products are now covered by free six-month warranty with six-month renewable and twenty-four-month full-term extensions available. Forty percent dealer margin on all warranties.

☐ *The Puzzler,* a wordsearch puzzle from **Tara** (Box 118, Selden, NY 11784; 516-331-2537), allows teachers to select from three puzzle sizes, hide words up to thirty-one letters long, up to ninety-nine words in a puzzle. Answer key; stores for reuse. No computer experience required. Eighty-column printer. $52.

☐ **Advanced Operating Systems** (450 St. John Road, Michigan City, IN 46360; 219-879-4693) has released *Busicomp,* an integrated business system to handle the accounting needs of small businesses. Six levels of security open up the eleven interactive program sections, covering accounts receivable, accounts payable, inventory control, general ledger, fixed assets, and payroll. Generates forty-one reports, formats designed to correspond to standard national DSF forms. $1,500.

☐ *Financial Management System III* is a high speed, user friendly record file system for home and business from **Computer Management Systems** (1039 Cadiz Drive, Simi, CA 93065; 805-526-0151). One to three key record entry, file chaining, unlimited number of records; features real time balance as entries are made, one hundred user defined codes. Applesoft in ROM. $120.

☐ **A.M. Electronics** (3446 Washtenaw Avenue, Ann Arbor, MI 48104; 313-973-2312) has announced the release of a new 5¼-inch disk drive. All units shipped with Apple-beige case, connecting cables, and ninety-day limited warranty. $395.

☐ **Southwestern Data Systems** (Box 582, Santee, CA 92072; 714-562-3670) is now offering *The Routine Machine,* a programming utility for putting machine language subroutines into an Applesoft program. Print using, array sort, array search, sound effects, more. No machine language programming knowledge required. $64.95. ☐ *Ampersoft Program Library, Vol. 1* is the first in a series of additional library disks to be used with *The Routine Machine.* The first disk includes array routines rename, delete, redimension, high speed read and write, many more. $49.95. ☐ *Financial Management System II,* designed for home and small business, allows entry of one month's checking, charge card, and cash accounts in minutes. Includes budget manager, account manager, auditor, check writer, more. $59.95.

☐ *PAL,* the first educational software to use a diagnostic/remediation approach to teaching reading, has been released by **USE** (2120-E Academy Circle, Colorado Springs, CO 80909; 303-574-4575). Keeps records for up to thirty students and updates progress; diagnoses specific deficiencies in reading ability and prescribes remediation exercises. Evaluates forty major reading skills and one hundred sixty subskills per grade level. Grades two through six. Applesoft. $99.95.

☐ *Computer Bits* is a weekly newspaper column syndicated by **The Register and Tribune Syndicate** (715 Locust Street, Des Moines, IA 50304) featuring the latest computer news, simple programs, product evaluations, Computer Assisted Learning information, more. Write to your local paper to request they carry it.

☐ Five new adventure series have been released by **M.A.C. Software** (Box 27, Chillicothe, OH 45601), in which the player must outwit his adversaries in classic fairy tales to obtain his

Don't buy Apple Software until you read this book.

Don't settle for manufacturers' promotional material.

THE BOOK is a complete **critical analysis** of most Apple software available. Experts **review, rate** and **evaluate** the programs on 11 separate points. It covers Games, Education, Business and Utilities as well as hardware. Over 500 programs are reviewed.

Don't buy software until you read this book. With the overwhelming array of programs available you can't afford not to consult ***THE BOOK.***

INTRODUCTION

This edition the THE BOOK OF APPLE COMPUTER SOFTWARE – 1982 combines previous editions (some re-written) and new articles, reviews and evaluations. Judging from the response accorded the first edition, which immediately sold out, there is a great need for a guide to the hundreds of programs that compete for the Apple owner's dollars. With the introduction of the Z80 card, choices get even harder concerning what to purchase; therefore, we dedicate this book to you, the consumer. We hope you will use it for a guide and as a reference to assist you in making intelligent and informed decisions when purchasing software.

Currently, the Apple Computer owner is presented with a bewildering selection of software from which to choose. On the one hand, this should please you in that, as the owner of probably the most popular micro-computer in the world, you have a wide and rapidly growing selection of software from which to choose. On the other hand, this wide and growing selection presents some problems. The vast majority of retail computer store staff people simply just do not have the time to adequately review each new piece of software that comes in their store. The problem is compounded if the new program is an extensive or complicated one, such as an accounting package or a word processing system, or a comprehensive data base management program. This does not mean that store personnel do not want to give you the best service possible; it's just that it is an almost impossible task. If you purchase software through the mail, the risks that you assume, without a reliable guide to assist you should be apparent.

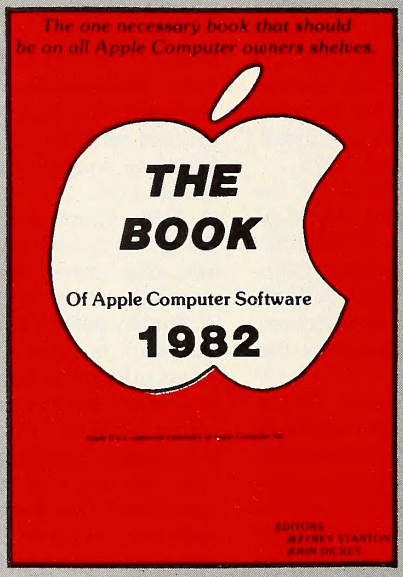
Other pitfalls await the uninformed buyer. For instance, in too many cases you cannot by the appearance of the package whether the program requires Integer Basic or Applesoft Basic or whether it needs 16, 32 or 48K of RAM. It is also often difficult to tell when you purchase a program on tape whether it can be transferred to disk or, if a disk program is purchased, whether it can be copied or not.

Another area that can present problems to the buyer is the similarity of software. A well-stocked computer store may possibly offer five different word processing packages, four assemblers, ten different adventure type games and/or several mail list programs, (the choices seem endless); all of which have obvious advantages and disadvantages as well as different prices.

The goal of "The Book" is to eliminate as many of these potential problem areas for the software buyer as possible.

We welcome any comments or criticisms from readers that will help us in reaching this goal.

*Obviously, Apple and Apple Computer Co. is mentioned many times throughout "The Book." Apple II is a registered trademark of Apple Computer Inc.



EDUCATION

BUSINESS

GAMES & ENTERTAINMENT

UTILITY PROGRAMS

TheBookCompany
16720 Hawthorne Blvd.
Lawndale, CA 90260
(213) 371-4012

NAME _____

ADDRESS _____

CITY _____ STATE _____ ZIP _____

CARD NUMBER _____ EXP _____

MasterCard ☐ Visa ☐

$19.95

goal. *Jack and the Beanstalk* introduces the series. $2 and blank disk, or $6. Disk one contains *The Three Pigs and the Wolf* and *The Thief of Baghdad*; disk two has *Little Red Riding Hood* and *Robin Hood*. Applesoft in ROM. $24.95 each.

☐ *The Sixth International Conference on Computers and the Humanities* will be held at **North Carolina State University** (Sarah K. Burton, Department of English, Raleigh, NC 27650; 919-737-3870), June 6 through 8, 1983.

☐ *Key Perfect* from **Micro-Sparc** (Box 325, Lincoln, MA 01773; 617-259-9710) generates a precision check code for each ten lines of Applesoft or Integer Basic, and each eighty bytes of machine language. Correct entries verified, incorrect exposed. Either DOS; 24K. $29.95. ☐ *Apple VIP II* lends machine language read and sort routines to the *VisiCalc Input Printer.* Prints grids in their column/row format, segments long formulas and prints in continuous blocks; grid option handles symmetrical spreadsheets up to twenty-six columns wide. $29.95.

☐ **Columbia Software** (5461 Marsh Hawk Way, Box 2235, Columbia, MD 21045; 301-997-3100) has introduced *Routeplanner,* a software package for planning trips with multiple stops. Accepts up to twenty-one route stops with coordinates obtained from maps, makes optional route adjustments, and computes the shortest distance. Database program stores up to four hundred map locations. $149.

☐ *Executive Briefing System* by Mitchell Kapor is being distributed by **Professional Software Technology** (180 Franklin Street, Cambridge, MA 02139; 617-497-2077). A presentation graphics package oriented to the business and professional user; creates, organizes, and displays "slide show" presentations on the Apple II. Run-time options of viewing time, multidisk wrap around, curtains, dissolves, spirals, and cuts may be specified prior to show or changed mid-show. Eight custom-designed fonts, true color text fonts, proportional characters, and text positioning flexibility. $199.

☐ *The Superplotter* from **Dickens Data Systems** (478 Engle Drive, Tucker, GA 30084; 404-923-3028) is a graphics package for applications in business, engineering, education, and math. Features pie graphs, standard bar charts, point and line graphs, mathematical function plotter, least squares polynomia curvefit generator, automatic graphics disk storage and recall, data file editor, overlay modes, a user tutorial, and keyboard image shapes that can be mixed with the user's own graphics displays. Allows specified input data points; graphics screen text editor can include text anywhere on graphics screen. $69.95.

☐ *The Model T Computer Slide System* from **Toucan** (1033 Battery Street, San Francisco, CA 94111; 415-392-2970) will produce color slides using an Apple and graphics tablet. Motorized 35mm camera controlled by computer; three character fonts in two sizes each. No programming necessary. Reproduction module, software, and graphics overlay, $3,495.

☐ **John Wiley and Sons** (605 Third Avenue, New York, NY 10158; 212-850-6000) has published *Apple Basic: Data File Programming,* by LeRoy Finkel and Jerald R. Brown, the first self-instructional manual to cover data file programming techniques on the Apple. Demonstrates how to program and maintain data files for billings, inventories, mailing lists, statistical information, more. $12.95, paper.

☐ **Smith Micro Software** (Box 604, Sunset Beach, CA 90742; 213-592-1032) announces the *Stock Portfolio System,* designed for the stock market investor. Controls cash, margin, and money market accounts; reporting of current portfolio values and the various account balances. Investment timing aids, reports for profit and loss, dividend income, and interest income and expense; designed to meet income tax filing requirements. $149.95.

☐ **Universal Data Systems** (5000 Bradford Drive, Huntsville, AL 35805; 205-837-8100) has announced a low-cost Bell-compatible 212A modem offering a savings to data communicators utilizing full duplex 300 and 1,200 bps channels in the same system. $695. ☐ Powered from the telephone line, the Model 212LP

manual answer unit requires no external AC power, offering full duplex 1,200 only bps asynchronous operation. Compatible with high speed 1,200 bps asynchronous channel of the Western Electric 212A; designed for desktop applications. $495.

☐ Scott Adams's *Adventure* series is being released with hires graphics, compressed and drawn using a palette of more than one hundred colors. The new programs support the Votrax *Type 'N Talk* voice synthesizer. *Adventureland,* the first in the series, is available now from **Adventure International** (507 East Street, Box 3435, Longwood, FL 32750; 305-862-6917). $29.95.

☐ **Bit 3 Computer Corporation** (8120 Penn Avenue South, Minneapolis, MN 55431; 612-881-6955) has introduced two new cards for the Apple. *The Dual-Comm Plus* provides two additional independent serial I/O interfaces, combining the features of the *Apple High Speed Serial Card* and the *Apple Communications Card* with handshaking. Thumbwheel switches select slot locations; sixteen switch selectable baud rates for each port range from 50 to 19,200 baud. On-board firmware provides printer and upper/lower case terminal/modem support; forty-column display or eighty-column card. $239. ☐ The *Memory Plus* 16K RAM expansion card adds 16K memory to a 48K Apple. Works with Pascal, DOS, CP/M, Fortran, Cobol, Pilot, *VisiCalc,* Integer, and Applesoft Basic. Three LEDs indicate memory select and read/write protect status. Firmware socket can hold alternate Apple Monitor ROM of 2716 EROM program. $149.

☐ *Bibliotek,* first software offering from **Scientific Software Products** (3171 Donald Avenue, Indianapolis, IN 46224; 317-299-0467) is designed to automate literature citation indexing for scientists, small libraries, and academicians engaged in research. Citation entry, modification, deletion, searching, sorting, and printing controlled through keyboard entries; five hundred entries per two-disk bibliography. Two disk drives, printer. $300.

☐ *SATN,* an applications journal providing additional customer support for *VisiCalc* users, is published bimonthly by **Software Arts** (Box 815, Quincy, MA 02169). Features information applicable to all versions of the program, each issue concentrating on one or more functions of *VisiCalc.* Six issues for $30.

☐ An aid to selling micro hardware and software, *CompuVision* is a point-of-purchase tool from **Avion Communications** (50 West Brokaw Road, Suite 64, San Jose, CA 95110; 408-295-2433) that allows customer random access of videotape messages or demonstrations of various software and systems, eliminating need for dealer demo. Plexiglass and formica stand-up console with VHS player, monitor, and interface board. $1,995.

☐ *The Personal Investor* from **PBL** (605 Harmony Circle Drive, Wayzata, MN 55391; 612-473-3769) connects with the Dow Jones News and Quotes Service to update a portfolio and retrieve business news. Generates four reports; sales and purchase expenses accounted for, stock splits calculated automatically and recorded, quotations include bid, ask, yesterday's close, today's open, high, low, last price, volume, and current dividend yield. Forty or eighty-column format. $95.

☐ *The Pascal Soft Disk Emulator* from **Legend Industries** (Box 112, Pontiac, MI 48056) supports the simulation of fast access disk drive units for the storage and retrieval of standard Apple Pascal 1.1 disk files. Introduces no hardware dependencies with respect to hardware use and development. Requires at least one Legend 64KC or 128KDE card and the Apple Pascal 1.1 Operating System. $49.95.

☐ *Federal Reports,* giving access to articles and reports on recent legislation, and *Future File,* including articles and interviews with futurists in business, politics, and the military discussing the eventual impact of new technology, are two new information sources in the national videotex service of **Compuserve** (5000 Arlington Centre Boulevard, Columbus, OH 43220; 614-457-8600). Subscribers may access for $5 per hour weekday evenings and all day Saturday, Sunday, and holidays.

☐ *Test Writer,* a program for the preparation of multiple choice and other types of tests and examinations, is available from **Persimmon Software** (502 Savannah Street, Greensboro, NC 27406; 919-275-5824). Upper and lower-case printout, instructions to the test taker, computation of point values, and complete answer key. $35.

☐ *The Graphics Magician,* by Chris Jochumson, David Lubar, and Mark Pelczarski, includes machine language subroutines that can be attached to your own programs for arcade-quality animation, storage for hundreds of pictures on a disk, and extended shape table features. Use of all colors and angle preservation on rotation and scaling; editors for preshifted shapes, paths, and animation of up to thirty-two independent objects. Nonprotected disk, in keeping with the new policy of **Penguin Software** (1206 Kings Circle, West Chicago, IL 60185; 312-231-0912). $59.95.

☐ The new *Omniware* software line from **Educational Computing Systems** (106 Fairbanks Plaza, Oak Ridge, TN 37830; 615-483-4915) features *Omnifile,* a full-feature file manager and report generator. $49.95. ☐ *Omnitrend* is a multiple regression trend analysis program with statistical calculations and hi-res graphics. $44.95. ☐ *Omnigraph* is a flexible data plotting program that allows X/Y plots, bar charts, and pie charts. $39.95. ☐ The *Omnitest* educational system allows you to build question and answer files on any subject and review in drill or quiz game mode. $29.95. Introductory prices until May 1.

☐ **WIDL Video** (5245 West Diversey, Chicago, IL 60639; 312-622-9606), publisher of the Apple Directories, has released the second edition of *The Apple II Blue Book,* a master directory of software, hardware, peripherals, and information. Includes more than five thousand software and hardware listings and more than seven hundred fifty producers; resource section lists reference manuals, publications, user groups, time sharing systems, more. $24.95.

☐ **Earthware Computer Services** (Box 30039, Eugene, OR 97403; 503-344-3383) has released an educational version of their educational game, *Volcanoes.* Includes two documentation booklets, two disks, and a teacher's manual. Available only to educational institutions. $49.50.

☐ *Word Processors and Information Processing,* by Dan Poynter, available from **Para Publishing** (Box 4232-88, Santa Barbara, CA 93103; 805-968-7277), aids in purchasing word processing equipment, supplies, and services. Resource section lists associations, publications, products, and services available to support the needs of the user. $11.95.

☐ *The Magic Keyboard,* latest relative of the *Paddle-Adapple* from **Southern California Research Group** (Box 2231, Goleta, CA 93118; 805-685-1931), reassigns keys to a 10-keypad, hex keypad or programmer's keypad without extra hardware. Nine dip switches change standard Qwerty keyboard layout to Dvorak simplified, American simplified, alphabetical keyboard, and others. Will reencode with software. User installed; includes key decals. $89.95.

☐ Richard Garriott, also known as Lord British, will be an instructor at this summer's **Computer Camp** (1235 Coast Village Road, Suite G, Santa Barbara, CA 93108; 805-967-2011). The author of *Ultima* will be demonstrating the fine points of game theory, subroutines, and machine language at the Santa Barbara location in July, with a possible seminar later at Lake Tahoe if his schedule permits. ☐ A kid can now go to camp—and a computer retailer can take a 100 percent mark-up—by picking up an application packet at a computer store. Application and $100 deposit are left with dealer, who forwards the application to Santa Barbara. Finalized list of participating retailers to be announced.

☐ The *Supercolor* card, compatible only with the RGB analog type color monitor, both from **Electrohome** (809 Wellington Street N., Kitchener, Ontario, N2D 4J6), expands the video capability of the Apple to provide a spectrum of 256 different programmable colors, with any sixteen software selectable. Features pure white text capability in hi-res mode, standard Apple colors at power up, one plug-in module into peripheral slot. $350.

☐ The version 2.1 update of the **Solarsoft** programs is now

Apple II

Quality Disk Software from SPECTRUM

Apple II

PERSONAL FINANCE MASTER:

The premier personal and small business financial system. Covering all types of accounts 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) $74.95

COLOR CALENDAR: $29.95

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 from 8:00 A.M. to 5:30 P.M.

BUSINESS SOFTWARE:

Both Programs $249.95

A user-friendly yet comprehensive double-entry accounting system employing screen-oriented data input forms, extensive error-trapping, data validation and machine language 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.
- Extensive check register management system (similar to our PFM).
- Prints checks and mailing labels.
- Produces these reports:
  - Transactions Journal
  - Account Ledgers
  - Income Statement
  - Balance Sheet
  - Account Listings

Requires Apple II, 48K RAM, 1 disk drive. $149.95

ACCOUNTS RECEIVABLE: A flexible system with these features:

- Up to 500 accounts and up to 500 invoices per diskette.
- Prints invoices and customer statements & address labels.
- Interfaces to General Ledger.
- Interactive screen-based invoice work sheet.
- Produces these reports:
  - Aged Receivables
  - Sales Analysis
  - Customer Listings
  - Invoice Search

Requires Apple II, 48K RAM, 2 disk drives $149.95

Both Programs $249.95

UNIVERSAL BUSINESS MACHINE:

An electronic spread sheet structured around a 100 row × 20 column table. User defines row and column names and equations forming a unique template. Table elements can be multiplied, added, subtracted, divided, summed, averaged and accumulated. Hundreds of unique templates can be created, used, stored and recalled for later use. Supplied with 8 standard templates ready for use covering these subjects.

- Cash Flow Analysis
- Proforma Profit & Loss
- Proforma Balance Sheet
- Real Estate Investment
- Sales Forecaster
- Source and Use of Funds
- Job Cost Estimator
- Inventory Analysis

Price (Apple II, 48K) $89.95

ELECTRONICS SERIES VOL I & II:

Entire Series $259.95

LOGIC SIMULATOR: SAVE TIME AND MONEY. Simulate your digital logic circuits before you build them. **CMOS, TTL,** or whatever, if it's digital logic, this program can handle it. The program is an interactive, menu driven, full-fledged logic simulator capable of simulating the bit-time response of a logic network to user-specified input patterns. It will handle up to 1000 gates, including **NANDS, NORS, INVERTERS, FLIP-FLOPS, SHIFT REGISTERS, COUNTERS** up to 20 user-defined random, or binary input patterns. Accepts network descriptions from keyboard or from **LOGIC DESIGNER** for simulation (48K) $159.95

LOGIC DESIGNER: interactive **HI-RES** graphics program for designing digital logic systems. Draw directly on the screen up to 10 different gate types, including **NAND, NOR, INVERTER, EX-OR, T-FLOP, JK-FLOP, D-FLOP, RS-FLOP, 4 BIT COUNTER** and **N-BIT SHIFT REGISTER.** User interconnects gates using line graphics commands. Network descriptions for **LOGIC SIMULATOR** generated simultaneously with the **CRT** diagram being drawn $159.95

MANUAL AND DEMO DISK: Instruction Manual and demo disk illustrating capabilities of both programs) $29.95

ELECTRONIC SERIES VOL III & IV:

Entire Series $259.95

CIRCUIT SIMULATOR: Tired of trial & error circuit design? Simulate & debug your designs before you build them! With **CIRCUIT SIMULATOR** you build a model of your circuit using **RESISTORS, CAPACITORS, INDUCTORS, TRANSISTORS, DIODES, VOLTAGE** and **CURRENT SOURCES** and simulate the waveform response to inputs such as **PULSES, SINUSOIDS, SAWTOOTHS,** etc... all fully programmable. The output is displayed as an **OSCILLOSCOPE-STYLE PLOT** of the selected waveforms or as a printed table of voltage vs time. Handles up to 50 nodes and 100 components. Requires 48 **RAM** $159.95

CIRCUIT DESIGNER: Interactive **HI-RES** graphics program for designing electronic circuits. Draw directly on the screen up to 10 different component types, including those referenced above. Component interconnect list for **CIRCUIT SIMULATOR** generated automatically, Requires 48K **RAM** $159.95

MATHEMATICS SERIES:

Entire Series $49.95

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. Printer, Disk, I/O routines. $19.95

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 **DERIVATIVE,** determine the **ROOTS, MAXIMA, MINIMA, INTEGRAL VALUE** $19.95

MATRIX: A general purpose, menu driven program for determining the **INVERSE** and **DETERMINANT** of any matrix, as well as the **SOLUTION** to any set of **SIMULTANEOUS LINEAR EQUATIONS** $19.95

3-D SUFAACE 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 $19.95

ACTION ADVENTURE GAMES:

Entire Series $29.95

RED BARON: Can you outfly the **REO BARON?** This fast action game simulates a machine-gun **OOGFIGHT** between your **WORLD WAR I BI-PLANE** and the baron's. You can **LOOP, OIVE, BANK** or **CLIMB**-and so can the **BARON.** In **HI-RES** graphics plus sound $14.95

BATTLE OF MIDWAY: You are in command of the **U.S.S. HORNETS' OIVE-BOMBER** squadron. Your targets are the Aircraft carriers, Akagi, Soryu and Kaga. You must fly your way through **ZEROS** and **AA FIRE** to make your **DIVE-BOMB** run. In **HI-RES** graphics plus sound..... $14.95

SUB ATTACK: It's April 1943. The enemy convoy is headed for the **CONTROL SEA.** Your sub, the **MORAY,** has just sighted the **CARRIERS** and **BATTLESHIPS'.** Easy pickings. But watch out for the **DESTROYERS** - they're fast and deadly. In **HI-RES** graphics plus sound. $14.95

FREE CATALOG-All programs are supplied on disk and run on Apple II w/Disk & Applesoft **ROM** Card and require 32K **RAM** 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
142 Carlow, P.O. Box 2084
Sunnyvale, CA 94087

master charge

VISA

FOR PHONE ORDERS:
(408) 738-4387
DEALER INQUIRIES INVITED.

available from **Solarsoft** (Box 124, Snowmass, CO 81654; 303-927-4411). Revisions made are to the *Sunpas* and *Sunop* files; *Tswing* files are in a new format due to weather files. Version 3.0 will be released concurrently with Volume III of the *Los Alamos Passive Design Handbook.*

☐ **Jo Ann Brissenden** (Box 1484, Lafayette, CA 94549; 415-283-5990) would like to start a California Apple telecommunications club. Interested users can write to her or leave a message on her computer.

☐ **Dynacomp** (1427 Monroe Avenue, Rochester, NY 14618; 716-442-8960) has introduced five new programs. *The Olde Gin Parlour* is a hi-res card game. $18.95. ☐ *Util* is a disk utility. $19.95. ☐ *Shape Magician* is a shape table generating utility. $29.95. ☐ *Softnet* is a pipeline analysis program. $129.95. ☐ *Payfive* is a payroll management package. $149.95.

☐ *Travel Sensor Chess,* combining varied game strength with a physically compact sensor unit, is the latest chess computer from **SciSys** (489 Fifth Avenue, New York, NY 10017; 212-682-7600), the smallest currently on the market. Sensorboard automatically enters all moves as pieces are advanced; LEDs illuminate coordinates of the computer's next move. Eight levels of play; board positions can be stored, allowing continuation of match after switching off. $50.

☐ **W. H. Nail Company** (275 Lodgeview Drive, Oroville, CA 95965) is distributing the *Egbert RTTY Program* for the amateur radio user. Cassette ports connect directly to the transmitter/receiver; requires no hardware interface. Game port-driven push-to-talk, 60, 67, 75, and 100 wpm Baudot and 110 Baud ASCII, type-ahead buffer, canned messages, more. ROM Applesoft; either DOS. $42.45.

☐ Supersonic tanks, heat-seeking missiles, explosive balloons, a flock of airborne neon tetras, and the bluebird of nastiness are the resident menaces of Tony Suzuki's *Star Blazer* from **Broderbund** (1938 Fourth Street, San Rafael, CA 94901; 415-456-6424). Your fighter-bomber is charged with five different missions that tax your ingenuity and ability to evade and attack. $31.95. ☐ *General Ledger with Payables,* by Hal Faulkner, will handle normal general ledger and payable functions for financial accounting, plus some management accounting functions. Capacity of two thousand ledger accounts, unlimited journal entries, two hundred open payable accounts, nineteen hundred open invoices, values up to $9,999,999,999.99, one thousand checks per disk, and user definable account numbers up to ten digits. Documentation segmented with tabs which correspond to numbered functions on the screen. 64K, two disk drives, printer. $495.

☐ *6502 Assembly Language Subroutines,* by Lance A. Leventhal and Winthrop Saville, presents an overview of assembly language programming for the 6502 microprocessor and a collection of more than forty subroutines that can be used in applications and as guidelines for complex programs. Code for array and bit manipulation, code conversion, interrupt service routines, and others. First title in a new series on assembly language subroutines from **Osborne/McGraw-Hill** (630 Bancroft Way, Berkeley, CA 94710; 415-548-2805). $12.99, paper.

☐ *Apple Alarm,* from **Andent** (1000 North Avenue, Waukegan, IL 60085; 312-244-0292), functions as a heat, smoke, motion, or moisture detector; attaching floor mat, door switch, fire alarm, or other on-off sensor to paddle buttons and causing your Apple to sound an alarm or keep time from the moment triggered. No specialized equipment necessary. $20 ☐ *Hypnosis,* an aid to suggestive relaxation, behavior modification, and trance induction, utilizes visual and auditory sensory stimulation to induce and potentiate hypnotic states. Forty-eight thousand settings. $20.

☐ *The Gutenberg Word and Print Processing Program* from **Micromation** (1 Yorkdale Road, Suite 406, Toronto, Ontario, Canada M6A 3A1; 416-781-6675) now has a complete print driver for the Epson printer, permits the user to define an unlimited number of character sets, and reads DOS 3.3 binary picture files. $315.

☐ Rodnay Zaks's *From Chips to Systems: An Introduction to Microprocessors* explains the operation of a computer system, how the components are interconnected, and how micros were invented. No preliminary knowledge of microprocessors required. Published by **Sybex** (2344 Sixth Street, Berkeley, CA 94710; 800-227-2346). 576 pages. $14.95.

☐ *Introduction to Microprocessors: Experiments in Digital Technology,* by Noel Smith, is a how-to guide to the use of integrated circuits, featuring thirty-five experiments encouraging their use in electronic projects. Illustrated with one hundred fifty schematic diagrams illustrating key design principles, from gates and timers through major integrated circuit families, to the microprocessor and its use in microcomputers. From **Hayden Book Company** (50 Essex Street, Rochelle Park, NJ 07662; 800-631-0856.) 176 pages. $10.95.

☐ Integrating all database descriptions from previous issues with databases since available on-line, the *Directory of Online Databases, Volume Three,* from **Cuadra Associates** (1523 Sixth Street, Santa Monica, CA 90401; 213-451-0644) supercedes the previous volumes. Covers databases available through international telecommunications networks, and those accessible through on-line services connected to networks serving only one country or a limited set of countries. New index identifies the telecommunications network connections associated with each on-line service. $29.95.

☐ *The Hardisk Accounting System,* developed by **Great Plains Computers** (113 Broadway, Fargo, ND 58102; 701-293-8483) is an expandable double-entry business system consisting of general ledger, accounts receivable, accounts payable, inventory, point of sale, sales order entry, purchase order entry, payroll, fixed asset management, and mailing labels. All modules interactive, with complete audit trails. From $395 to $595 per module.

☐ *Applefest/Boston,* produced by **Northeast Expositions** (824 Boylston Street, Chestnut Hill, MA 02167; 800-343-2222), will be held May 14 to 16, 1982, at Boston's Hynes Auditorium, concurrent with the annual meeting of the *International Apple Corps.* It is the largest Apple computer show in the country, featuring over three hundred displays and booths, and every major Apple product, program, and accessory on display and for sale. $6 per day, $10 for two days, $15 for three days.

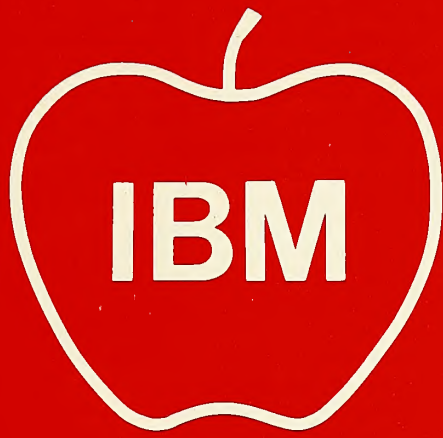
☐ A tutorial aid package designed to take the first time user through the *DB Master* file management system has been developed by **Erich A. Schmitt Associates** (96 Fordham Drive, Aberdeen, NJ 07747; 201-566-4594). Acquaints the user with responses required in file creation, searching, and report definition and production. One sample file used throughout. $24.95.

☐ **Island Graphics** (Box V, Bethel Island, CA 94511; 415-684-2664) has released *The Illustrator,* an intergrated color graphics tool kit for the computer artist, featuring a palette for mixing seventeen billion color combinations. Pictures may be saved or moved from disk to disk and paints stored for touch-ups. Operations include draw, fill, magnify, frame, box, circle, lines, slide pictures, change colors, invert colors, complement colors, and add text. User definable brushes. $95.

☐ The *Model 85 Digital Memory Oscilloscope* from **Northwest Instrument Systems** (Box 1309, Beaverton, OR 97075; 503-297-1434) fits as a module into the Apple using the display and keyboard as an oscilloscope screen and control panel, the Apple for waveform processing, and disk memory for waveform storage. Performs signal averaging and DVM readout, fast Fourier Transform, auto- and cross-correlation, power density spectra, and integration and differentiation. Supports hard copy output of the waveform display with user-added comments. $995.

☐ *Adventure to Atlantis,* the latest adventure from **Synergistic** (830 North Riverside Drive, Rewnton, WA 98055), has been shipped to distributors and is beginning to appear at retailers. Unfortunately, there was a flaw in the master for one-third of the disks, as a result of which the game will not save while the player is on an island. The company is replacing all unsold disks; anyone who has bought one may send it to the address listed above for their free replacement.

☐ *Doctor's Office Companion* is now available from **High Technology Software** (Box 14665, 2201 NE 63rd, Oklahoma City, OK



Write IBM® on your Apple®.

Our UCSD p-System™ lets you write applications on your Apple II that will run without recompilation on the IBM Personal Computer.

And that's not all.

Your UCSD p-System applications will also run on Commodore, Philips, Radio Shack and Texas Instruments — in fact, on just about any popular microcomputer you can name. Because SofTech Microsystems' UCSD p-System is the only microprocessor-independent operating system for 8-bit and 16-bit computers.

Our enhanced UCSD p-System gives you everything you need to develop genuinely portable, compact applications for almost every microcomputer owner. It allows you to work in any combination of UCSD Pascal™, BASIC and assembly language. And it provides support for dynamic memory management and multitasking, with a full arsenal of enhancements such as a powerful Screen Editor, Native Code Generators, TURTLEGRAPHICS and XenoFile™, for easy access to CP/M® disks and files.

So if you're interested in maximizing your hardware investment, developing portable applications and adding thousands of users to your customer list, contact SofTech Microsystems today.

Our UCSD p-System is available for distribution licensing as well as single copy purchase. Visa and MasterCard orders are welcome.

Write or call:
SofTech Microsystems, Inc.
9494 Black Mountain Road,
San Diego, CA 92126.
(714) 578-6105,
TWX: 910-335-1594.

SOFTECH
MICROSYSTEMS
A SUBSIDIARY OF SOFTECH

73113; 405-478-2105). Patient data is gathered for automatic printing of insurance forms, insurance and cash billing, definition of up to fifteen income categories and ten "street" accounts. Customizable CPT file, monthly aging report, history report of all transactions for any guarantor. Requires Corvus or four floppy disk drives, 132-column printer. $995.

☐ **MPC Peripherals Corporation** (9424 Chesapeake Drive, San Diego, CA 92123; 714-278-0630) has released *PROM-IT EDS,* an EPROM programmer for the Apple. Programs 8K, 16K, and 32K EPROMS by changing a personality module. Textool zero insertion force socket and EPROM power down switch utilized for removal and insertion of EPROM without powering down. Memory mapped space permits execution of routines by the 6502; hex files downloaded from any system with RS-232 port. $129.50. ☐ The *AP-SIO,* an asynchronous serial input/output interface card with auto LF/no LF, strip incoming LFs, half duplex/full duplex, and lower to upper case conversion/no conversion firmware options, features a serial communication protocol jumper block, eliminating non-standard wired cables and jumpered circuit boards. Plug-in compatibility with all operating systems; two-year warranty. $129.50.

☐ *The Software Expo,* to be held May 1, 1982, at the **Software Store** (11768 West Pico Boulevard, Los Angeles, CA 90064), will feature exhibits and demonstrations by software distributors and user groups. Authors and programmers will be on hand to chat and sign autographs. The expo will also raise money for the Muscular Dystrophy Association through entrance donations and an arcade-type game competition in which participants will receive pledges of contributions to MDA based on their points scored. Prizes will be awarded for highest score and most money pledged.

☐ *The Applegrator II,* a precision laboratory integrator, is now available from **Dynamic Solutions** (61 South Lake Avenue, Suite 309, Pasadena, CA, 91101; 213-577-2643). Designed for commercial testing labs with heavy analysis requirements; features specific applications software for chromatography, spectroscopy, colorimetry, and flow measurement, plus general purpose software for pulse integration and data acquisition. Peak detection routines, sequence of operations and reports customized to specifications. Samples waveforms at rates up to 20Khz and stores up to ten thousand data points; data reviewable in expanded or compressed form. No programming experience required. Includes Apple, monitor, A/D converter, and precision timer; VisiTrend and VisiPlot free with purchase 'til June 1. $6,480. Upgrade, $3,980.

☐ **Apple Computer** (10260 Bandley Drive, Cupertino, CA 95014; 408-996-1010) has released new versions of all software for the Apple III. Anyone possessing one of the original release models of the machine (the bad Apples, so to speak) can have it replaced free along with a new batch of software. Contact your dealer for details.

☐ **SouthWest EdPsych Services** (Box 1870, Phoenix, AZ 85001; 602-253-6528) has developed the *Cassette Interface,* a hardware device to start and stop most cassette machines. Plugs into game paddle port and cassette remote plug. $49.95.

☐ *Phone Chronicle,* a new accessory for small businesses, lawyers, accountants, and consultants, from **Sycon** (3040 Scott Boulevard, Santa Clara, CA 95050; 408-727-2751), logs all outgoing calls, noting date, phone number, time, and length of call, and three-digit individual employee account/code. One hundred call on-board memory. PC board, phone plug, cable, disk, and manual. $395.

☐ **Edu-Ware** (Box 22222, Agoura, CA 91301; 213-706-0661) has released 3.0 versions of *Edu-Ware Fractions* and *Edu-Ware Decimals,* featuring a "learning manager" that enables a student, teacher, or parent to control sequence of units, minimum/maximum values in examples, and test scoring criteria. Hi-res color, optional sound effects, and two upper and lower case font sizes. $49 each.

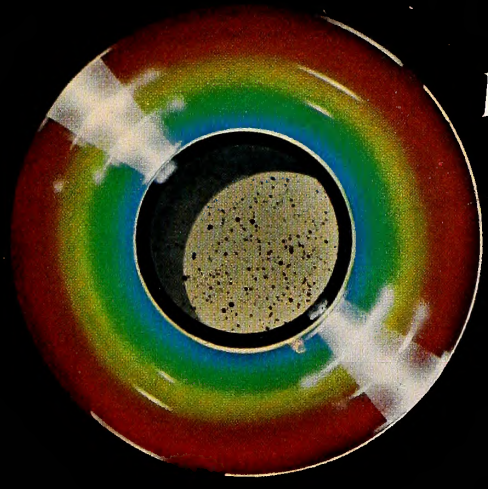
☐ *A Guide to Investor Software* from **Clark Software** (1730 West Mulberry Street, Shamokin, PA 17872; 717-644-1392) lists the names, addresses, and phone numbers of all companies offering software for investors. The programs are described in detail, including price and required configuration. Sections on database services, associations, books, and computer games. $5.95.

☐ **Faxscan** (3148 Dorf Drive, Dayton, OH 45418; 513-263-8475) has released their *Model AD-3 Real World Apple Interface Card.* Analog to digital conversions and vice versa, sixteen user definable I/O ports, two multimode timers. $200. ◨

On-Line Systems PULLS AHEAD
WITH NEW PRODUCT UPDATES
Designed to *BEAT* the COMPETITION

ON-LINE

APRIL RELEASES



EXPEDITER II+

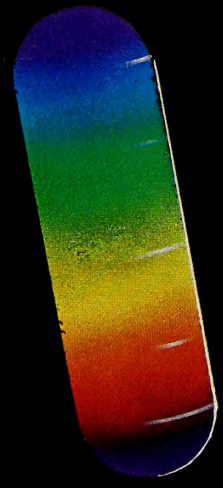
A new completely redesigned Expediter made to keep memory needs DOWN!

WITH
New Features
and
New Documentation

GENERAL MANAGER 1.5

A new release of our Powerful Data Base Manager

Includes new applications
AND
New easier to read documentation



SEE THEM *at your local computer store!*

Also . . . MARAUDER

The *ultimate* in Alien Invasion

They have landed and are taking over the city. Steadily they are making their way across the city, destroying everything in their paths. The town has been evacuated and your regiment has retreated leaving you, alone in the city, at the mercy of the aliens.

The aliens have you surrounded, and laser shots fly from all directions. Your movements are confined but you haven't given up. If you're going to live, you'll have to concentrate on where the shots are coming from and where you're going because if you don't, you'll get caught in the **CROSS FIRE**.

CROSS FIRE is a unique new game by JAY SULLIVAN featuring HI-RES graphics and sound, smooth quick animation, and some of the best arcade challenge available anywhere. **CROSS FIRE** runs on any 48K APPLE II/II PLUS DOS 3.2 or 3.3 and is available now for $29.95 on disk from your local computer store or you may order directly from.....

ON-LINE systems

36575 Mudge Ranch Road - Coarsegold, CA 93614 - 209-683-6858

ADD $1.00 FOR DIRECT ORDERS **VISA, MASTER CHARGE, C.O.D. or CHECKS ACCEPTED**

ON-LINE SYSTEMS introduces arcade gaming as an art form. THRESHOLD, by WARREN SCHWADER and KEN WILLIAMS, features fast smooth animation, HI-RES graphics, and more challenge than you'll find in any other arcade game on the market.

THRESHOLD is an arcade game with alien attackers galore. In fact, there are more monsters out there than we expect you'll ever see. THRESHOLD was designed to be an arcade game that you won't get bored of, and that means a constantly changing game with a graduated skill level, but more than that, THRESHOLD means constant fun.

THRESHOLD runs on any 48K Apple II or II Plus DOS 3.2 or 3.3. Available now for $39.95 on disk from your local computer store or you may order directly from. . . . .

ON-LINE systems

36575 Mudge Ranch Road - Coarsegold, CA 93614 - 209-683-6858

ADD $1.00 FOR DIRECT ORDERS

Visa, Master Charge, C.O.D. or Checks Accepted

Two New Additions to our Best Selling Adventure Series

Hi-Res Adventure # 3
CRANSTON MANOR
by Ken Williams & Harold DeWitz

ULYSSES and the Golden Fleece

HI-RES ADVENTURE #4
Our First 2 Disk Adventure!

BOB DAVIS, Co-ordinator for TIME ZONE, the "Ultimate" HI-RES Adventure project, has teamed up with KEN WILLIAMS to bring the high adventure of ancient mythology to your APPLE. "CAST-OFF" with ON-LINE SYSTEMS and ULYSSES, and see if you can find the Golden Fleece.

ULYSSES is a HI-RES Adventure epic set in ancient Greece.

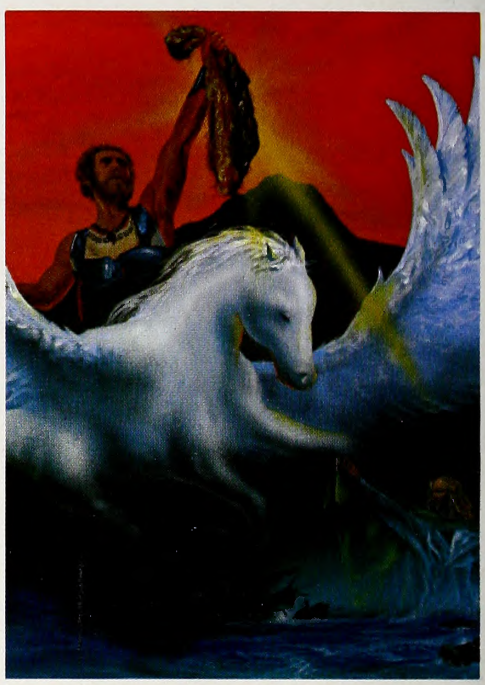
PRICE $34.95

CRANSTON MANOR

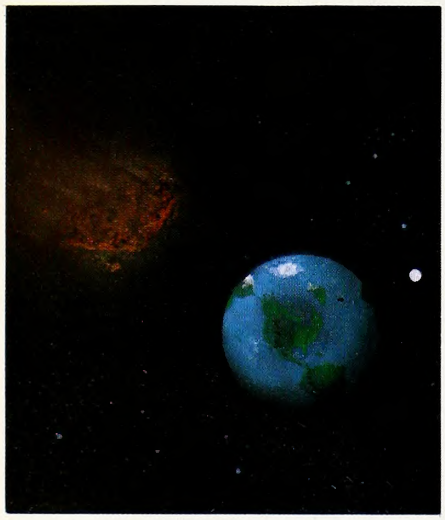
HI-RES ADVENTURE #3

In this HI-RES Adventure, you explore the estate of Old man Cranston and find the riches that are hidden throughout the manor and surrounding grounds. Be careful! Rumor has it that the disembodied spirit of the old man roams the estate guarding his fortune.

CRANSTON MANOR is a HI-RES Adventure game set in contemporary times in Coarsegold California.
PRICE $34.95

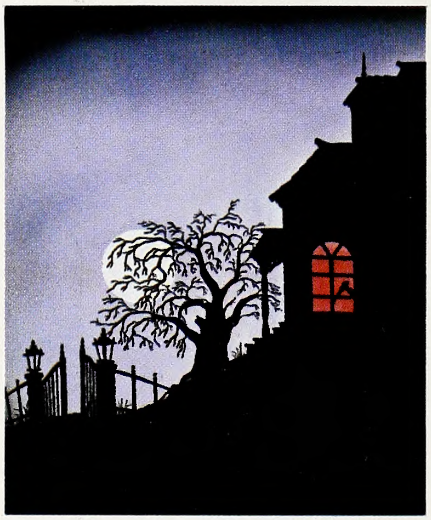


Hi-Res Adventure #4
ULYSSES
by Bob Davis & Ken Williams



Hi-Res Adventure #0
MISSION ASTEROID
by Ken & Roberta Williams

Written as an introduction to the HI-RES Adventure family of games, MISSION: ASTEROID is designed to acquaint beginning adventure players to the wonderful world of HI-RES Adventures.
PRICE: $19.95



Hi-Res Adventure #1
MYSTERY HOUSE
by Ken & Roberta Williams

When you enter the house, you are pulled into the mystery and intrigue as your companions are murdered one by one. Be careful, you may be next! Can you solve the mystery and leave the house alive? The secret passage way may lead you to the answer.
PRICE: $24.95

Hi-Res Adventure #2
WIZARD AND THE PRINCESS
by Ken & Roberta Williams

After one year on the best sellers list, this adventure is still going strong. To win this game you will have to cross deserts, chasms and oceans in your search for the kings daughter. If you find her, half of the kingdom of Serenia is yours.
PRICE: $32.95

ON-LINE systems

Visa, Master Charge, COD or Checks Accepted
For direct mailing add $1 for shipping.

36575 Mudge Ranch Road • Coarsegold, CA 93614 • 209-683-6858

starring JULIUS CAESAR - CHRISTOPHER COLUMBUS - CAVEMEN - ROBIN HOOD - BENJAMIN FRANKLIN - CLEOPATRA and YOU
With a Cast of Thousands
executive producer KEN WILLIAMS • produced by BOB DAVIS
☆ ☆ ☆ written and directed by ROBERTA WILLIAMS ☆ ☆ ☆
project development: TERRY PIERCE, ERIC GRISWOLD, RORKE WEIGANDT, JEFF STEPHENSON
Admission $99.95

UA	ULTIMATE ADVENTURE
	48K APPLE II or II PLUS

ON-LINE systems
36575 Mudge Ranch Road • Coarsegold, CA 93614 • 209-683-6858
Add $1.00 for shipping VISA, Mastercard, C.O.D. or checks accepted

mouskattack

by JOHN HARRIS
and
KEN WILLIAMS

RAT ALLEY is infamous among the legions of the plumber. Rumor has it that extremely large rodents live in the maze of tunnels that make up the area. The last plumber that ventured into RAT ALLEY disappeared without a trace. (The only clues they found as to his demise were a hard hat bent to look like Mickey Mouse ears, and a pipe wrench with a bite out of it.)

You are Larry Bain, ace Plumber, of Rodent Rooter plumbing service. Your job is to install pipe through the many levels of RAT ALLEY.

MOUSKATTACK is an unbeatable blend of quick thinking arcade action for one or two players.

Available on disk for $34.95. MOUSKATTACK runs on any 48K APPLE II or II+ DOS 3.2 or 3.3 and 32K disk ATARI 400/800.

Get MOUSKATTACK at your local computer store or order directly from:

ON-LINE systems

36575 Mudge Ranch Road • Coarsegold, CA 93614 • 209-683-6858

Visa, Mastercharge, Checks, COD

Add one dollar for shipping

Ulysses

and the Golden Fleece

ON-LINE systems

HI-RES ADVENTURE #4 *by Bob Davis & Ken Williams*

209-683-6858 • Visa, Master Charge, C.O.D. or Checks Accepted.

The Ultimate Experience in Alien Invasion . . . .

MARAUDER

AN OUTSTANDING
ARCADE ENDEAVOR
BY
RORKE WEIGANDT
AND
ERIC HAMMOND

MARAUDER runs on any 48K Apple II / II Plus with DOS 3.2 or 3.3 and is available now for $34.95 on disk from your local computer store or order directly from

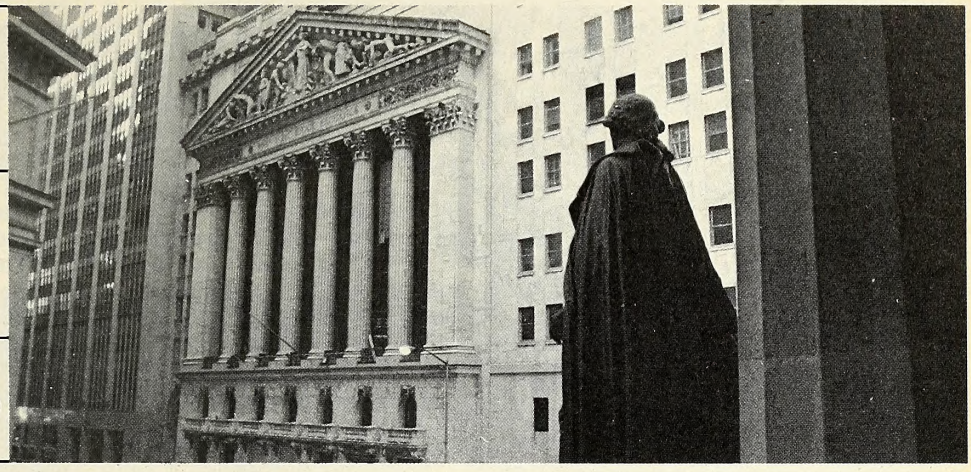
ON-LINE systems

36575 MUDGE RANCH ROAD • COARSEGOLD, CA 93614 • (209) 683-6858

Add $1.00 for Shipping
Visa, MasterCharge, Check, C.O.D.

Buttonwood Apples

BY KEN LANDIS



When it comes to investment, any strategy that's profitable is correct. Investment software packages reflect diverse methods for making money, each of which must have worked for someone sometime to have made its way to commercial software status. The methodology of choosing among strategies then becomes a matter of finding the one that most closely matches your convictions.

Any strategy is not okay for designing software. In evaluating investment software packages, we've concentrated on design and execution. Each package reviewed in this column is put through its paces by several professional investors who are at home with computers—a modern-day Buttonwood association whose efforts are much appreciated.

This month, Buttonwood Apples features the *Dow Jones Market Analyzer,* an outstanding package from RTR Software in conjunction with Dow Jones.

Dow Jones Market Analyzer, by RTR Software, Inc. & Dow Jones & Company, (Box 300, Princeton, NJ 08540; 800-257-5114) $250.

Vendor support: Excellent.

Backup policy: Copyable with protection chip.

Hardware compatibility: 48K RAM Apple II Plus with one disk drive, DOS 3.3; 48K RAM Apple II with a 16K RAM expansion card and one disk drive, DOS 3.3; for use of the auto fetch feature: Hayes micromodem II or an acoustic coupler.

First released in January, this package is a highly versatile technical analysis package. It consists of excellent documentation, one copyable program disk, a Versa computing E-Z port, and a protection chip.

The E-Z port is supplied with the package to make the installation of the protection chip as convenient as possible. The chip is installed via the E-Z port into the game I/O socket on the Apple motherboard. The software is readily copyable with any standard DOS 3.3 copy program; thus, there is no problem with producing back-up copies.

The package is marketed in conjunction with Dow Jones & Company and uses the Dow Jones database for data retrieval. Manual entry of data is possible by obtaining price information from a newspaper or financial trade publication, and entering this information via the Apple keyboard.

When the package is first run, the user must supply the local Dow Jones access numbers, password, number of drives, printer slot, type of modem (auto or manual), and modem slot, entering them via the setup selection of the main menu.

After these housekeeping chores have been taken care of, the user establishes the portfolio list on a data diskette. The package will begin the connection sequence for the Dow Jones database via either Telenet or Tymnet. The user is prompted for which network is desired. (A reminder—Dow Jones is accessible through Telenet only during prime time hours; at other times, Tymnet must be used.)

Once connection has been established by the package, it retrieves twenty-four days' worth of price and volume information for each stock on the portfolio list. Twenty-four days is the minimum amount of data required by the system to perform many of its subsequent calculations and hi-res charting. Data will then be stored on what the documentation describes as an RTR data diskette.

The package uses two types of data disks. One is the RTR data disk, which is used to hold historical price and volume data. Each RTR data disk may hold 104 stocks for up to 128 time units, or 52 stocks for up to 256 time units. The other disk is a temporary work disk that holds current price and volume information until it is stored on the RTR data disk. You implement the storage process with a conversion utility that's accessible through the package's menu.

Each RTR data disk and temporary work disk must contain files with like types of securities; bonds, options and equities may not be on the same data disks.

When you are first establishing a stock list on an RTR data disk, the package will prompt you to supply certain information. The first request is for a three-letter tag. The first letter consists of the character required by the Dow Jones database to identify the type of security data to be retrieved, for example, (,) for stocks, (/) for corporate bonds, (—) for options, and (+) for mutual funds. The other two letters are up to you. You may wish to use this tag to denote a given portfolio or class of securities such as "ht" for high technology issues or "bc" for blue chips.

You are then prompted for the number of characteristics you wish to follow. A one indicates that you are following only the closing prices. A two indicates that the closing prices and volume are to be used. A three indicates that high, low, and closing prices are being used, and a four indicates that the high, low, close, and volume are being used. If you choose to enter the data manually, the package affords you the option of entering data in either decimal form or in eights. Most of the major financial publications, including newspapers, publish stock price information in eighths. This facility eliminates the need for the investor to convert the information before entry.

You are then asked for the number of decimal places to be stored on the data disk. A handy chart in the documentation suggests the number of places required as determined by the price of the stocks you're tracking. The number of decimal places decreases as the price of the stock rises. This is because the emphasis on fractional values decreases as the overall stock price increases.

Integrated into the package is a full set of utilities for editing, listing, updating, and alphabetizing data. The packages also allow you to adjust historical data for a stock split.

When information is received via the Dow Jones database, the package will provide the user with an error table at the completion of the fetch cycle. The error table will show any possible errors in the data file due to transmission problems, unexpected data retrieved from the database (such as a day the stock was not trading), or interruption of the fetch. The er-

ror table will show at what point the interruption occurred. If an interruption did occur, the investor will be required to log-on again to the database to complete the retrieval.

When the data file has reached it's maximum size, the system will automatically begin dropping the oldest data first. If the investor wishes to retain this information, one of the editing functions should be used.

The major value of the *Dow Jones Market Analyzer* lies in its charting capability. The purpose of the package is to make available in graphic form the results of the technical calculations; this makes their interpretation and analysis easier.

One section of the documentation is devoted to the interpretation of the graphic data created by the package. It is an informative and useful section for both the neophtye and the seasoned veteran. The documentation also lists suggested readings in the field of technical analysis. A succinct explanation of technical analysis is provided in the first paragraph of the interpretation section:

> Technical stock market analysis is based on the premise that an indication of future stock prices can be obtained by studying past stock prices and trading volume. Price movement is the result of the imbalance between supply and demand, which itself reflects the expectations of the investing community. Technical analysis, then, is an attempt to recognize this imbalance and thereby improve market timing. To this end, the technical analyst relies on data. He attempts to develop methods for identifying trading patterns reflected in that data. He attempts to develop methods for identifying trading patterns which have, in the past, been associated with price movement. Such methods may be entirely visual or entirely calculational.

The package performs two major classes of charting; individual and comparison. An individual chart is, as the name implies, a chart of one individual security. The hi-res screen is divided into two parts. The top two-thirds of the screen is used for charting prices and the bottom one-third for charting volume data, displaying data, and performing certain types of analysis.

For all charts, the horizontal axis represents time units. A time unit may be a day, week, month, or whatever time interval you choose. The vertical axis of the stock chart is used to represent price. At a given time unit, the corresponding high-low-close data is represented by a vertical bar extending from the low to the high with a tic-mark to the right of the bar signifying the closing price.

When the stock chart first appears on the screen, the volume chart at the bottom represents the relative magnitudes of the volume data collected. The process by which this calculation is performed is explained in the documentation. An investor who wishes to see the actual volume at a given time unit may do so by using one of the available commands.

Below the volume chart is data pertaining to the stock being charted. The average daily volume, the stock symbol of the stock being charted, and the number of time units displayed, are presented in this section of the screen.

To chart this information, all the investor need do is specify the security to be charted (by its ticker symbol) and the number of time units to be displayed. The system will then automatically retrieve the data file and display the chart and the information. Once the program has finished drawing the chart, the user will be in what is described as the "flashing caret mode."

The flashing caret mode is the feature of the package that allows the investor to perform various computations and analyses. The most powerful features are available while in this mode.

Moving Averages. There are three types of moving averages: simple, weighted, and exponential. The investor may specify the number of time units for each moving average. The investor can modify any of these averages in order to construct trading bands (high, low, or close), center the average, or shift the average. The results of the moving average calculations are drawn onto the price chart in one of the five selectable colors.

Straight Line Constructions. The investor can plot a horizontal line at a specified price level, possibly to indicate a threshold or decision point. The investor can also plot a linear least-squares fit of the closing prices displayed on the screen.

The package also allows the user to plot trend lines. The investor will be asked to specify the two points that will establish the trend line. The line may be plotted using highs, lows, closes, or any point that lies within the vertical price boundaries of the chart.

The investor can plot a line parallel to the most recently plotted trend line. This feature may be used to indicate threshold or decision points.

The package allows the investor to construct and plot 1/3 to 1/2 speed resistance lines on the price screen. These lines are typically used to determine the level of secondary and tertiary support when an uptrend line is broken by the security or resistance when a downtrend line is broken.

Change Scale. This feature allows the investor to extend the chart horizontally by adding blank time units into the future. The user can extend trend, least-squares, and parallel lines into the future for projections.

The investor may also change the scaling on the horizontal axis. This allows replotting of the security with different time units.

The vertical scale can also be changed to conform to previous print-outs, so that they may be cut-and-pasted together, or to expand a given price section of the chart.

Indicator Charts. These are line charts of values computed using the historical volume and price data of the particular stock. These values are plotted in place of the volume chart at the bottom of the screen so that the computed and actual price data can be compared.

The six indicators available are:

1. nvi: negative volume indicator—a curve that relates a drop in volume to the corresponding change in closing prices
2. pvi: positive volume indicator—relates a rise in volume with the change in the closing prices
3. cvi: cumulative volume indicator—a commonly used tool to maintain a running total of excess up volume over down volume over time
4. vol: volume line chart—exhibits volume density
5. pvt: price volume trend—adjusts the cumulative volume by adding or subtracting a percentage of each successive day's volume. The percentage is determined by the price change for the day, which may be plus or minus
6. dvi: daily volume indicator—the indicator is adjusted depending on where the closing price is relative to that day's trading range

The investor can also superimpose a moving average on one of these indicator charts or can input formulas into the package with the user jump routine facility. This feature is clearly explained in the documentation.

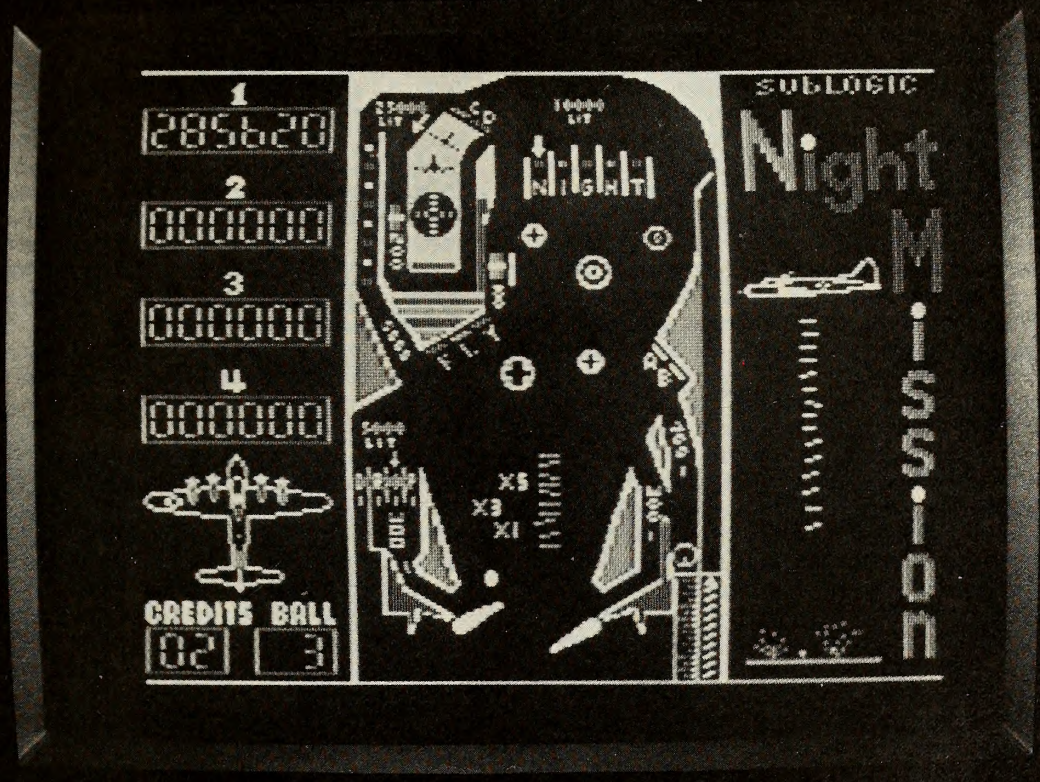
All the formulas that are used by the package are enumerated in a section of the documentation that gives the investor the opportunity to examine not only the charts, but the formulas that generate them.

The *Dow Jones Market Analyzer* is unquestionably the most cost efficient technical analysis package available for the Apple today. Considering its price, performance, and flexibility, it is a technical analysis tool no investor who uses such techniques should be without.

Residents of the Buttonwood tree welcome comments from readers on such issues as strategy, product satisfaction, product review requests, and so on. The input of other investors will be of value to us all. Write to Buttonwood Softalk, 11021 Magnolia Boulevard, North Hollywood, CA 91601. ◻■

Introducing...

A2-PB1
PINBALL



This beautiful and sophisticated simulation offers superior ball action, superior flipper control, and many subtle perfections which will result in your superior enjoyment.

Please don't compare A2-PB1 Pinball with other pinball simulations. Nothing else even comes close. $29.95

The engineering and graphics experts opening a new era in computer simulation.

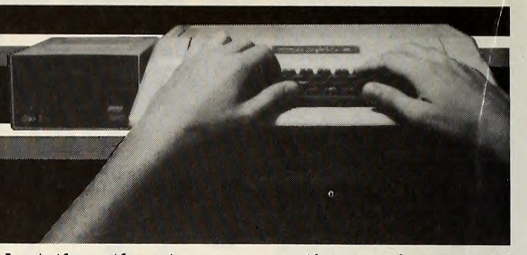
subLOGIC
Communications Corp.
713 Edgebrook Drive
Champaign, IL 61820
(217) 359-8482
Telex: 206995

See your dealer . . .

or for direct orders, add $1.50 and specify UPS or first class mail.
Illinois residents add 5% sales tax.
Visa and MasterCard accepted.

MARKETALK

Reviews



Unless otherwise noted, all products can be assumed to run on the Apple II, Apple II Plus, and Apple III in the emulator mode and to require 48K and one disk drive. The requirement for ROM Applesoft can be met by RAM Applesoft in a language card.

Track Attack. By Chris Jochumson. There was this neat little slot-car setup on the market about fifteen years ago called the Getaway Chase Game that featured a thirties gangster-type black sedan and a police car—much bigger than normal slot cars—tearing around a scale cardboard set of Chicago until they smashed into each other. (There may have been more to it than that; at any rate, it was a heck of a lot of fun.)

Upon this basic premise—surely the best Milton Bradley ever came up with, not counting Water Wiggle—the folks at Broderbund have rung some interesting changes. While the primal pleasure of seeing large black plastic chassis flying in all directions is necessarily absent from *Track Attack,* it does have many charms all its own and appeal for the model railroader as well.

In the setting of a tortuously laid out railroad switching yard, you, the driver of a hopped-up '86 Pierce-Apple, must endeavor to intersect with a heavily laden gold train at various rail crossings, relieving a boxcar of its payload by the cunning expedient of crashing through it. The trick is to avoid crashing into the engine, an empty boxcar, or the yard guard's '52 De Soto that relentlessly patrols the grounds, stealing back your hard-stolen gold and generally making life tough.

The ornate and semiabstract layout of the yard is bewildering, though you soon learn where the best crossings are to make your raids. The controls may drive you to despair, as you must signal a turn somewhat in advance of an intersection or you keep right on going, and you have no brakes and no reverse gear. Once you learn that you must make three ninety-degree turns to go back the way you came, things become easier. But not much. Recommended that you select the I, J, K, M control diamond on the keyboard rather than the A, Z/right and left arrow configuration, which the mind more readily transposes in the heat of the moment.

The faster you can get the gold out of the train and back to your storage area, the more it is worth, as the price declines a dollar a second. Intercepting the guard car after it has retrieved a portion of your ill-gotten gains and before it can redeposit them at the train supply depot rates extra points for guts and spares you a hundred-point penalty. Here you're apt to find that the one time you want to run into this implacable little creep, he's suddenly shy.

At the second level, which you may ascend to any time you feel ready, you leap from boxcar to boxcar (a very nicely animated sequence) until you reach the cab, and take over the train. If you make it—and you may not—you are now the train, picking up gold deposits on the tracks and still avoiding the inevitable '52 De Soto. Here you are provided a free graphic demonstration of the political philosophy which states that radical ideas threaten institutions until they are eventually accepted, thus becoming institutions, which are threatened in turn by radical ideas. You, the former outlaw, join the establishment, your greed unabated; but the guard, the train's former ally, remains your eternal foe, the moral conscience of the game, unchanged by the shift in power and the vicissitudes of commerce. Some compensation for not getting to watch the big plastic cars fly across the living room.

Track Attack, by Chris Jochumson, Broderbund, (1938 Fourth Street, San Rafael, CA 94901; 415-456-6424). $29.95. AC

Tax Beater. By Jack and Carol Lennard. The authors, tax professionals, looked at the other tax preparation packages and saw that the software didn't insulate the user from mistakes, such as overclaiming deductions in areas where the IRS has posted limits.

Their solution was to provide a tax package that asks for minimal data; then adjusts deductions to conform to regulations. Finally, they calculate the tax umpteen ways to determine which method is best for the taxpayer.

Tax Beater is the easiest tax software to use, but the end result lacks the documentation capabilities of competitive packages. If you plan to be audited, this one may not be for you. But for the taxpayer who only wants reliable numbers in the most painless fashion possible—if painless is a word that can be applied to taxpaying—then *Tax Beater* scores high.

An interesting added attraction is the program's ability to assess your Schedule A from the IRS's viewpoint. The program will tell you whether your deductions are high, low, or average for your income. A nifty touch.

First versions sometimes calculated FICA rebates inaccurately, but few got into the hands of the public before DataMost made the correction. ART

Tax Beater, by Jack and Carol Lennard, DataMost (19273 Keyne Street, Northridge, CA 91326; 213-366-7160). $129.95.

Tax Manager. By TASO. Micro Lab's new entry into the income tax preparation market takes a different approach. As you fill in the data, various schedules are created that provide the user with documentation.

Guidelines on the reverse of the two-sided disk are helpful in understanding what the Internal Revenue Service will and won't allow, but the user will still need to refer to various IRS bulletins for caveats on those allowances. For instance, political contributions are deductible, but only to $100; this program won't prevent the user from overclaiming in ignorance.

Tax Manager takes Micro Lab's usual approach to data entry, modularizing the data and saving each module. That can make for lots of disk access time, but the completely documented results make the extra time worthwhile.

An added fillip is a series of questions on the flip side. The Apple translates the user's answers into instructions about which forms will be required. Done before the rest of the program is run, this module provides excellent methodology for proceeding; and methodology is half the battle in doing taxes.

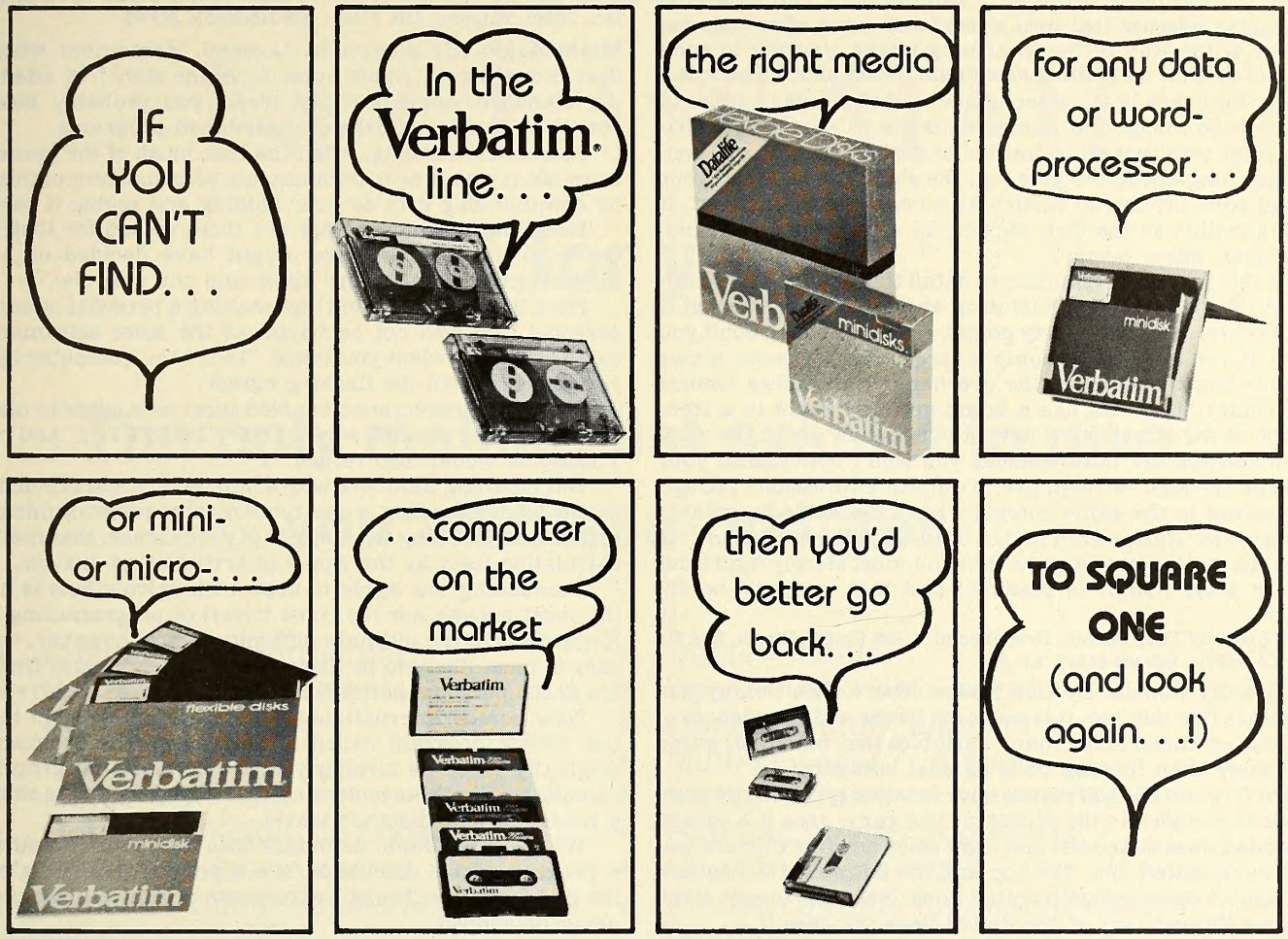
The initial copies of *Tax Manager* had a problem with data-handling in the single disk drive format, but these have been rectified. Users with those first copies can get a free update from their dealer. ART

Tax Manager, by TASO, Micro Lab (2310 Skokie Valley Road, Highland Park, IL 60035; 312-433-7550). $150.

Star Blazer. By Tony Suzuki. Not the latest in cosmic sports jackets, *Star Blazer* is the arcade game that really proves that less is more. No prolonged explosions here; no screaming dives, warning klaxons, or roaring holes in the fabric of space and time. Just you and your "vintage World War III" fighter-bomber, whispering over a sparse, rural, vaguely Indo-Chinese landscape.

To call it a five-level *Pegasus II* would not do it justice, however. It wouldn't describe the bombs that miss the tank, bounce off the cactus, spin end over end, and plop into the sand; nor would it mention the fragment of your fuselage that arcs lazily to earth, trailing pixels of smoke, after you interface with an oil rig. Or maneuvering under the spare fuel tank parachuted down to you by your supply plane, seeing the parachute lines cut by a flying neon tetra, and having to catch the tank in mid-fall. Or playing cat-and-mouse with the sky mines

THE FLOPPY UN-PUZZLE



THERE'S NO PUZZLE . . . when you ask us for a Verbatim catalog and price sheet. Or we can answer most questions by phone.

WE PLAY IT BACK . . . Verbatim.

Every bit of every byte of every track of every disk is pretested. Meets all requirements for IBM, Shugart, ANSI, ECMA, ISO and APPLE.

HEAD CLEANING KITS Removes harmful contamination buildup from read/write heads for 8" and 5 1/4" drives.

FULL LINE . . . ALWAYS IN STOCK

OVERNIGHT DELIVERY* ***If order phoned in before 1 p.m. same day shipment.**

FSI

6901 Canby Avenue, Reseda, CA 91335
(Los Angeles) Phone (213) 705-4202

2680 Bayshore Frontage Road, Suite 1030, Mountain View, CA 94043 (415) 962-9275

LARGEST MASTER DISTRIBUTOR FOR VERBATIM REMOVABLE MAGNETIC STORAGE MEDIA

and the lumbering transports that drop them; a threat in themselves because they are *slower* than you.

The ingenuity required to skrag the supersonic tank on level two is a thing of beauty and a joy forever. (On level four, though, the ingenuity required to zap the tank, *and* avoid the heat-seeking missiles that turn around and come after you, *and* stay out of the way of the swarming tetras, *and* try to snag your fuel drop, is something else again. Good luck.) There is a strategy for every level; a technique; a style. Lots of style. In saturation bombing, the bombs detonate in rhythmical patterns. After your last jet is downed or destroyed, play will continue just long enough to give you the sight and sound of your last fuel pack drifting to earth with no one left to receive it. It impacts grittily in the dirt, kicking up a little spray of sand. Game over. Sheer poetry.

It is this fearsome attention to detail that makes all the difference. If you miss your fuel drop and it lands on a target it will, of course, explode—forty points. (Hi-octane fuel, don't you know.) If you attempt to bomb a target nestled under a tree and your bomb catches in the overhanging branches (minus forty points), it *sounds* like a bomb getting caught in a tree.

None of the effects slow down the action or get in the way, and no corners are cut. Nuances you won't notice until your fifth time through, if then, are given full expression. Though other games in the same category have the same or greater difficulty, *Star Blazer* even makes the frustration a pleasure. If you're not cutting it, you know that you must simply figure out a better way, that it is possible, and that you will be rewarded. AC

Star Blazer, by Tony Suzuki, Broderbund, (1938 Fourth Street, San Rafael. CA 94901; 415-456-6424). $31.95.

Zero Gravity Pinball. By Don Fudge. There aren't many pinball games like this one. It takes three hands and the wisdom of Zeus to keep ahead of the many variables that make this game much more than flipping balls against bumpers.

Zero Gravity Pinball comes with detailed and humorous instructions included in the program. The game area is a square with dotted lines on the left and right sides and five flippers just inside each dotted line. The top and the bottom of the square have sizable openings with dotted lines. Needless to say, if the ball goes through any of the dotted lines you lose it.

Hitting the space bar for the bottom and any other key for the top activates force fields that keep the ball from escaping. The ten flippers keep the ball from escaping on the sides. Each of the flippers and each of the force fields is ineffective unless the ball is close to it.

Sound complicated? It gets worse.

Only one of the ten flippers works at a time. If it's red, it works; if a flipper is green, the ball will go right through it. You make a flipper work by turning the knob of the game paddle until the flipper turns red. (On a black and white monitor, a working flipper has a black dot on it.) But if you turn the knob too much it might give you a bogus flipper that is red but useless. Turning the knob some more either fixes this or results in a superbogus flipper that is red but useless. Turning the knob some more either fixes this or results in a superbogus flipper that is there and not there.

Coping with all this at first is like trying to shave a lion's beard. You get killed so fast you barely know what happened. It is important not to drop the paddle when stabbing frantically for the space bar. At the lower levels the ball is moving relatively slowly and you have some time to react. At the higher levels the ball moves much faster and only the most confident and skilled players can survive.

One last aggravating feature will plague expert and novice alike. The center hole acts like a bumper usually, but every so often it turns into like a black hole and swallows the ball. There isn't a whole lot you can do about it.

Fudge's animation is consistently first-rate and the color graphics spice things up. Like any good pinball game, Fudge's version offers you the challenge of controlling the ball, but it takes longer to master this here than in conventional two or four flipper games.

Emerging as a dark horse in the computer pinball sweepstakes, *Zero Gravity Pinball* is certainly original and for this reason alone deserves to do well. The fact that it is a darn good game will not go unnoticed. DH

Zero Gravity Pinball, by Don Fudge, Avant-Garde Creations (P.O. Box 30160, Eugene, OR 97403; 503-345-3043). $29.95.

MatheMagic. By Joseph R. Luciano. Remember when you first brought your Apple home from the store? In addition to the standard Apple-provided disks, you probably bought a couple of games or disks of contributed programs.

Late in the evening, when the first blush of the games had worn off, it was time to demonstrate your mastery of this thing by commanding it to do your bidding and seeing it respond.

Recalling that computers are most valued for their number-crunching abilities, you might have decided on a very modest project: make the Apple add one plus one.

Here is when you found that making a personal computer a personal tool was not as trivial as the store salesman indicated, because when you typed "1+1," the computer ignored you and returned the flashing cursor.

A tad of perseverance enabled most new users to discover that the proper method was to enter "PRINT 1+1" and return. The Apple would then return "2."

But handling numbers with some facility and efficiency required understanding some fundamental programming techniques and grasping the concept of precedence, that method of calculation used by the Apple to arrive at an answer.

Persuading the Apple to cope with calculations is neither the most arcane nor the most trivial of programming tasks. Regardless of the ultimate difficulty factor, however, the new user is most likely to be disappointed when he first finds that his $2,000 personal computer won't even cope with 1+1.

Now comes International Software Marketing to the rescue with a program called *MatheMagic*. The program was originally billed as turning your Apple into a programmable calculator. That's as understated as billing the space shuttle as a means of long-distance travel.

While *MatheMagic* definitely does turn your computer into a programmable calculator, it's a programmable calculator the likes of which Texas Instruments or Hewlett-Packard has never developed.

It allows you to design formulas and apply labels to the formulas by which to recall them from disk. These labels can then be used within other formulas to call the original formula. The capability of the program is indicated by the fact that the user can nest formulas six deep before the program will sigh in weariness.

The real pleasure of working with *MatheMagic* is that it's overcome the limitations found in so many microcomputer software products. Most micro software will do exactly what the author designed them to do and not a whit more.

To folks who cut their computer teeth on a micro, that seems like a realistic limitation. But the fact is that software developed for bigger systems will often have capabilities beyond what their authors have conceived by virtue of the large amounts of code produced.

The result is that the innovative user can often coax from a program results that surprise the authors—much as authors are presently coaxing from their Apples capabilities Steve Jobs and Steve Wozniak didn't dream of when they were tinkering in that infamous Silicon Gulch garage.

Using the program with two disk drives is a dream, unfortunately the same cannot yet be said for using it with one drive. ISM discovered a serious bug in the one-drive version of 1.5A and are presently modifying the program to make the appropriate fixes.

The company has pledged to replace all 1.5A versions purchased by single-drive users upon receipt of the warranty card. However, the wise one-drive buyer will await version 1.5B before seriously considering the product.

Certain of the author's choices, particularly as they pertain to one-drive use, don't sit well with experienced users. On the other hand, those choices do not dim the achievement of plac-

If you only do things first class
then
1st CLASS MAIL™
is for you.

The only mailing list system to give you all these features.

- Custom label formatting
- Custom screen prompts
- Print envelopes
- Print master list
- Merge mail lists
- Optional 10-digit zip codes
- Up to nine labels across
- Converts Continental's MAILROOM files
- User selected message on any line of label
- Custom punctuation between fields
- Default inputs
- User defined field lengths
- Optional return address
- Multi-key search & sort
- Multiple labels for each entry
- Handles foreign addresses
- Up to nine-line label
- Printer controls for custom label printing
- Prints single or continuous forms
- Allows for expanded print or other special printer capabilities

1st CLASS MAIL from Continental Software is more than just a simple mailing list. With this menu-driven user-friendly system, you can create your own specialized entry and retrieval list to suit your needs. No longer are you stuck with a manufacturers on-screen prompts and field lengths. Do you need a longer last name field and two address lines? 1st CLASS MAIL can handle it. Do you need a seven-line label? 1st CLASS MAIL can do that too. Do you want to print envelopes with return address and an attention line for all? 1st CLASS MAIL will do it for you!

NEW
from
CONTINENTAL SOFTWARE

16724 Hawthorne Blvd., Lawndale, CA 90260 • (213) 371-5612

Introductory Price $74.95

NAME _____
ADDRESS _____
CITY _____ STATE _____ ZIP _____
CARD NUMBER _____ EXP _____
MasterCard ☐ Visa ☐
California Residents add 6% sales tax.

Apple][is a registered trademark of Apple Computer Inc.
Hardware Requirements: Disk drive, DOS 3.3, APPLE][48K, Applesoft in ROM. Printer and second disk drive optional.

GREAT LAKES is proud to announce...

The most versatile RAM boards for your Apple II now offer MORE

MORE WORKSPACE — Add dynamic RAM to your computer with one or more of these boards.

64KC - 64K DYNAMIC MEMORY BOARD by Legend Industries. For only $349 an additional 64K of dynamic RAM using only one peripheral slot and less power than a 16K board. Includes demo disk with Memory Master & manual.

128KDE SOFT DISK — 128K RAM board by Legend Industries. The first 128K RAM card for the Apple II computer. Uses "state of the art" 64K chips. Includes demo disk with Memory Master, Disk Emulator, Legend Mailer and a comprehensive manual. All for $750.

DISK EMULATOR SYSTEM — Two 64KC cards with Disk Emulator software for $700.

MORE SPEED — Choose the disk emulator that you need to significantly increase the speed of the disk I/O.

DISK EMULATOR by Legend Industries. For Applesoft programs using DOS 3.3. $49.95.

PASCAL SOFT DISK EMULATOR by Ferox Microsystems and Legend Industries. Allows the Legend 64KC or 128KDE boards to be used as disk drives in the Apple Pascal System. Supports simulation of up to 5 disk drives. Only $49.95.

CP/M FAST DISK by Evan Tobin for Q.E.D. Systems. Increase the speed of your CP/M application programs by using 64KC or 128KDE boards as your disk drive. A significant speed improvement for only $69.95.

MORE UTILITIES — Additional uses for your Legend 64KC and 128KDE.

MEMORY MASTER by Legend Industries. Relocates the DOS 3.3 Disk Operating System to provide the user with a full 44K of programmable RAM in the Apple motherboard. Manages Firmware cards. Works with most RAM boards. $34.95.

LEGEND MAILER by Legend Industries. Uses the 64KC or 128KDE to provide the large capacity mailing list that you need for only $34.95.

MORE CONFIDENCE — Legend Industries has extended the warranty on the 64KC and 128KDE board to 180 days.

MORE INFORMATION — Dramatic software products using the 64KC and 128KDE are in the final stages of development. Contact Great Lakes Digital Resources for a catalog describing existing and new products.

Go to your local dealer and see these products today!!

For dealer orders or if your local dealer does not carry these products - contact:

GREAT LAKES DIGITAL RESOURCES
P.O. Box 32133
Detroit, MI 48232
(313) 772-6544

VISA/Master Card Accepted
on Individual Orders

ing a versatile tool in the hands of Apple users.

First-hand viewing at a computer store should convince most users of the value of this program to their library. ART *MatheMagic*, by Joseph R. Luciano. International Software Marketing, Suite 421, University Building, 120 East Washington Street, Syracuse, NY 13202 (315) 474-3400. $89.95.

Pascal Utility Package 1. By Al Weiner. The Apple Pascal operating system has a number of advantages over Apple DOS. Many experienced Apple Basic programmers are interested in making the switch to Pascal but are held back by the complete incompatibility of the two operating systems. DOS and Apple Pascal record data in very different formats on disk. This means that all the text files and data files laboriously developed through the use of programs employing DOS commands are inaccessible to Pascal programs. The Basic programmer may be willing to recode programs into Pascal, but to require that all that data (or text) then be re-entered is simply asking too much!

Gryphon Microproducts offers a solution to the problem of moving DOS files to Pascal disk in the first of a series of Pascal utility packages, the *Pascal Utility Package 1*. One of the utilities provided in this package permits Basic users to transfer DOS files to Pascal-formatted disk.

The *Pascal Utility Package* (*PUP1*) offers a number of distinct facilities, of which DOS-to-Apple Pascal file transfer is only the most important. Other features are cataloguing of Pascal and Basic disk, a printout program for Pascal program files, a configuration program that integrates the user's printer and Mountain Computer Clock into the software system, if they are present, and an automatic startup program that sets the system date by reading the calendar clock.

The *PUP1* documentation directs the user to transfer a system startup program and an exec file from PUP1: to the user's boot disk. In order to take advantage of the automatic clock accessing feature and the printer configuration features, the user must execute a one-time configuration option in the *PUP1* program. This program asks the user whether a Mountain Computer clock is installed, whether the system has a printer, and, if so, what the characteristics of the printer are. (One question here is a bit tricky. When asked how many lines the printer prints on a page, be sure to specify the actual number of lines that should be printed on the page—perhaps sixty—rather than the number of lines that could fit on the page—sixty-six for most printers using forms of standard 11-inch length.) Users who do not have the clock card or who have a non-standard printer may choose not to take advantage of the automatic startup features, which add a number of seconds to the normal reboot time. They can do this simply by not transferring the two special startup files to their boot disk. All the other features of the *PUP1* utility can be used simply by *x* (executing the *PUP1* program and choosing the desired utility from the screen menu.

Once the startup files have been transferred and the configuration option has been executed, subsequent boots of the system cause an execution of an automatic soft configuration program. This is a convenient way to boot for the first time every day, since it sets the system date used by disk filing operations. Users with Mountain Computer clock cards will appreciate the automatic access of the clock to find the system date. (Those without the clock are prompted to type in the date.) The automatic startup feature also soft-configures the system for the characteristics of the user's printer. This is particularly convenient for use with printers that provide an automatic line feed upon receipt of an ASCII carriage return symbol. The Apple Pascal system ordinarily adds a line feed to each carriage return sent to the printer itself, but the *PUP1* configuration program can disable this automatic line feed for systems with such printers.

The formatting print utility provided is appropriate for Pascal program source files. Unlike the formatting printer programs discussed in "There's a Powerful Word Processor Hiding in Your Pascal Apple" (*Softalk*, February 1982), it does not provide special features for word processing, such as an underlining capability. It does provide several other useful features, however. It automatically posts date and time in a page header in the formatted printout. This time and date come from the clock if one is installed. Users who do not have the clock can still use the automatic time-and-date posting feature by using the Pascal Editor to create a simple text file called DATE.TEXT on the boot volume. This file will then be used as the source for the time and date data by the printout program. The user also has the option of adding another header line to each page of the printout at the time the utility is used. An unusual feature among Pascal printout programs; the output is echoed to the system console, permitting the user to monitor the printout without walking over to the printer.

The most powerful utility provided in *PUP1* is the DOS-to-Pascal file transfer feature. Text files, binary files, and Applesoft program files can all be transferred directly. When an Applesoft program is transferred, the Basic language tokens are automatically expanded into their normal full text expression in the Pascal volume source file. This utility does not translate programs from Basic to Pascal. Rather, it transfers files from DOS-formatted disk to ordinarily incompatible Pascal-formatted disk.

In addition to programmers, other users are likely to find a wealth of applications for this utility. For example, the owner of a commercial communications package written in Apple Pascal might prefer to create text files using a DOS-based word processing program. These files could later be transferred to a Pascal disk via *PUP1* for eventual telephone transmission using the Pascal communications software.

The documentation consists of a modest 24-page booklet that adequately introduces the features of the *PUP1* utility package. Further documentation, including notes on the improvements provided by the current software update, is given in a number of text files with the .HELP suffix. These can be printed out using the formatted print utility or they can be read from the screen by using a help option in the *PUP1* program.

In the interests of user friendliness, program corrections contained in revised releases of the *Pascal Utility Packages* will be sent free of charge to owners of earlier versions upon receipt of the original disk by Gryphon Microproducts. Release 1.2 corrects defects discovered in the file transfer utility of earlier releases. AM

Pascal Utility Package 1, by Al Weiner, Gryphon Microproducts (Box 6543 Silver Spring, MD 20906; 301-946-2585) $39.95.

Computer Foosball. By Keithen. Foosball is the latest game to be interpreted for the Apple, following in the footsteps of computerized versions of baseball, football, pool, and pinball. In its original form, foosball is a game of fast action and skill.

Wrist and eye coordination made foosball a challenging parlor game. Translated into electronic media, *Computer Foosball* is fast and tough—a terrific two-player game. Get yourself a good set of paddles and an opponent whose feelings you can't hurt.

It's the Gruds against the robots. The screen displays the micro playing field with the two teams occupying their specific places. Each side has eleven players in four columns evenly spaced apart. The center column of each team comprises five players and the goals are defended by three player configurations. Offensive maneuvers are accomplished with a column of three players. Each player on the two teams is capable of kicking the ball forward and backward.

All the players on a side move at the same time up and down when you twist the knob of the game paddle. Pushing the button causes the players to kick. The object of the game is ball control and scoring goals. The electronic soccer ball, if you will, is very well animated, displaying English manners and other pool ball characteristics.

The solitaire version pits the human as Gruds against the computer as robots. There are two speeds, fast and normal; even at normal speed the computer is a pretty tough opponent. Old-time foosball champions beware. The computer will make you look like an ape. Frequently, scoring shots are deflected off the back of your own defending player into the goal.

Pong-like bouncing of the ball off walls and other players is the major skill required to win at *Computer Foosball*. Even so, sometimes the ball moves so fast that it takes a great deal of luck to have things go your way. Many hours of playing will surely result in more skill and less luck, but there is a limit to your ability to react fast enough.

The two-player version will be a hit at rowdy parties. When two evenly matched opponents face off, it can get pretty exciting. The first person to reach five goals, or more in the case of a tie, walks off the victor.

Computer Foosball allows up to four players with paddles, two on each side. Four people swearing is just as invigorating as two people swearing.

Every so often the ball gets stuck in a corner or stuck bouncing between walls; if you do a control-B the ball is served again. After each game the computer plays a short version of "The Stars and Stripes Forever"; the loser has the option of using control-S and not receiving insult on top of injury. DH

Computer Foosball, by Keithen, Sirius Software (10364 Rockingham Drive, Sacramento, CA 95827; 916-366-1195). $29.95.

The Amdek Digital Video Multiplexor (DVM) is a complicated circuit that allows an Apple II to output its text, hi-res and lo-res graphics on an **Amdek Video II** red/green/blue (RGB) display monitor.

Most video monitors that display color draw the pictures you see by aiming three electron guns at the picture tube. The electrons shot from these guns are aimed at small spots on the picture tube—those little dots of color you see when you look at a television picture close-up through a magnifying glass. Each electron gun lights up a specific color in dots; one red, one green, and one blue. You may have toyed with colored lights in grade school and found that when these colors are mixed, you can create all the colors of the rainbow just by varying which ones you mix and by varying the intensity of the lights; you get white by mixing red, green, and blue together in equal intensities.

Your television set or composite video monitor takes one signal, the picture signal (called "composite video"), and decodes the picture into the proper red, green, and blue signals for each of the electron guns. The guns then fire their electrons across to the picture tube, illuminating the small dots of color that, when viewed as a whole, compose the image represented in the composite video signal.

If you haven't guessed yet, an RGB monitor requires more than one input. In the case of the Amdek Color II, five signals are required: a red signal, a blue signal, a green signal, a horizontal synchronization signal, and a vertical synchronization signal. The color signals convey the pictorial information, and the synchronization signals convey information that keeps the picture stable on the screen.

Why build an RGB monitor when its function seems similar to that of a television set? The answer is resolution. Resolution is a quality that describes detail, or clarity, particularly in the context of pictures or images.

Think of resolution this way: when you look at a newspaper photo under a magnifying glass, you can see that the picture is made up of little blobs of ink. The smaller these blobs are, the more detail the picture has, and consequently the greater the resolution. An Apple II has a lo-res graphics mode of 40 spots across by 48 up-and-down and a hi-res graphics mode of 280 across by 192 up-and-down.

In lo-res an Apple II can display sixteen colors, whereas in hi-res it can display only six colors. This trade-off isn't specific to the Apple; many computer graphics circuits lose choice of colors as the resolution of the picture increases.

The Amdek Color II offers much higher resolution than a television set or composite video monitor is capable of; and a special bonus is added by having separate red, green, and blue inputs—color-on-color drawing is made possible! The clarity with which the Amdek Color II draws pictures is quite striking in comparison to the Amdek Color I, and, of course, much clearer than a television set can handle. With the appropriate circuitry, a character or line can be drawn in any color and superimposed over a background of any other color. Unfortunately, the DVM board doesn't have this capability.

A benefit of the DVM circuit is that it can adapt the output of 80-column display boards (the instructions included describe how to adapt a Videx Videoterm-80 to the DVM). You can view 80-column outpout on a Color II, whereas the same picture would be quite poor on a Color I.

In addition to the 80-column capability, you can type in instructions to turn on or off the RGB electron guns separately. In so doing, white pictures (either graphics or text) with the red and blue guns shut off appear green. There are eight basic colors possible with an RGB monitor:

Colors Turned On	Resultant Color
None	Black
Blue	Blue
Green	Green
Blue and Green	Cyan
Red	Red
Red and Blue	Magenta
Red and Green	Yellow
Red and Green and Blue	White

In fact, only these eight color combinations are possible with the Amdek Color II and the DVM board. An inexpensive modification to the Amdek Color II will give it an additional input, "Intensity," which gives the set of basic colors an additional eight colors to choose from.

Nor was there advantageous use made of the color-on-color capabilities of the Color II; a green line drawn over a blue background still has the color fringing "zebra stripe" effect it always had on the Apple II. Also, colors on the RGB monitor appear as vertical bars of color, whereas they appear as solid on a composite video monitor or television set.

Some games—*Space Eggs,* for example—played on the Apple II with the DVM board, turn out well. The graphics are very clear and brightly colorful; space ships and alien critters take on a whole new personality. However, *Bug Attack* and many of the hi-res adventures didn't improve; in fact, you may prefer them on a television. I suspect that game authors are going to stick to the television set-oriented graphics, and not write games that look flattering on an RGB monitor.

All in all, the Amdek DVM board did its job—converting the Apple Text and graphics into an RGB signal for the Color II monitor, but the end result may not be worth the overall price to you. The DVM board is priced at or over $200, and RGB monitors like the Color II are nearly $1000. RK

Amdek, 2420 Oakton Street, Suite E, Arlington Heights, IL 60005; (312) 364-1180.

Hi-Res Secrets. By Don Fudge. *Hi-Res Secrets* is a complete graphics tutorial for the Apple. The package comes with four disks and about two hundred sixty pages of documentation in a looseleaf binder. It covers all hi-res graphcis subjects except three-dimensional animation. Sound and tone routines are also fully explained. A working knowledge of Basic and a simple background in assembly language are assumed.

The main focus of the package is on drawing and animating different types of shapes. Three types of shapes are explained: vector shapes, which are the type in the Applesoft manual; block shapes, which are used for extremely fast animation in arcade-type games; and hplot shapes, which are used for outlines of objects and do not animate quite as well.

Several types of animation are discussed for each type of shape. The main types of animation are simple (both one and two page), shift (ROR and ROL), and TABLE (much faster than most other types of animation). Routines are included to convert one type of shape to another and to create animation sequences.

Animation sequences are used when you animate an object such as a man where the arms and legs move independently of the motion of the man. There are demos and utilities on the disks for each type of shape and animation. LISA-compatible source files are supplied for almost every routine, and many of the machine language routines are explained step-by-step in the documentation.

Several other important concepts are covered in the manual that relate to animation. These include the Applesoft collision counter, the advantages and disadvantages of *draw* and *xdraw,* and the use of screen flipping.

Animation and shape drawing comprise only about half the package. There are also programs and explanations that cover 560-dot resolution, show how to draw circles, ellipses, and other geometric figures, and show how to make all the white lines in a picture look thicker and whiter.

A substantial part of one disk is devoted to various sound routines. There is a program to write music where pressing different keys will play different notes. It also allows the user to store the notes on disk and play them back. A short game is included that demonstrates some of the many sounds that can be produced by noise routines. Of course, source files are included for the sound routines.

The fourth disk deals with the use of color on the entire hires screen. The palette program allows the user to fill the screen with almost any color—there are twenty-one basic colors with many potential mixtures.

The color filter allows you to filter out any color or complement the entire screen. For instance, if the screen is white, filtering out violet would result in a green screen. Complementing the screen would change it to violet. From there, setting the high bit would result in blue. By using the palette and then the filter, some very interesting pictures can be designed.

The documentation for *Hi-Res Secrets* is quite comprehensive. Occasionally, it is confusing when the author refers to "Program 7 of disk 28B," instead of naming the actual program, otherwise, it is excellent. Again, the user is expected to know Basic and some assembly language. If you have been following Roger Wagner's Assembly Lines, then you are well prepared.

If you are interested in graphics programming and dream of writing programs like *Raster Blaster* or *Sneakers,* then *Hi-Res Secrets* is probably a good starting point. If you are just interested in how hi-res works, this tutorial is a great place to learn. PS

Hi-Res Secrets by Don Fudge, Avant-Garde Creations (Box 30160, Eugene, OR; 503-345-3043.) Either DOS. $125.

LET US HELP YOU MAKE MONEY

TAX PREPARER by HowardSoft:
A professional package for year-long record-keeping and year-end tax filing; prepares federal returns for 10 schedules and 11 forms, including income averaging, min and max taxes, depreciation and capital gains, and much more; easy to operate because of self-prompting instructions, automatic posting among forms, and powerful itemizing features; prints in IRS format for direct filing, including printing on preprinted Form 1040; annually updated at a fraction of original cost.

CREATIVE FINANCING by HowardSoft:
A flexible package for evaluating real-life loans & investments; provides cash flow & R-O-I projections as well as payment tables and objective decisions; can be used to determine yield to maturity of discounted bonds, present value of annuities stream, buy vs. lease decisions, APR of loans including wraparounds and analyses of all new loan types.

REAL ESTATE ANALYZER by HowardSoft:
A tool for objectively comparing alternative investments and projecting future results; professional in both analyses of cash flow & R-O-I and client-oriented report printouts; allows what-if studies for changes in property values, loans, rents, taxes, operating expenses and the value of money; perfect for cash and profit projections on new or old rental property.

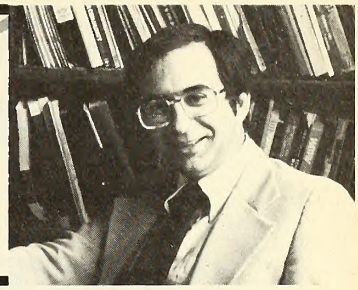
ALL 3 ARE UP-TO-DATE WITH THE LATEST TAX LAWS, INCLUDING ACRS DEPRECIATION AS CREATED BY THE 1981 ECONOMIC RECOVERY ACT

Howard Software Services
for the SERIOUS Personal Computer User

8008 Girard Ave., Suite 310 | La Jolla, CA 92037 | (714) 454-0121

Mind Your Business

BY PETER OLIVIERI



Well, spring is rapidly approaching, and, while this may not have much significance to those of you who've been enjoying relatively continuous warm weather, it certainly means a lot to those of us who've been suffering the winter blues.

Let's begin this month by profiling a business user. Much can be learned from what other people are doing. Not only is this a way to get useful information about the hardware and software other users are pleased or displeased with, it also has the potential for sparking a new idea or application that might prove useful to you in your own business.

Business User Profile. Computers and photography may seem at first like unlikely mates. On further reflection, the relationship between these two areas begins to develop (an awful pun!) more clearly. One business user who has linked computers and photography in a creative way is Greg Adams, a certified professional photographer whose company is the Adams Home of Professional Photography in Morgantown, West Virginia.

When Adams first considered acquiring a computer, he intended to use it to address the bookkeeping function of his business. It soon became apparent, however, that the Apple could do much more than just bookkeeping. It could format and print letters, sort and prepare mailing labels, monitor the checking account, and categorize expenses.

One of the greatest services the Apple supplies to the organization is the ability to provide a quality, personalized letter to every customer who inquires about what services are provided.

The equipment chosen for these tasks includes an Apple II Plus system with 64K, two floppy disk drives, a monitor, and a one thousand word per minute printer. The software currently being used includes Muse's *Supertext II* word processing package and a bookkeeping package from BPI. The total cost of this particular system (hardware and software) was a bit under six thousand dollars.

Adams's company uses the word processing program to send out letters to prospective clients, to send follow-up letters during sales promotions, and to send reminder letters to all overdue accounts.

The firm uses three software packages for handling mailing lists and labels: *ApplePost* (Apple Computer), *Address Book* (Muse), and *Mailing List Database* (Synergistic Software). The last of these packages is the one the company uses most.

With this system, Adams can maintain an updated mailing list at all times and can sort the list by name, address, telephone number, state, zip code, a special code of his choice, or by any combination of these elements. What he now does quickly and easily used to take several weeks. New promotions have also become easier to initiate. He keeps a list of anniversaries and birthdays and, shortly before these special dates, sends out promotional letters to suggest portrait as a nice gift or personal remembrance.

The BPI bookkeeping system has allowed the firm to keep much better track of what bills are outstanding and for how long. Sales are broken out into categories at the end of each month and summarized at the end of the year. All in all, much valuable information for planning and decision making is now available. Adams is quite pleased with his investment in the Apple and recommends it highly to the small business owner.

A Plan for Using a Computer in a Small Business. As mentioned last month, we'll be presenting a guidebook, during the next several columns, that covers planning for and using a computer in your business. Without trying to be "all things to all people," we'll attempt to present guidelines for those already in business (but just starting), those who are considering using their Apples in their businesses, and those who might like to start a business using their Apples.

We anticipate having brief sections dealing with such topics as where the computer can be used effectively, what kinds of new businesses you might start with your Apple, how to determine the computer needs of your business, how to choose hardware, how to choose software, and planning for growth. In addition, we anticipate that some miscellaneous topics may be of interest. It's important, for example, to become familiar with service options, strategies for choosing diskettes, cleaning and maintaining equipment, the options available with hard disks, the impact of the video disk, the potential for networks, graphics applications, and adding a communications option to your Apple.

Since a previous column dealt with the most common business applications, it should be sufficient merely to summarize them here. Typical first applications are mailing list maintenance, accounts payable, accounts receivable, and word processing. Soon thereafter, inventory control, order entry, and perhaps payroll applications are instituted.

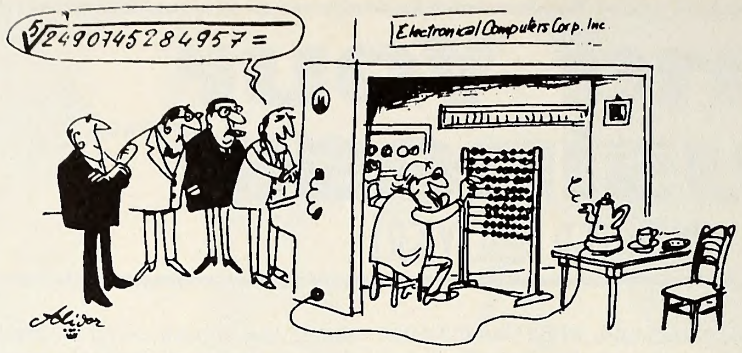
Eventually, a database system is acquired to provide management with instant access to important business information. This is often followed by financial modeling and forecasting applications, intensified use of graphics (for output reports), and perhaps the addition of a hard disk.

Starting Your Own Business. Many people who have personal computers in their homes are looking for ways in which to use the computer in a new side business. This is precisely how many of the advertisers of products in *Softalk* got started. It is also how the Apple itself came to fruition. So, be wary! That sideline business of yours may blossom into something quite substantial. In today's economy, anything that has the potential of adding to one's income should be explored thoroughly. This is not meant to imply that everyone who starts up an inhouse business will be successful, but such a venture certainly deserves some attention.

The first question is: What kind of a business? This is really limited only by one's creativity and ingenuity. To be successful, you need a good idea, some aggressiveness, some talent, and a lot of persistence.

There are generally two directions you can take. The first is to offer a service to local organizations. It's likely that your service would provide some of the more traditional applications. For example, you might prepare and maintain mailing lists for local clubs or organizations or teach about microcomputers in your town's continuing education program. You might use your word processing system to type term papers for students, prepare reports for local government officials, or type promotional material for area businesses. Or how about maintaining statistical summaries for the local bowling league? Computer dating services that use microcomputers to match clients are springing up in some areas. Perhaps the cable television company in your area might be interested in having analyses done concerning viewer preferences.

The second option is to create something new. This can be very exciting and also very profitable. Here, you can either do



Caceta Illustrada/Madrid
Reprinted From World Press Review/March 1982

your own original programming if you have the talent or, more likely, take an existing package and develop a specialized application for it.

You might, for instance, take one of the database packages and design a system that enables a police department to keep track of the burglaries in a town (location, time of day, day of week, amount taken, value of stolen property, property list, method of entry, and so on). You could then offer such a system for sale. Be sure to select an application that has the potential for a fairly significant volume of sales. In the example given, if the system were well designed and accepted, every police department would be a prospective client.

If you do not wish to get involved in the development of a total system, you can develop a specific application for use with an existing package and market it to owners of that package. Obvious examples of this trend are the many *VisiCalc* application packages. Perhaps you can think of a new one. How about *VisiCalc* applications for attorneys? Could you develop a database system for a local politician's office? Lots of areas are as yet untouched. Be aggressive! Try out your ideas. There are plenty of great ideas that never escape from people's heads. Let yours out.

Product Review. Software Technology for Computers is offering database users three versions of its *IFO Database Package.* Version 1 is a low cost package ($120) that will keep records of inventories, patients, clients and so on. It uses a single disk drive. Version 2 (at $200) requires two drives and has a more flexible report formating capability. In addition, version 2 supports some eighty-column boards, lower case, and the Epson printer. Version 3 has been designed for the Corvus hard disk drive. Therefore, both storage capacity and retrieval speed are significantly improved. It is offered at $450.

The hardware required to run *IFO* is, of course, dependent upon which version you are using. It's safe to say, however, that you need at least the following: a 48K Apple II Plus (or Apple II with Applesoft on ROM), DOS 3.3, a printer (40, 80, or 132-column serial or parallel printer), and the appropriate disk drives.

As for the characteristics of the *IFO* series, the table summarizes some of the features included.

Feature	Version 1	Version 2	Version 3
Characters/field	25	25	25
Fields/record	20	32	32
Records/file	1000	1600	12000
Sorting	Yes	Yes	Yes
Multi level search	3-level	5-level	5-level
Summary reports	No	Yes	Yes
Averages	No	Yes	Yes
Calculator made	Yes	Yes	Yes
Totals	Yes	Yes	Yes
Subtotals	Yes	Yes	Yes
Back-up disk	Free	Free	N/A

These are only a few of the features that are part of *IFO* but they should help you to determine if you want more information about these packages.

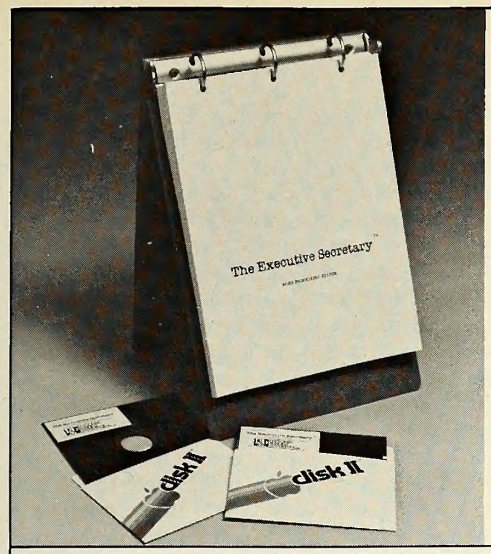
One of the nice aspects of *IFO* is its upward mobility. You can start with Version 1 and, as your needs and talents grow, move upward quite easily to the more advanced versions. Another unusual characteristic of this database package is its ease of use. You can get started with *IFO* in a few minutes. Ease of use has been a prime focus of Software Technology for Computers in the development of their software. The user guide we looked at (for version 1) is clearly written and takes the reader step-by-step through the design and use of a database.

Actual screen examples, sample input forms, and output reports are absent from this manual (as they are from many). But if you are sitting at your Apple while going through the manual, the use of the system is perfectly clear. If you are considering the acquisition of a database package, *IFO* should be one of your candidates.

Software Technology for Computers can be reached at Box 428, Belmont, MA, or (617) 923-4334.

PFS Database Users. Software Publishing Corporation has recently come out with additional supporting materials for its *PFS* database series. All *PFS* users are being sent a copy of the *PFS Forms Sampler.*

The concept underlying the sampler is simple. The booklet contains a collection of forms designed and submitted by a cross section of *PFS* software owners. Included are screen



We'd like your apple® to meet The Executive Secretary™

The professional word processing system for the Apple™ computer

check this chart:

GENERAL Editing, printing, form letters, mail-merge, and electronic mail all in one package at one price?	✓	
User's manual designed for the user?	✓	
User-controlled configuration of printers, slots, drives, and 40/80 column editing?	✓	
THE MANUAL Complete index included?	✓	
Organized as a set of lessons?	✓	
Easel-bound for ease of use?	✓	
THE DOCUMENT PRINTER Integrates files from DB MASTER's Utility Pack™, The Data Factory™, Visicalc™, Information Master™, and most DIF™ files.	✓	
Accepts keyboard input at print time?	✓	
Supports all major printers, including Centronics 737 and IBM ET- series?	✓	
Has IF and related commands to allow conditional printing of information based on the contents of a database or on keyboard input?	✓	
Prints page headers of arbitrary complexity?	✓	
Prints page numbers wherever you want them?	✓	
Automatically generates alphabetical index for words you specify?	✓	
Supports file chaining and file nesting?	✓	
Has multi-level outline indenting?	✓	
Has left- and right-justified tab stops?	✓	
Gives full control of all margins, dynamic text reformatting, centering, and justification?	✓	
Supports Thunderclock™ and the CCS clockcard for automatic dating?	✓	

INTEGRATED CARD FILE FEATURES Allows multiple card files per disk?	✓	
Allows user to define size and content of 'cards' in each file?	✓	
Generates new subset card files based on search or sort criteria for an existing file?	✓	
Incorporates one/multiple line report printer for card files?	✓	
Allows totals and subtotals during report printing?	✓	
THE DOCUMENT EDITOR Keeps up with professional typing speeds?	✓	
User-defined phrase abbreviations?	✓	
40 or 80 character edit modes user-selectable?	✓	
Supports Smarterm™, Superterm™, Videoterm™ and Full View 80™.	✓	
Uses real shift key?	✓	
Supports file merge and unmerge?	✓	
Global search and replace?	✓	
Block operations: move, transfer, delete?	✓	
Character/word/line: insert/replace/delete?	✓	
Allows embedded commands to control special printer functions?	✓	
ELECTRONIC MAIL SYSTEM Menu driven?	✓	
Multiple document queuing?	✓	
Fully automatic with Hayes modem?	✓	

Published by:

Available at computer stores nationwide. $250 suggested retail.

SOF/SYS, INC.

4306 Upton Avenue South
Minneapolis, Minnesota 55410
Phone (612) 929-7104

Now available: The Executive Speller™. Allows on-the-fly corrections and specialized dictionaries. 25,000 word capacity. Suggested retail $75.00.

samples of forms in the areas of education, media, medical, sales, science, small business, and contracting. There are more than thirty forms in the booklet. In addition, the company will soon be releasing a companion booklet, *PFS Report Sampler*, that will contain actual samples of reports that have been designed using *PFS*.

This idea has real merit. Users need as much support and assistance as they can possibly get. Few users know the proper way to design a data input form or even a final output report. In addition, some of the packages available are not as clear as one might expect when it comes to explaining how to input data and get results printed out, making things all the more difficult for the poor user. If you own *PFS* or are considering purchasing it, make sure you inquire about getting these new publications. Perhaps other software developers will consider making similar samples available to their present and prospective customers.

The Readers Speak. "I find myself taking my Apple back and forth between my home and my office. Can this damage the machine at all? Are there any suggestions you might have that can make this process easier? My setup (an Apple with two drives) is portable enough but it's a pain in the neck carrying it back and forth." A. G. Atlanta, Georgia.

Well, you could simply move into your office—or move your office into your home. As another solution, perhaps you could buy another Apple. What, neither of these is viable? Then let's consider your Apple.

In the long-run, moving your Apple around is simply not good for your Apple. After all, a computer is a piece of electronic equipment with a lot of very sensitive pieces. Pulling boards out, disconnecting wires, and so on is hazardous to your machine's health.

Furthermore, *never, never* just pack your Apple, disks piled on top, into the back seat of your car—or into the trunk—to transport it. To the praise of the designers, most Apples survive transitions from place to place. But moving it is risky. Incurred problems might be as trivial as some dirt entering your disk drive or as serious as your Apple entering your floor as you trip or lose your grip.

If you *must* move your equipment around, purchase a carrying case. Several firms advertise cases in computer magazines and most have booths at computer shows. Carrying case prices range from $99 to a few hundred dollars depending upon capacity, quality, and detail. Cases are built for any combination of Apple, disk drive or drivers, monitor, and printer. Some of the cases are lightweight for local travel; others can survive the rigors of long-distance air travel. Pick the one to suit your needs. It's a small investment to make to protect a bigger one—and it's a lot easier to carry, too.

A Reader's Question for You. One of our readers needs information about phone communications, especially those using the Hayes modems.

This reader has hired a tyro programmer who has been doing remarkably well in developing programs the company needs. But communications routines are causing some problems. There seems to be a timing problem; there are so many delays in the program that data is transmitted at roughly 110 baud instead of 300 baud. There also are no error-checking routines to check for lost data.

This user is not necessarily looking for a complete program or solution—a reference book or article would be enough. Does anyone have a suggestion?

Some readers have written in with enthusiasm about forming a business user group. As a beginning, Mind Your Business will mind the business of maintaining a database of all who express interest in this plan to facilitate more formal organization later on. In the meantime, let's use this column as a means of sharing experiences, questions, and needs with one another.

Database Summary. Summaries of particular topics that have been discussed over the course of a year or so can be quite useful. To this end, we are putting together a summary of all the features of databases. Much has changed since we did this last.

We'll be asking vendors of the various programs to complete a fact sheet on their databases (characters per field, records per file, and so on), and we'll add ratings where applicable. We'd especially like to include users' ratings.

If you use a database package currently or have had experience with one in the past, please send your comments along. Include the name of the package, the version number, and your date of purchase (approximate will do). Have fun rating the programs for ease of use, documentation, vendor service and support, and product strengths and weaknesses. We have a lot of valuable data to share with one another.

Let's aim for a summer issue. Get moving!

New Questions and Closing Comments. If you have not yet completed the questionnaire that was included in the March issue of *Softalk* (remember, "Olivieri's Inquisition"?), please do so now and return it.

Inquisitors are traditionally relentless, and Olivieri is no exception. Here are some additional questions for you:

Do you use a mailing list program in your business? If so, which one? What are its strengths and weaknesses?

What kind of printer do you use? What do you like or dislike about it?

Got it? Then here's a summary of what you can do to make Minding Your Business work for you. Send: (1) your Olivieri's Inquisition; (2) help for our business user; (3) what mailing list package you are using; (4) what printer you own; (5) your rating of the databases you use or have used; and (6) anything else that strikes your fancy.

So long, and have a great month. ■

Apple Computer Special Delivery Software, 10260 Bandley Drive, Cupertino, CA 95014; (800) 538-8400; in California, (800) 672-1424. Muse Software, 347 Charles Street, Baltimore, MD 21201; (301) 659-7212. EPI Systems 1600 West 38th, Suite 444, Austin, TX 78731; (512) 454-2801. Synergistic Software, 5221 120th Avenue, S.E., Bellevue, WA 98006 (206) 226-3216. Software Publishing Corporation, 1901 Landings Drive, Mountain View, CA 94043; (415) 962-8910.

Screen Director

Screen Director™ retrieves and displays any standard image file on your Apple's video monitor, large screen TV or color printer. For only $150, Screen Director™ gives you a filmmaker's finesse in boardroom presentations, management briefings and demonstrations for clients.

With Screen Director™ you can scroll forwards and backwards through any color displays, with dissolves or cuts from one image to the next — all at the touch of two buttons. You can even enhance your **Apple Business Graphics** with title slides in a variety of fonts and colors.

Spice your Apple with cinema.

Screen Director™ includes both the two-button controller and all necessary software.

Screen Director™ is a product of Business & Professional Software, Inc., creators of **Apple Business Graphics.**

Apple is a trademark of Apple Computer, Inc.

For information on dealers of Screen Director™ in your area, please return this coupon to

Business & Professional Software, Inc.
143 Binney Street
Cambridge, MA 02142

or call (617) 491-3377.

Yes, please send me a dealer list for Screen Director™ !

Name_____ Title_____

Company_____ Address_____

City_____ St_____ Zip_____ Phone (___)_____

☐ I would like additional information on **Apple Business Graphics** and its use

☐ with this printer or plotter: _____

☐ with these other programs: _____

☐ to produce xerox color copies, overhead transparencies and 35mm slides

☐ I am a dealer.

Softalk 4/82


APPLE
FEST

The Second Annual Applefest Plays in May

BY TOMMY GEAR

Who cannot remember the thrill of the day you purchased your first Apple? Or the rush of imagination that accompanies slipping a new piece of software into the disk drive, and discovering the ease with which it allows you to mine its potential? Or even the ecstasy of reaching 200,000 points at *Crossfire*? (In my case 50,000—ho hum.)

If any of this elicits even a faint "blip" in your memory, come May you will assuredly be ripe for Applefest '82, the world's largest Apple-specific computer show for users and producers, a veritable one-stop shopping spree for everything Apple.

It really began last year with Applefest '81, the brainchild of Jonathan Rotenberg and the Apple/Boston user group of the Boston Computer Society, which broke all records as the best organized, most successful show of its kind. It was the first national computer show devoted exclusively to Apple computers.

Leading Off with Jobs. Expanded and redesigned, this year's Applefest is again produced by the Boston Computer Society. It's also the first event of its kind to receive official endorsement from Apple Computer Inc., the guys that started it all. In fact, keynote speaker for Applefest '82 will be none other than the chairman of the board himself—Steve Jobs.

Applefest '81 proved the practical value of presenting the state of the art in the world of Apple with software, peripherals, and accesories geared specifically to the Apple user. Just being there was enjoyable, but beyond the general atmosphere, which was open and friendly, the great diversity of people attending and their high degree of enthusiasm for the variety of offerings placed Applefest apart from previous computer shows.

Disappointed by other Apple shows he had attended, John Williams (On-Line Systems) called Applefest "the best computer show I've ever been to in my life." Roger Tuttleman (Sensible Software) appreciated the chance Applefest afforded to meet the "names" behind the many companies.

Art Malin (Santa Clara Systems), Tim Hartley (Hartley Courseware), and Steve Boker (Data Transforms) all agreed on the great response their respective products received at Applefest, which was reflected in the business it generated for them throughout the year.

A Special Excitement. But the more than two million dollars' worth of sales that it generated is only one obvious gauge of its success; more telling perhaps is the fact that Applefest left everyone who attended with a feeling of excitement that they had been a part of something special, a feeling that is carried over into the mounting anticipation of Applefest '82.

Five times larger than last year, with the number of exhibitors tripled, Applefest '82 can expect an attendance of twenty thousand to thirty thousand Apple users and prospective buyers. To accommodate all this under one roof, Applefest will be using New England's largest (150,000 square foot) and most modern convention facility, Hynes Auditorium, which is part of the Prudential Center Complex and adjacent to the Sheraton Boston Hotel.

Easily accessible by public transportation or by car via primary highways, the auditorium is surrounded by Boston's best restaurants, clubs, theaters, and stores. This year, Applefest will be

fully air-conditioned, and those attending can enjoy a restaurant facility on the premises with the exhibits. Ample parking is conveniently provided at the auditorium site for a nominal fee. The building is designed without stairways, in consideration for the handicapped, who can avail themselves of elevators and specially reserved parking nearby. Widened aisles in the exhibit area promise to make getting around inside easier too.

Lots To See and Learn. In addition to almost four hundred exhibitors providing a tremendous array of things Apple-related, numerous education programs, much expanded from last year, will run continuously throughout the three-day event. Here's a brief preview of some of the seminars to be presented at Applefest '82: There'll be return engagements by Daniel Bricklin (Software Arts Inc.) speaking on "After *VisiCalc:* The Future of Personal Computer Software," and Leonard Freed (Microcom) speaking on the Apple and telecommunications. Also speaking will be *Softalk* columnist and author of the just-published *Assembly Lines: The Book,* Roger Wagner (Southwestern Data Systems), on how to make the Apple do amazing things with machine language; Greg Tibbetts (Microsoft), another *Softalk* columnist, on the Apple and CP/M; Mark Schwartz (Cavri Systems) on interactive video; Mark Pelczarski (Penguin Software), whose graphics column begins in *Softalk* next month, on creating color graphics with the Apple; and Jock McCleen (Terrapin) on Apples, Turtles, and LOGO. Other seminars will encompass the topics of how to buy business software, how to use the Apple as an aid to making investments, and how entrepreneurs can get involved in the Apple world.

A new feature this year will be the special software spotlights. Happening every hour throughout the show (with repeats so you don't miss anything), in-depth demonstrations will be presented using large video screens to spotlight a wide variety of different software packages. Hundreds of products can be viewed in this way, covering the areas of business, education, utilities, and games. The authors of the software will be present for you to talk with and question.

A Hundred Apples To Play With. The popular hands-on rooms return this year, expanded to include one hundred Apple II and Apple III computers for you to "test drive." For Apple owners and prospective owners alike, the hands-on rooms offer a marvelous opportunity to try out the Apple or learn more about its many applications. An informative multimedia show developed by Apple Computer will also be running continuously for the duration of the festival.

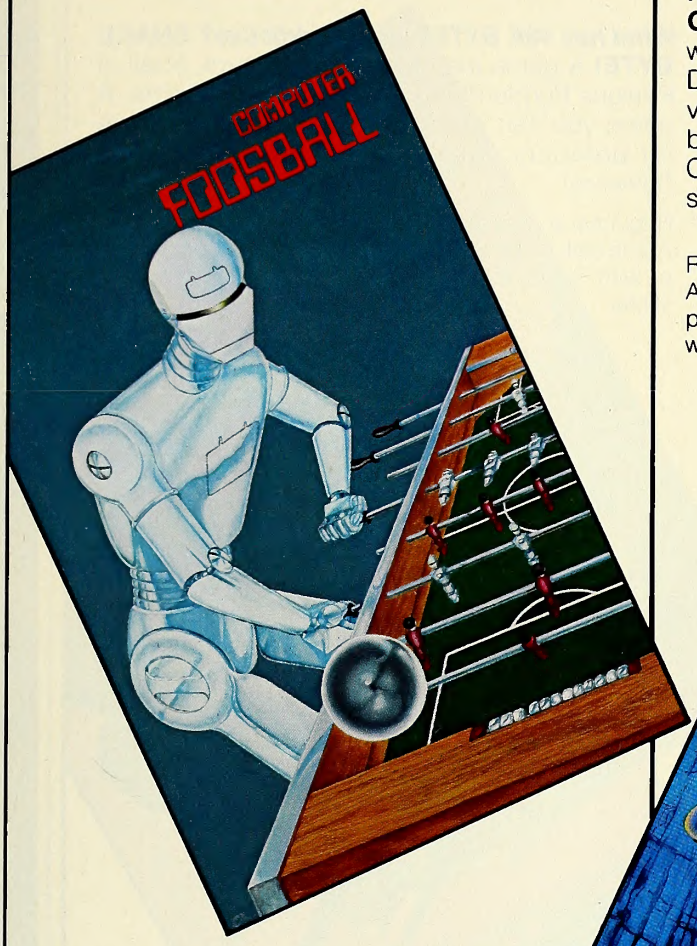
Let all who attend Applefest/Boston '82 be forewarned: wear comfortable shoes and bring lots of money—there will be much to see and much to be purchased. Tickets, which include everything at a single price, are $6 per day, $10 for two days, or $15 for a three-day ticket. Days and hours of the festival at Hynes Auditorium are May 14, 15, and 16, 11 a.m. to 6 p.m.

Through an extensive promotional campaign using direct mail and national advertising, many people will be drawn to Boston for the festival, and accommodations are even now becoming difficult to obtain. For assistance in arranging accommodations and to purchase tickets, those who wish to attend may call or write National Computer Shows, 824 Boylston Street, Chestnut Hill, MA 02167, or telephone (617) 739-2000.

For those who cannot go to Boston to attend, there is good news. Throughout 1982, Applefest will be touring to the larger regional cities so, if you live near Chicago, Houston, or San Francisco, you will still have the opportunity to partake of the experience of Applefest.

All Apple, pure and simple, not diluted by the wares of other microcomputer manufacturers—Applefest '82, promising to be an impressive motivation for prospective buyers who have not yet taken a byte and the one show taken very seriously by all Apple users worth their sap. ◻■

The Action Continues ...



COMPUTER FOOSBALL

NOW YOU CAN PLAY FOOSBALL IN YOUR PAJAMAS! No more dressing up and braving the elements to get your kicks in a smoky, noisy arcade. Now, you and up to three of your friends can play Foosball at home. Divide into Grud and Robot teams and bash the ball around the screen. Have a Foosball slumber party. Save money on gas. Save money on drinks. Save money on clothes. Computer Foosball. Sirius Software cares about your comfort.

Requires an Apple II or II+ Computer with 48K and Apple disk drive. Computer Foosball is playable by one or two players with Apple paddles and by four players with Sirius Software's Joyport™ and 2 sets of Apple paddles.

JELLYFISH

KILLER JELLYFISH ATTACKS NUCLEAR RECOVERY SUB! While trying to remove nuclear waste capsules from the ocean floor, the U.S.S. Dogstar was attacked by giant jellyfish and vicious octopuses. "When we shot them, they just broke up into smaller pieces," stated the ship's Captain. Another sub, possibly Soviet, was sighted during the under-sea mission.

— Navy Intelligence

Requires an Apple II or II+ Computer with 48K and Apple disk drive. Jellyfish is a one or two-player game, playable with keyboard, Apple paddles and joystick, or with Sirius Software's Joyport™ and Atari-type joysticks.

Sirius Software, Inc.
10364 Rockingham Drive, Sacramento, CA 95827
(916) 366-1195

Contact your local computer dealer for more information. Dealer inquiries invited.

The Adventure Continues . . .

TWERPS

NOW! From the company that brought you Sneakers, Beer-run and a host of other blockbusters: TWERPS! Sirius does it again. Another whimsical, challenging game with a cast of top-starring characters. Meet Captain Twerp! Thrill at the shooting Orbiters! Be amazed by the swooping Glingas! Gasp in terror at the drooling Gleepnoks! A game you'll want to tell *all* your friends about.

Requires an Apple II or II+ Computer with 48K and Apple disk drive. Twerps is playable with keyboard, Apple paddles and joystick, or with Sirius Software's Joyport and Atari-type joysticks.

SNAKE BYTE

***What has 48K BYTES and is addictive?* SNAKE BYTE!** A game that works like a charm. A tail of Perilous Purple Plums that's ahead of its time. A game you can sink your teeth into. An antidote for boredom. Snake Byte. Fangs alot, Sirius Software!

Requires an Apple II or II+ Computer with 48K and Apple disk drive. Snake Byte is playable with keyboard or with Sirius Software's Joyport and Atari-type joysticks.

Sirius Software, Inc.
10364 Rockingham Drive Sacramento, CA 95827
(916) 366-1195

Contact your local computer dealer for more information. Dealer inquiries invited.

. . . Sirius Software

BORG

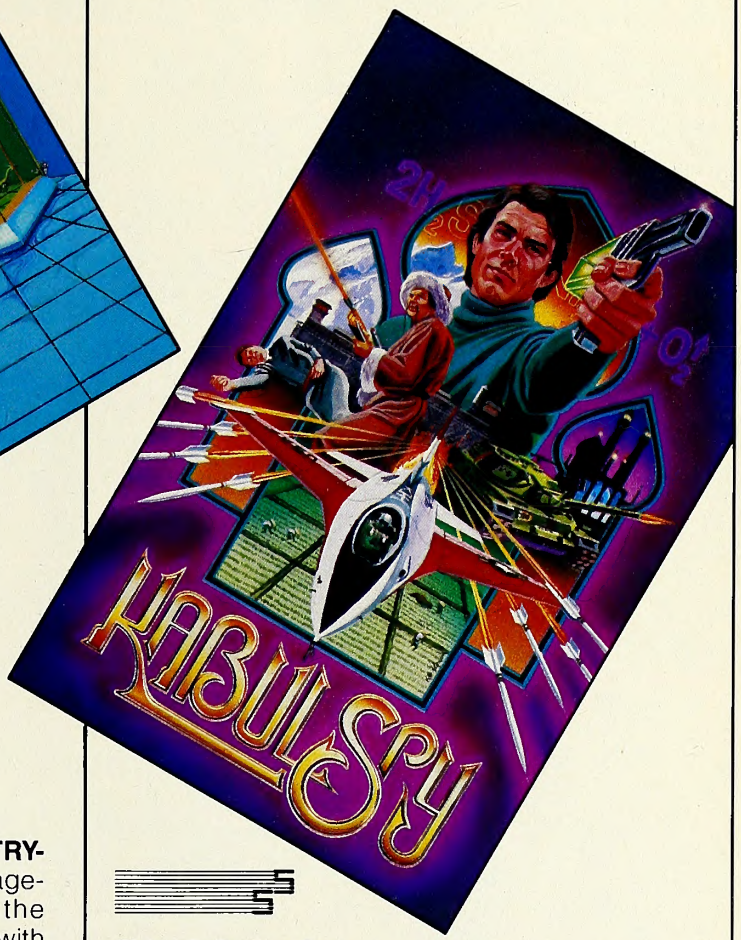
DERANGED GRUD TERRORIZES COUNTRY-SIDE! Protected by Borg, the invincible Drage-roo, and a notorious band of dragons, the infamous Grud has surrounded his hide-out with electrified mazes. Can no one crack the code and rid us of this menace? —Sirius Press Service, Grundonia

Requires an Apple II or II+ Computer with 48K and Apple disk drive. Borg is playable with keyboard, Apple paddles and joystick, or with Sirius Software's Joyport and Atari-type joysticks.

KABUL SPY

TO: KABUL SPY **FROM:** S.S. HQ **MISSION: FIND PROFESSOR PAUL EISENSTADT.** Cap-tured in May of 1981, he is being held by the KGB somewhere in North Afghanistan. You will be given: a money belt, 300 Rubles, a small pistol, a sharp knife and . . . a cyanide pill. We will help you if we can. Caution advised.

Requires and Apple II or II+ Computer with 48K, AppleSoft and Apple disk drive. Kabul Spy is playable with keyboard.



Sirius Software, Inc.
10364 Rockingham Drive Sacramento, CA 95827
(916) 366-1195

Contact your local computer dealer for more information. Dealer inquiries invited.


PRINT
PRINT "PARTSFILE" CREATE MODIFY PRO
PRINT PRINT "TYP
PRINT "1 TO CREATE A PART
PRINT "2 TO USE OR EDIT
PRINT "3 TO QUIT"

The Changing Character(s) of the Apple III

BY JOHN JEPPSON

Tired of the same old crowd? Want to meet some new characters? Here's how to make your own. Give your Apple III programs a new look by typing with custom-made characters you create yourself.

Operating a computer means typing. We type, the machine "types," and neat little characters appear on the screen. But the shape of each individual letter or number or punctuation mark is not frozen in metal as it is on a typewriter. The character shapes are stored as information in a special part of the computer's memory, the character set. If that information can be changed, and in the Apple III it can, then the computer can put on a whole new face.

The Key To Horsing Around. In most small computers the character set information is fixed. It is usually stored in permanent read-only (ROM) memory and cannot be changed. But one of the fundamental features of Apple III is a commitment to keeping everything in software. Apple III contains only 4K bytes of ROM memory, which is used for booting and for internal diagnostics. The entire operating system is on disk and is booted into RAM memory. The advantage, obviously, is that the system can be upgraded periodically by simply booting a revised operating system from another disk. Changing character sets on Apple III is a similar process. You simply change the information stored on disk or in RAM memory or both and, presto, a new character set is up and running. The trick is knowing which information to change.

Please note that we are not discussing drawing letter-shaped figures on a graphics screen. That is possible, usually with great difficulty, on many computers. On the Apple III you may modify the characters used on the ordinary text screen to achieve a wide variety of special effects. Using line segment characters, you can create neat forms and outlines for menus and tables. Or, by rapidly reprinting a series of *picture fragment* characters, each one slightly different, you can produce such remarkable animation as is in the horse sequence on the demonstration disk.

Apple III actually comes with four complete sets of characters, called *fonts*. They are named *Standard, Roman, Apple* and *Byte*. From Business Basic one can easily switch to any of these fonts by using the *Download* program and the *Download.inv* invokable module. Or, from Pascal, you can use the *Unitstatus* procedure. Also, the System Configuration Program has facilities for transferring any of these sets into the *SOS.DRIVER* file of a boot disk. Then, when that disk is booted, the specified font will be the system character set in which text is printed.

A complete character set consists of 128 separate characters that correspond to ASCII codes 0 to 127. Normally the Apple III console will *print* only characters 32 to 127. The first thirty-two ASCII codes (0 through 31) are control codes. Instead of generating a printing character, they signal the console to perform special functions such as moving the cursor. So it would seem that the first thirty-two character positions are useless, since the console won't print them.

The Open-Apple Policy. Apple III, however, has a special shift key, the Open-Apple key, that increases by 128 the value of the ASCII code produced by any other key. Holding Open-Apple while typing the key combination for any code in the range 0 through 127 actually transmits a corresponding shifted code in the range 128 through 255. Open-Apple sets the eighth or *high bit* (number seven of 0 through 7) of the character byte which, in binary, is the same as adding 128. When the console receives a code in the high range (128 through 255) its actions are a little complex, but, in essence, the console prints high-bit characters just as if they were not shifted, except that the thirty-two codes in the range 128 through 159, instead of executing control functions, actually print the characters in positions 0 through 31 of the character set. Thus one can see all the characters by printing codes 32 through 127 (regular characters) and codes 128 through 159 (characters 0 through 31). It's as though the first thirty-two characters were wrapped around and placed at the end of the set.

The following program uses Business Basic's powerful *print using* and *image* commands to display all the characters in a neat array. The first thirty-two characters are placed at the end as if they were numbered 128 through 159. Assuming you are presently using the Standard font you will see that each of these first thirty-two characters is a cute two-letter representation of the traditional control-code name.

```
      PROGRAM SHOW.SET
10    PRINT
20    FOR i=32 TO 47
30      FOR j=0 TO 7
40        number=i+16 * j
50        PRINT USING 100;number,CHR$(number);
60        NEXT j,i
100   IMAGE 3#,2R,5x
```

When you design a character of your own you can store it in place of any one of the 128 characters in the set. But the first storage site to consider is the block of thirty-two "extra" characters at the beginning. That way you won't lose the services of any of the ordinary characters in the rest of the set. For example, you could make a math symbol such as the Greek letter pi (π) and place its defining information in the slot for ASCII code (2). It will then appear on the screen when you type the key combination Open-Apple-control-B. Or, from a Basic program, you can print the string "π = 3.14159" with the statement:

```
100   PRINT CHR$(130);" = 3.14159";:REM (130 = 2 + 128)
```

As a matter of fact, the Roman and Apple fonts supplied with the Apple III each include pi as ASCII code (18).

The only disadvantage of using the 0 through 31 block is that printing a long series of such characters from a program may require a prodigious amount of extra typing. For example, a string composed of the first five ASCII characters is generated by:

```
100   G$ = CHR$(128)+CHR$(129)+CHR$(130)+CHR$(131)+CHR$(132)
110   PRINT G$
```

That gets old pretty fast. But this restriction applies only if you intend to make a printout of your program listing on an external printer. Such devices tend to be unsympathetic about exotic ASCII codes. If you are just typing in a program for your own use to run and store on disk, you can simply use the appropriate key combinations as you would for any other string. Thus:

```
100   G$ = "@ABCD":REM but holding OpenApple and control keys
110   PRINT G$
```

The Righteous Power of 2. As we noted, a complete character set contains definitions for 128 separate characters. Each character is completely defined in eight bytes of binary code. So there are 128 × 8 = 1024 bytes of information in a character set. This is exactly two 512-byte blocks, a *block* being the minimum amount of data transferred in any read or write opera-

tion between the Apple III and a disk. It's no coincidence that all these numbers are even powers of two. Around computers, if a number is not a power of two, it's probably wrong.

So the character set as a whole contains 1,024 bytes of information physically located one after another in sequence, either on a disk or as consecutive bytes in memory. The first character, ASCII 0, begins at byte 0 and runs for eight bytes (0 through 7). The next character, ASCII 1, starts at byte 8 and runs through byte 15. In general, the eight bytes defining character ASCII *N* begin at byte 8*N. Thus the definition of letter J (ASCII 74) is stored as eight bytes beginning at byte 8 × 74 = 592 (bytes 592 through 599). The next letter, K, begins at byte 600.

If one catalogs the contents of the Business Basic Disk, one finds a fontfile named *STANDARD*. This file is exactly 1,024 bytes long and consists solely of the 1,024 binary data bytes defining the standard character set. Bytes 592 through 599 of this file are found to be (in decimal form):

32 32 32 32 32 34 28 0

These eight bytes completely define the letter J. Each byte, of course, is a pattern of eight binary bits (either 0 or 1), and it is the sequence of bits, in each of the eight bytes, that is mapped onto the screen as a pattern of dots. We see the resulting pattern as a character.

The Apple III monitor screen is divided into eighty columns and twenty-four rows. Thus, there are 80 × 24 = 1920 different printing locations on the screen. Each location is a *character cell* and consists, in turn, of a seven-by-eight array of dots. In other words, each cell has eight rows of dots, and each row is seven dots wide. The eight bytes of the character definition correspond to the eight rows of the cell and, for each byte, the low order seven bits (0 through 6) are mapped onto the seven dots in the corresponding row. If a bit is 1, then that dot is turned on, and in this manner the character's form is created. The eighth or high bit (number seven of 0 through 7) is a control bit, the purpose of which we shall see in a moment.

Doing What Comes Naturally. Figure 1 shows how the eight bytes that define letter J form a map that produces the letter. Note that the character cell dots are numbered from left to right (0 through 6), whereas the corresponding bits of the binary data byte are numbered from right to left. Of course, it seems only natural to us to number screen dots from left to right. And it seems equally natural to write all our numbers, including binary numbers, the other way around, with the least significant (units) column at the right. Computers are not blessed with an intuitive understanding of what is natural and are therefore not troubled by these problems.

The high bit (number seven of 0 through 7) is not mapped onto the screen. This bit controls what happens to the other seven bits (0 through 6) when text is printed in inverse mode. If the high bit is 0, then the foreground and background colors are simply reversed, but if the bit is 1, then that row of dots flashes alternately between normal and inverse. Interesting effects can be achieved. In normal mode the high bit does nothing.

The problem of constructing a new character resolves itself into the task of converting a dot-array pattern, drawn perhaps on graph paper, into the corresponding binary data bytes, and then somehow storing those eight bytes in their proper place in a character set file. The accompanying program, *Design Characters Utility,* is intended to carry out this task in a convenient and pleasing manner.

Screen Dots		Decimal	Hex	Binary
Bit 0123456 7*				Bit 7654 3210
0000010 0	byte 0:	32	20	0010 0000
0000010 0	byte 1:	32	20	0010 0000
0000010 0	byte 2:	32	20	0010 0000
0000010 0	byte 3:	32	20	0010 0000
0000010 0	byte 4:	32	20	0010 0000
0100010 0	byte 5:	34	22	0010 0000
0011100 0	byte 6:	28	1C	0001 1100
0000000 0	byte 7:	0	00	0000 0000

* Bit #7 is a control bit.

Figure 1. Character cell bit map.

Before we can present the program itself, it is necessary to clarify some rather confusing aspects of files. A file is an external chunk of information that, for purposes of this article, we assume to be on a floppy disk. A file contains certain data, a sequence of binary data bytes, stored there for future use by the computer. But there is more to a file than the information it is intended to contain. Many files, not all, also contain other internal bytes that give the file structure. They may define where one portion of information ends and another begins, or they may tell the computer how to interpret the stored data bytes, whether as an integer, an ASCII character, or whatever. There are in fact several different patterns of file structure into which the data proper might be placed.

What's in a Name? Every file also has a name. The name, however, is not stored in the file itself. It is kept in a separate file, the disk directory. Thus a filename might be changed without affecting the contents of the file at all, altering only the directory listing. Ample facilities are provided for changing a file's name.

Similarly, every file has a *type name*. This is also an entry in the directory, stored along with the filename, the file's location and other information describing the file. There are quite a number of type names: Basic text, Basic data, Pascal text, Pascal data, Basic Program, Pascal Code, System (SOS), Foto (graphic pictures), and Font. And a few others. Naturally the type name is expected to reflect the actual physical type as it really is, which sounds simple . . . but it's not.

In theory, one is never supposed to have to mess around with type names. Their only real purpose is to detect errors, and since you obviously never make any errors you are unlikely ever to encounter them. Someone, however, might try to access what is expected to be a Basic textfile but give instead the name of a Pascal codefile.

The resulting mess is likely to bomb the program; or, much worse, the program might actually run with this misinterpreted garbage and produce meaningless results. People might be led to do ridiculous things. A former movie actor might run for president. So languages such as Basic and Pascal automatically check to be sure that the file type is the type appropriate for the command being executed. Specifically, what the language checks is the type name in the directory. If it isn't right, you get zapped with an error.

The other side of the coin is that the language assigns a type name when it creates a file. For example, *saving* a Basic program creates a file that, not unexpectedly, is type-named *Basic Program.* But there are some rather strict limitations. From within a Basic program, using the language Basic, one is allowed to create or *open* only three types of files: Basic textfiles, Basic datafiles and a type that Basic calls unknown. This type is created when a new file has been *opened* and then *closed* again prior to storage of any information. If information is stored, using PRINT#1 or WRITE#1, that act determines the type name, either *text* or *data* respectively. But without information storage or a type designation in the *create* statement, the file catalogs as "UNKNWN"—an ugly type which we shall carefully avoid.

An attempt to *open* a file of any type other than *text, data,* or *unknwn* stops the program with an error. In particular, there is no possible way from Basic to create or *open* a file with the directory type name "Fontfile." Note that we are talking about an entry in the directory, not the file itself. Strangely enough, it is possible to create and store on disk a "Basic" file, albeit an abnormal Basic file, which is identical in every respect to a Fontfile, except that the directory will call it a "Basic" file. It is then very frustrating to have a perfectly good Fontfile (internally) and have the system refuse to use it because of some name in the directory.

It Would Smell as Sweet. So, you ask, why not change the type name? After all, it's just a name, like the filename. Well, from Pascal you can do just that. The Pascal *Filer* has an *Alter*

option specifically intended for that purpose. It asks what the new file type shall be, and you reply "Fontfile," and it's done . . . if you have Pascal. How about the System Utilities Program? It also has a "Filer." In fact, the System Utilities Filer and the Pascal Filer are almost identical. The System Utilities Filer implements almost every single option available in the Pascal Filer—except this one. Did they forget, or run out of room on the disk, or did the designers think that people who use Basic have no business fooling around with changing type names? Whichever it is, the result is the same. If you don't have Pascal, you can't change the type name. But take heart; all is not lost. By sufficiently sneaky and devious means we can work around them.

All too often one runs afoul of such "legal" restraints. In fact, Pascal, with its strong typing is notorious for this. It is said that the real difference between English law and French law is that in England everything is permitted unless it is specifically prohibited while in France everything is prohibited unless it is specifically permitted. By those lights the people who design computers and computer languages are definitely French. They make everything illegal and then we all spend the next twenty-five years trying to get around it.

To use a file, any file, the computer must transfer it to RAM memory. Even if you think you are writing to one small record in a large random access file, the computer always reads in a minimum of 1 block (512 bytes), changes the small area in RAM, and writes the block back out again to the disk. So to manipulate a Fontfile the computer must read in the 1,024 bytes and put them in RAM. Basic can't handle it. Basic can access only Basic Textfiles and Basic Datafiles. But on the Business Basic Disk there is a machine language invokable module, *Download.inv,* which neatly does the job. *Download.inv* has two separate procedures: *getfont* and *loadfont. Getfont* reads a Fontfile (and, alas, only a file type-named Fontfile). It transfers the information from disk to program memory. *Loadfont* is then used to transfer the information from program memory to the system character set area where it can begin to function. In between, while the information is in program memory, it is available to Basic programs and can be altered at will.

Indelible Ink. When *getfont* transfers information from disk to program memory it maps each byte of the 1,024-byte Fontfile into a corresponding 1,024-byte section of RAM. But it can't put this chunk just anywhere or you would never find it again. One supplies an address by giving the name of a program variable. *Getfont* puts the data in the variable, and later, when you access the variable, there's the data. The variable you choose must be the right size, at least 1,024 bytes. If it's too small, *getfont* will slop data over onto surrounding memory, probably with disastrous results.

In Pascal one can specify and allocate a block of 1,024 consecutive memory positions by declaring a packed array of bytes:

```
var fontinfo: packed array [0..1023] of 0..255;
```

The data is transferred from disk to the variable fontinfo, and you can look at any byte you wish by:

```
info := fontinfo[byte_number];
```

Unfortunately, Basic doesn't have any one-byte variables. But one can achieve an equivalent result by creating an array of two-byte integers:

```
100 DIM a%(511)
```

In an array, integers are placed one right after another in sequence. Since each integer is two bytes, an array of 512 integers (0 through 511) has the required 1,024 bytes. It is essential to use the percent symbol (%), which specifies integer, because Basic otherwise will think you want "real" numbers, each of which occupies four bytes. The *getfont* procedure is designed to expect an integer array as a storage address, and

loadfont expects an integer array as the source of data to move to the character set area.

Since each integer contains two bytes, the eight bytes defining a particular character can be called for only as a sequence of four integers. To look at individual bytes it is necessary to split the integer into its two component bytes. Basic has a neat way to do it (an example of each step is on the right):

First convert the integer to a four-character hexadecimal string.

```
100 hexstring$ = HEX$(integer%)                    7210 —> "1C2A"
```

Split the string in half

```
110 leftbyte$ = LEFT$(hexstring$,2)                "1C2A" —> "1C"
120 rightbyte$ = RIGHT$(hexstring$,2)                   —> "2A"
```

Make each half a four-character string

```
130 leftbyte$ = "00" + leftbyte$                   "1C" —> "001C"
140 rightbyte$ = "00" + rightbyte$                 "2A" —> "002A"
```

Convert each string to an integer

```
150 leftbyte% = TEN(leftbyte$)                     "001C" —> 28
160 rightbyte% = TEN(rightbyte$)                   "002A" —> 42
```

This may be condensed to:

```
100 leftbyte% = TEN("00" + LEFT$(HEX$(integer%),2))
110 rightbyte% = TEN("00" + RIGHT$(HEX$(integer%),2))
```

Unfortunately, the *getfont* procedure checks up on the type name. It works well on the Fontfiles provided on the Business Basic Disk, but it will accept only a Fontfile. It will not accept any sort of "Basic" file even though the file itself is exactly right and just has the wrong type name; unless, of course, you can change the type name with Pascal. It would seem there is little point in creating such a file if it cannot be used. But the System Configuration Program has a procedure for the specific task of assigning a new character set to the SOS.DRIVER file on a boot disk. This procedure also requires a file in the Fontfile format, but fortunately it isn't too picky about the type name. It will accept the "abnormal" Basic files and thereby provides a way to use your new character set as the system set on a boot disk.

What, specifically, is an "abnormal" Basic file? Or a normal one, for that matter? Information is stored in Basic files as bytes of binary code, just as it is in any file. But Basic text and data files also have a structured network of other code bytes separating and identifying the various segments of stored data. So whenever information from a Basic file is read back into the computer, using INPUT#1 or READ#1, those structural bytes are expected, and if they are not there, the system generally goes right off the deep end. Fontfiles, however, have no structural code. They consist solely of raw binary data, 1,024 bytes of it. Basic is lost. If such a file happens to have one of the "Basic" typenames, then a Basic program can *open* it. But subsequent *read* and *input* statements choke on the unstructured code.

It is actually possible, slowly, to create a pseudo-Fontfile by placing raw binary code in a text file with the *print* statement:

```
100  PRINT#1;CHR$(0);CHR$(1);CHR$(13);
```

and so on. A *for-next* loop takes about fifteen seconds to transfer 1,024 bytes from an array in memory to the disk. But, fortunately, there is a quicker way. On the Business Basic Disk the invokable module *Request.inv* contains two procedures, *filread* and *filwrite*, which effect the same transfer in about a second. Furthermore, with *filread* one can get the information back again into memory, which *input* cannot do. And since the resulting file has the unstructured format of a fontfile and is a fontfile in all but name, it is accepted by the System Configuration Program for placement in an *SOS.DRIVER* file.

The Tail Wags the Dog. Thus we have a small conflict. In our program we will want to access the four character sets already provided with Business Basic. Lots of interesting characters there. *Getfont* in *Download.inv* loads these fonts very nicely. But *Download.inv* has no corresponding means for storing new character sets back onto the disk again. So we must use *filwrite* for storage. *Filwrite* and *filread* are very fast, very efficient. We can store a new character set and promptly recover it again. But these procedures absolutely require that the file be *opened* from Basic. And that, in turn, requires one of the three Basic files. One cannot *open* a fontfile. It's really silly. The files themselves are identical. The procedures are just hung up on type names, and these, inexplicably, cannot be changed. Well, obviously, we will just have to have two separate input procedures and two types of font storage files. But it certainly isn't elegant.

Once we have created a new character set, as by this time you are doubtless panting to do, we will need a convenient way to use it. The following program is a rewrite of Apple's *Download* program on the Business Basic Disk. The only change is that instead of reading only Fontfiles it tries first to read a Fontfile and then, if the first effort fails, it trys again with *filread*. Thus it can handle either type of Font storage.

Program Newdownload

```
10   REM — font downloading utility
20   INVOKE"/bosic/downlood.inv","/bosic/request.inv"
30   DIM o%(512):orroy$="a%"
40   INPUT"Which FONT do you wont to use? ";o$
50   IF LEN(o$)=0 THEN END
60   IF o$="?" THEN CAT:GOTO 40
70   ON ERR GOTO 110
80   expr$=CHR$(34)+a$+CHR$(34):PERFORM getfont(@expr$,@orroy$)
90   ON ERR GOTO 170
100  GOTO 150
110  ON ERR GOTO 170
120  OPEN#1,a$
130  PERFORM filread(%1,@orroy$,%1024,@count%)
140  CLOSE#1
150  PERFORM loadfont(@orroy$)
160  GOTO 40
170  IF ERR=255 THEN END: ELSE PRINT"Error. . .Error":GOTO 40
```

Overcoming Tails and Roses. *Design Characters Utility*, the program at the end of this article, provides an array of squares representing a single character cell. The dot pattern of any existing character can be displayed, or one can "X" in squares to create and modify characters. The program then calculates a new set of definition bytes and stores the new character at a specified ASCII number in the set.

The program also provides for loading fonts into a temporary integer array. Some or all of these characters may then be incorporated in the new character set array. And *Show.Set* is available to display the entire character set on the screen after temporarily activating it with *loadfont*.

The *Design Characters Utility* is designed to run if a lazy typist (surely, gentle reader, neither of us) omits all the remarks. Variable names can also be truncated to one or two characters if duplicates are avoided. But most of the statements that use integer variables really need integers, so don't omit percent symbols.

As an example of the benefits of special characters, follow these steps for a trial run; you'll also bootstrap a nice improvement in the program:

1. Be sure the program is saved on disk.
2. Make and store the characters in figure 2, labeled ASCII 1 through ASCII 11. Be sure the proper character is assigned to the proper character number. Store the new character set on disk (as, for example, "newset").
3. Run *NEWDOWNLOAD* to get the new Font working.
4. Reload the main program and change lines 340 through 370 as shown. When you have saved the new version, run it. Mark some of the squares and see what happens when the cursor falls on a previous mark.

Note: type the string characters in lines 355, 370, and 375 while holding both Open-Apple and control. These are high-bit characters—"A" = CHR$(129), and so on.

```
340  saurce$="/valume/newset":REM yaur pathname here
345  marker$=CHR$(139)
350  frame.tap$="ABCBCBCBCBCBCBD    ABD":REM gap must = 2 spaces
355  frame.md1$="I I I I I I I I    I I"
360  bit.numbr$="0 1 2 3 4 5 6      7"
365  frame.md2$="EBFBFBFBFBFBFBG    EBG"
370  frame.bat$="HBIBIBIBIBIBIBJ    HBJ"
```

Beware the Snake in the Grass. A word or two of caution: Great care must be taken in entering ASCII characters 128 through 159 directly from the keyboard, particularly in *print* statements. It is extremely easy to enter a *true* control code should Open-Apple not be firmly held. Nothing will show on the screen and you will not realize it has happened. But the code will be there in your program, waiting.

And never *retype* over a high-bit character. *Retype* reads the screen but ignores the high bit. It therefore places in memory a naked, unshifted control code; and since control codes are not echoed to the screen, *retype* simply appears not to be working. The cursor doesn't even move. But with each stroke *retype* stores in your string another drop of deadly venom that later, possibly much later, will emerge to dissolve your monitor screen into chaos too horrible to be described.

Design Characters Utility

```
10   GOSUB 300:REM initialize
20   HOME
25   VPOS=2:PRINT USING"78c";"Design Character Utility"
30   VPOS=6:PRINT"Select Optian:"
35   PRINT
40   PRINT TAB(10);"1. Laad FONT —> tempset%"
45   PRINT TAB(10);"2. Shaw character set (0..31 appear as 128..159)"
50   PRINT TAB(10);"3. Mave characters: tempset% —> charset%"
55   PRINT TAB(10);"4. Create/madify a character"
60   PRINT TAB(10);"5. Disk save new character set"
65   PRINT TAB(10);"6. Quit"
70   VPOS=20:PRINT message$
100  GET g$:REM input aptian number
105  ON VAL(g$) GOSUB 1000,1200,1400,2000,1600,500
110  GOTO 20
299  REM - - - - - - -
300  left=25:REM (subrautine) initialize
305  tap=6
310  message$=""
315  DIM charset%(511),tempset%(511),systemset%(511)
320  DIM char.def%(3),temp.bits%(15),byte.bit%(7,7),hexbyte$(7)
322  REM
323  REM    FONTfile "/basic/standard" and all 3 af
324  REM      these invakable madules must be available.
326  REM    Change pathname if nat an valume "/BASIC"
327  REM
335  INVOKE"/basic/readcrt.inv","/basic/dawnlaad.inv","/basic
     /request.inv"
340  saurce$="/basic/standard"
345  marker$="X"
350  frame.tap$="+-+-+-+-+-+-+-+-+    +-+":REM gap must = 2 spaces
355  frame.md1$="I I I I I I I I      I I"
360  bit.numbr$=" 0 1 2 3 4 5 6       7"
365  frame.md2$="+-+-+-+-+-+-+-+-+    +-+"
370  frame.bat$="+-+-+-+-+-+-+-+-+    +-+"
375  array$="systemset%":GOSUB 1800:REM laad "system" fant fram disk
380  RETURN
499  REM - - - - - - -
500  HOME:END:REM Quit aptian
999  REM - - - - - - -
1000 VPOS=8:REM (start) "Laad FONT —> tempset%"
1005 PRINT SPC(3);"—>"
1010 VPOS=20:PRINT CHR$(30);
1015 INPUT"Specify saurce pathname: ";saurce$
1020 IF saurce$="" THEN RETURN
1025 array$="tempset%"
```

```
1030 GOSUB 1800:REM laad fant from disk
1035 messoge$="Chorocter set"+fontnome$+" -> tempset%"
1040 RETURN
1199 REM - - - - - - -
1200 HOME:REM (stort) "Show set" aptian
1205 VPOS=6:PRINT"Show:  tempset% (1)?"
1210 PRINT" or  chorset% (2)?"
1215 PRINT:PRINT"ony other key terminates 'Shaw set'"
1220 PRINT"     (blonk screen indicates set empty)"
1225 PRINT:GET g$
1230 IF g$<>"1" AND g$<>"2" THEN RETURN
1235 IF g$="1" THEN orray$="tempset%":ELSE array$="charset%"
1240 PERFORM laadfont(@array$)
1245 FOR i=32 TO 47
1250   FOR j=0 TO 7
1255     number=i+16*j
1260     PRINT USING 1290;number,CHR$(number);:REM terminal ";"
1265     NEXT j,i
1270 GET g$
1275 arroy$="systemset%"
1280 PERFORM loadfont(@array$)
1285 RETURN
1290 IMAGE 3#,2R,5X
1399 REM - - - - - - -
1400 VPOS=10:REM (stort) "Mave tempset% —> charset%"
1405 PRINT SPC(3);"—>"
1410 VPOS=20:PRINT CHR$(30);
1415 INPUT"Specify first ASCII number (default = 0): ";first$
1420 IF first$="" THEN first$="0"
1425 INPUT"Specify lost ASCII number (default = 127): ";last$
1430 IF lost$="" THEN lost$="127"
1435 INVERSE:PRINT" — warking — ":NORMAL
1440 FOR i=4*VAL(first$) TO 4*VAL(last$)+3:charset%(i)=tempset%
     (i):NEXT i
1445 messoge$="ASCII ("+CONV$(first$))+".."+CONV$(VAL(last$))
1450 message$=message$+") —> charset%"
1455 RETURN
1599 REM - - - - - - -
1600 VPOS=12:REM (start) disk save aptian
1605 PRINT SPC(3);"—>"
1610 VPOS=20:PRINT CHR$(30);
1615 INPUT"Specify pathname far storage file: ";dest$
1620 OPEN#1,dest$:CLOSE#1:DELETE dest$:REM safer ta delete than averwrite
1625 CREATE dest$, TEXT:OPEN#1,dest$
1630 array$="charset%"
1635 PERFORM filwrite(%1,@orray$,%1024)
1640 CLOSE#1
1645 messoge$="Character set stared in "+CHR$(34)+dest$+CHR$(34)
1650 RETURN
1799 REM - - - - - - -
1800 fantname$=CHR$(34)+saurce$+CHR$(34):REM (start) laad fant fram disk
1805 ON ERR GOTO 1820
1810 PERFORM getfant(@fontname$,@array$):REM perhaps it's o FONTFILE
1815 OFF ERR:GOTO 1835
1820 OFF ERR:OPEN#1,saurce$:REM if nat FONTFILE then try as Basic file
1825 PERFORM filreod(%1,@array$,%1024,@caunt%)
1830 CLOSE#1
1835 RETURN
1997 REM
1998 REM ---------- "Create/Madify" aptian (main) ----------
1999 REM
2000 GOSUB 2200:REM get ascii number: get specified character
2010 IF ascii$<>"" THEN GOSUB 2300:REM canvert integers —> bits
2015 GOSUB 2400:REM draw frame
2020 IF oscii$<>"" THEN GOSUB 2500:REM mark bits on frame
2025 GOSUB 2600:REM create/madify
2030 GOSUB 2700:REM scan frame and read bits
2035 GOSUB 2800:REM convert bits ta integer wards —> new char. definitian
2040 GOSUB 2900:REM present defining bytes and shaw new character
2045 GOSUB 3000:REM stare new character
2050 RETURN
2197 REM
2198 REM        ---------- Subroutines ----------
2199 REM
2200 VPOS=11:REM (stort) get ascii number
2205 PRINT SPC(3);"—>"
2210 VPOS=20:PRINT CHR$(30);
2215 INPUT"Enter <RTN> far new; ar Specify ASCII number: ";ascii$
2220 IF ascii$="" THEN RETURN
2225 PRINT:INVERSE:PRINT" — computing array pattern — ";:NORMAL
2230 FOR index=0 TO 3:REM get specified character
2235   chor.def%(index)=charset%(4*VAL(ascii$)+index)
2240   NEXT index
2245 RETURN
2298 REM - - - - - - -
2299 REM (stort) canvert chorocter definitian to bits
2300 FOR wardnum=0 TO 3:REM take each 16 bit ward af char.def%
2305   ward%=char.def%(wardnum)
2310   langint&=CONV&(ward%):REM use lang integers far camputatian
2315   IF longint&<0 THEN langint&=langint&+65536
2320   FOR bit=0 TO 15:REM canvert ta temparary array af 16 bits (= 1 ar 0)
2325     temp.bits%(bit)=CONV%(langint& MOD 2)
2330     langint&=langint& DIV 2
2335     NEXT bit
2340   FOR bit=8 TO 15:REM bits fram high byte ta raw of byte.bit% orray
2345     byte.bit%(2*wordnum,bit-8)=temp.bits%(bit)
2350     NEXT bit
2355   FOR bit=0 TO 7:REM bits fram low byte ta next raw
2360     byte.bit%(2*wardnum+1,bit)=temp.bits%(bit)
2365     NEXT bit
2370   NEXT wardnum
2375 RETURN
2399 REM - - - - - - -
2400 HOME:REM (stort) draw frame
2405 PRINT"<cursars>; <'X'> places; <Space> clears"
2410 VPOS=24:PRINT"<RTN> accepts; <ESC> aborts";:REM nate: terminal ";"
2415 WINDOW left-2,tap-2 TO left+20,tap+16
2420 HOME
2425 PRINT SPC(1);bit.numbr$
2430 PRINT SPC(1);frome.top$
2435 FOR index=0 TO 6
2440   PRINT index;frome.md1$
2445   PRINT SPC(1);frame.md2$
2450   NEXT index
2455 PRINT"7";frame.md1$
2460 PRINT SPC(1);frome.bat$
2465 TEXT:REM restare full screen viewpart
2470 HPOS=left:VPOS=tap
2475 RETURN
2499 REM - - - - - - -
2500 FOR raw=0 TO 7:REM (start) mark bits an frame
2505   VPOS=tap+2*raw
2510   FOR calumn=0 TO 6
2515     HPOS=left+2*column
2520     IF byte.bit%(row,column)=1 THEN PRINT morker$;
2525     NEXT calumn
2530   HPOS=left+17
2535   IF byte.bit%(row,7)=1 THEN PRINT marker$;
2540   NEXT row
2545 HPOS=left:VPOS=tap
2550 RETURN
2599 REM - - - - - - -
2600 PRINT CHR$(5);:GET g$:PRINT CHR$(6);:REM (stort) create/madify
2605 IF g$="x" OR g$="X" THEN PRINT marker$;CHR$(9);
2610 IF g$=" " THEN PRINT" ";CHR$(9);
2615 g=ASC(g$)
2619 REM maving obaut an frame
2620 IF g=8 THEN HPOS= HPOS-2
2625 IF g=21 THEN HPOS= HPOS+2
2630 IF g=11 THEN VPOS= VPOS-2
2635 IF g=10 THEN VPOS= VPOS+2
2639 REM wrap sides and skip gop
2640 IF HPOS<left THEN HPOS=left+17
2645 IF HPOS>left+17 THEN HPOS=left
2650 IF HPOS=left+14 THEN HPOS=left+17
2655 IF HPOS=left+15 THEN HPOS=left+12
2659 REM wrap top and bottom
2660 IF VPOS<top THEN VPOS=tap+14
2665 IF VPOS>tap+14 THEN VPOS=top
2670 IF g=27 THEN POP:RETURN:REM abart —> main menu
2675 IF g<>13 THEN 2600
2680 RETURN
2699 REM - - - - - - -
2700 HPOS=left-6:V=POS=23:REM (start) scan frame and read bits
2705 INVERSE:PRINT" — scanning and computing — ";:NORMAL
2710 FOR raw=0 TO 7
2715   VPOS=top+2*row
2720   FOR column=0 TO 6
2725     HPOS=left+2*column
2730     PERFORM readc(@value%):REM reads screen with invakable madule
2735     IF value%=32 THEN this.bit=0:ELSE this.bit=1
2740     byte.bit%(raw,column)=this.bit
```

TURNKEY SOFTWARE
presents

It will take skill, perception, and quick reactions to achieve ***your*** goal; survive through **CEILING ZERO.**

CEILING ZERO is a high-speed action game, written completely in 6502 machine code, featuring ***fast*** High Resolution ***color*** graphics, game speaker sound effects, and over 15 levels of difficulty.

CEILING ZERO
Created by Stephen Warady

Exclusively distributed by
SOFTSEL
8295 So. La Cienega Blvd., Inglewood, CA 90301
(800) 421-5770 In Calif. (800) 262-4242

```
2745      NEXT column
2750    HPOS=left+17
2755    PERFORM reodc(@volue%)
2760    IF volue%=32 THEN byte.bit%(row,7)=0:ELSE byte.bit%
        (row,7)=1
2765    NEXT row
2799  REM - - - - - - -
2800  FOR wordnum=0 TO 3:REM (stort) convert bits to char.def% integers
2805    FOR bit=8 TO 15:REM high byte first, then low byte
2810      temp.bits%(bit)=byte.bit%(2*wordnum,bit-8)
2815      NEXT bit
2820    FOR bit=0 TO 7
2825      temp.bits%(bit)=byte.bit%(2*wordnum+1,bit)
2830      NEXT bit
2835    reol=0:REM "reol" numbers permit exponentiotion in 2845
2840    FOR bit=0 TO 15:REM convert binory to integer
2845      reol=reol+temp.bits%(bit)*2 ^ bit
2850      NEXT bit
2855    word%=TEN(HEX$(reol)):REM convert to stondord 2's complement
        integer
2860    chor.def%(wordnum)=word%
2865    NEXT wordnum
2870  RETURN
2899  REM - - - - - - -
2900  HOME:VPOS=8:REM (stort) disploy new chorocter ond defining info
2905  PRINT"New chorocter definition:"
2910  PRINT:PRINT TAB(5);"As 4 integers: ";
2915  FOR i=0 TO 3:PRINT chor.def%(i);"  ";:NEXT i:PRINT
2920  PRINT:PRINT TAB(5);"As 8 hexodecimol bytes:  ";
2924  REM chonge 4 integers to orroy of 8 hexodecimol bytes (2 chor strings)
2925  FOR i=0 TO 3
2930    hexbytes$(i*2)=LEFT$(HEX$(chor.def%(i)),2)
2935    hexbyte$(i*2+1)=RIGHT$(HEX$(chor.def%(i)),2)
2940    NEXT i
2945  FOR i=0 TO 7:PRINT hexbyte$(i);"  ";:NEXT i:PRINT
2950  PRINT:PRINT"New chorocter's oppeoronce:  ";
2955  GOSUB 3200:REM lood ("octivote") new chor os ASCII 0
2960  PRINT"Normol: ";CHR$(128);" Inverse: ";
2965  INVERSE:PRINT CHR$(128):NORMAL
2970  VPOS=20:PRINT"Do you wont to revise chorocter? (y/n)"
2975  GET g$
2980  IF g$="n" OR g$="N" THEN RETURN
2985  IF g$<>"y" AND g$<>"Y" THEN PRINT CHR$(7);:GOTO 2975
2990  HOME:IF oscii$="" THEN oscii$="31"
2995  POP:GOTO 2010
2999  REM - - - - - - -
3000  HOME:VPOS=10:REM (stort) store new chorocter
3005  IF oscii$="" THEN 3035
3010  PRINT"New chorocter —> ASCII number ";VAL(oscii$);" (yes/no)? ";
3015  GET g$
3020  IF g$="y" OR g$="Y" THEN 3045
3025  IF g$<>"n" AND g$<>"N" THEN PRINT CHR$(7):GOTO 3015
3030  PRINT"N":PRINT
3035  INPUT"Store os whot ASCII number? (RTN for none): ";oscii$
3040  IF oscii$="" THEN messoge$="chorocter discorded":RETURN
3045  FOR index=0 TO 3:REM store in chorset%
3050    chorset%(4*VAL(oscii$)+index)=chor.def%(index)
3055    NEXT index
3060  messoge$="Chorocter stored os ASCII number "+CONV$(VAL(oscii$))
3065  RETURN
3190  REM  ************************************************
3191  REM
3192  REM on olternote method of using chorocter definition.
3193  REM see Stondord Device Driver Monuol p. 71
3194  REM        ond "REQUEST.DOC" progrom on Business Bosic Disk.
3195  REM
3196  REM
3200  ctrlnum=17:REM control code #
3205  count=1:REM the number of chorocters being looded
3210  ctrlist$=CHR$(count):REM this will be first byte of ctrlist$
3215  oscii.num=0:REM the (only) chorocter will be ploced os ASCII 0
3220  ctrlist$=ctrlist$+CHR$(oscii.num):REM odd oscii.num os 2nd byte
3225  FOR i=0 TO 7:REM odd the 8 definition bytes
3230    ctrlist$=ctrlist$+CHR$(TEN("00"+hexbyte$(i)))
3235    NEXT i
3239  REM "octivote" chorocter
3240  PERFORM control(%ctrlnum,@ctrlist$)".console"
3245  RETURN
3246  REM  ************************************************
```

VENTURES WITH VisiCalc

BY JOE SHELTON

Interested in doing investment analysis before you invest your money? Like to figure out how to do your budgeting and save on taxes? Ever wanted to use a PERT chart? You can do these things easily with *VisiCalc*. Over the next few months, we'll build these models and more. We'll begin this month by looking at some of *VisiCalc*'s less understood functions and how to use them.

VisiCalc III versus Apple II VisiCalc. First a note about different versions of the program. Apple III *VisiCalc* and *VisiCalc* for the Apple II in DOS 3.3 are nearly identical, but the Apple III version has some advantages over the *Apple II* version, all the result of hardware capabilities. The principal advantage is the larger memory offered by the III. Here's a chart summarizing the differences:

Memory Configuration	Maximum File Size
Apple II, 48K	18K
Apple II with 16K RAM card	34K
Apple III, 128K	64K
Apple III, 256K	192K

If there's one thing creative *VisiCalc* users will tend to do, it's to try to create a model larger than the available memory will allow. The 256K Apple III will permit users to develop just about any model they might need.

There are three other differences between Apple III *VisiCalc* and *VisiCalc* for the Apple II. The III has an eighty-column display that can show eight nine-character-wide *VisiCalc* columns, rather than the four columns that are displayed on the Apple II. The III also displays upper and lower case characters on the screen (as well as in hardcopy output). Finally, the *VisiCalc III* user has the advantage of Apple III's four cursor control keys.

Although peripherals can equip the Apple II with all but the last of these capacities, the Apple II *VisiCalc* cannot take advantage of them.

Both versions of the program—*VisiCalc III* and the sixteen-sector version for the Apple II—incorporate the search functions and Boolean functions that we'll be discussing in this column and in columns to follow.

Back to the Beginning . . . What Is VisiCalc? Those of you who have been using *VisiCalc* for a while might want to skip on to **The New VisiCalc**, but for those who are not very familiar with it, read on. *VisiCalc* is a simple mathematical tool. Think of it as a large electronic spreadsheet with 254 rows and 63 columns. The intersection of a row and a column is called a cell; there are 16,002 of them!

The large black area that fills the bottom three-fourths of the monitor screen is the visible worksheet. It might help to think of your screen as a window that slides around on the worksheet. Using the cursor control keys on the III and the left and right arrow keys and spacebar on the Apple II, you can move the cursor around the screen and see the letters and numbers that define rows and columns change as the cursor reaches the edge of the screen. The position of the cursor marks the place where you enter information into *VisiCalc*.

So much for what you see; now what does it do? You can enter three types of information in any cell. The first type is text or labels. The second is values (numbers). The third is formulas or functions (which are treated as values). That's it. That's all *VisiCalc* does. So why the big hoopla? Well, let's see if we can touch on it.

The text part is simple. Text allows you to label rows and columns. Values are simple, too. Values are just numbers, but functions and formulas use them for their computations. And functions and formulas are the real power of *VisiCalc*. They take values and solve for answers. As we continue, you'll see many different ways to use functions and formulas.

Templates are combinations of text, values, functions, and formulas that are developed by the user to solve problems. Once completed, templates can be saved to disk and recalled at a future date. Later in this article we'll design two simple templates, one to compute net present values, the other to figure simple interest.

Is it hard to design a *VisiCalc* template? Well, if you can do it on paper, you'll probably find it easier to do with *VisiCalc*.

VisiCalc is useful for those things like tax computations, forecasting, and budgeting, where you must do a lot of computation or where you want to change one or more variables and see the results. This is its real forte—the ability to build a model and have it do the computations. It is also useful for changing variables (like interest rates) and seeing what different results occur. This kind of analysis is commonly called "what if" analysis.

The New VisiCalc. What makes the new *VisiCalc*—the sixteen-sector version for the Apple II or version 1.1 for the Apple III—more powerful? The most important advance is in the area of functions. *Functions* are common formulas or algorithms that allow complex procedures to be completed simply, quickly, and—most important—automatically. *VisiCalc* now has some new search functions, as well as functions that simulate Boolean logic.

Search Functions. Let's start with the search functions. They're called Lookup and Choose. Boot *VisiCalc* so you can follow along. You'll learn more about this if you enter the examples whenever you see them and then experiment with them.

Search functions are handy for pulling data from tables. Figuring such things as payroll deductions (see last month's Ventures with VisiCalc) or insurance premiums are typical applications. Another way to use a search function might be for a calendar lookup, where you could enter dates (for things like interest calculations) and determine how many days, months, or years are between specified dates. We'll discuss some of these uses of search functions in later columns. Right now let's take a look at Lookup.

@Lookup. The entry that calls the *@lookup* function looks like this: @Lookup(v,range). The v represents a value; it can be a number, a reference to another cell, or another formula or function. The range represents a portion of a row or column that is defined by its beginning and ending coordinates. To specify the range, you enter the beginning coordinate, a period, and the ending coordinate. *VisiCalc* automatically displays the period as a three-dot ellipsis.

Muse Software Adds A New Dimension To Your Company's Apple™ Computer

Software by Muse is designed to toke your computer to the limit, so there's no limit to whot it con do for you.

Think of oll you could do with just these four Muse business programs.

Super-Text™

You'll leorn the bosics of text editing in mere minutes, then turn your computer into o word processor with feotures so odvonced it will continue growing with you yeor after year.

Whot kind of advonced features? Split screen for editing lorge documents, moth mode for operating stotisticol reports, optionol file linking for global seorch and print operotions, preview mode for reviewing line endings and page breaks before printing, form letter generotion ond moiling list monogement odd-on modules plus much more.
Only $150.00

Form Letter™

No motter how mony form letters and documents you generate, eoch con carry on individuol nome ond oddress, refer to your correspondent by name within the text, and include o speciol poragroph for thot person alone. The progrom will olso formot ond print on envelope ot the end of the letter. An add-on module to Super Text. **Only $100.00**

The Address Boak™

No more dogeored oddress files filled with scrotch-outs ond erasures. Our Address Book is on odd-on to Super Text that lets you add or change oddresses; list phone numbers; print lobels, sort oddresses; access stote ond your cotegory codes. Or use the Address Book olone as o simple moiling list.
Only $49.95

Data-Plat™

Data Plot converts hard-to-understond figures into cleor-at-o-glonce graphics. And you con choose the formot that works best: bar chort, line graph, scotter diogrom or pie chart —all in color, of course. **Only $59.95**

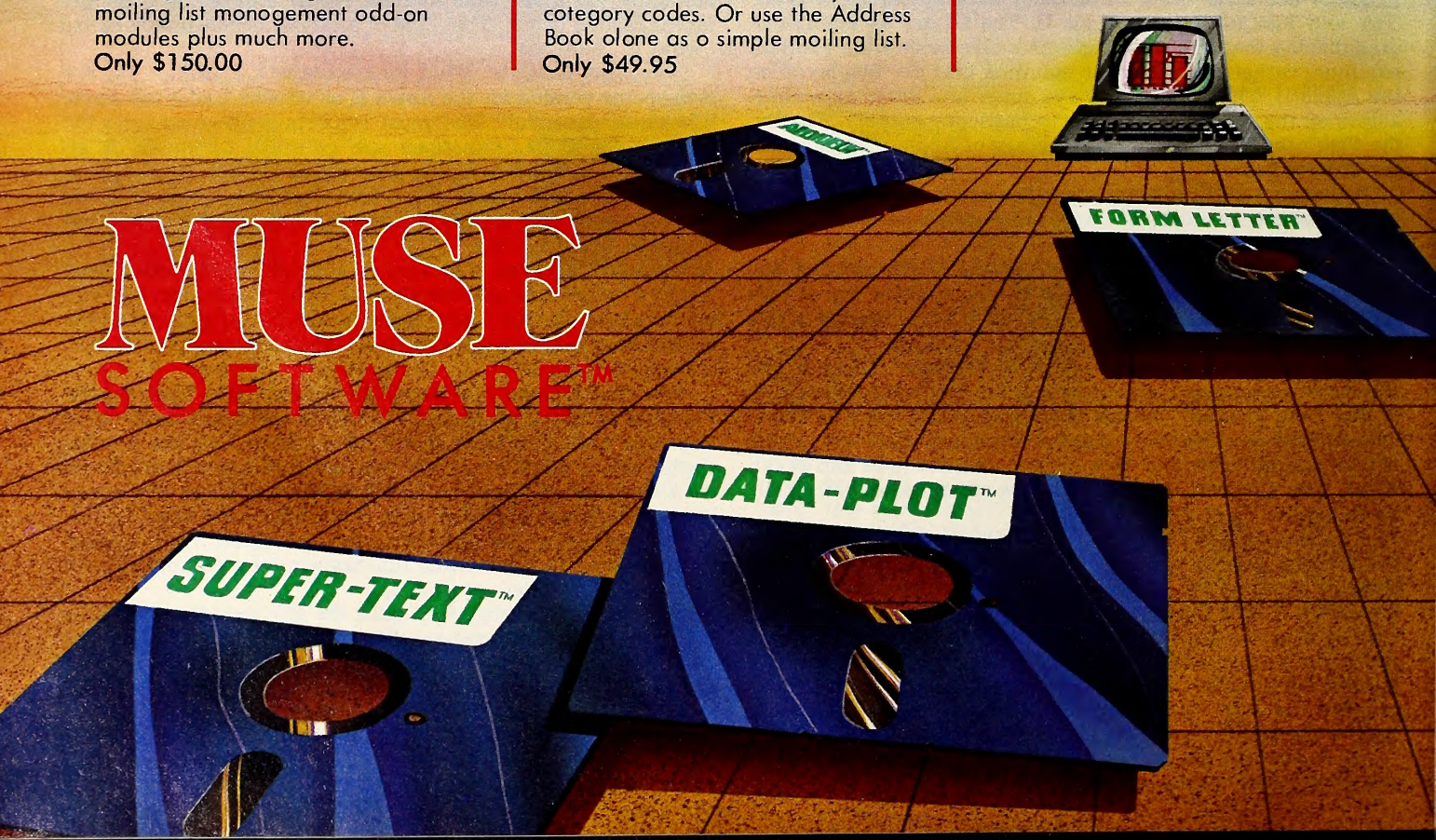
Get all the details about these and other advanced MUSE programs from your local dealer. Or call MUSE direct.

MUSE SOFTWARE™

347 N. Charles Street
Baltimore, Maryland 21201 (301) 659-7212

The best business software comes from Muse.

For Apple II™ or Apple II Plus™ with 48K.
Apple II is o trodemork of Apple Computer Corp.



	A	B	
1	5		<— The search value (v)
2			
3	89		<— @LOOKUP (A1,A5...A10)
4			
5	1	45	
6	3	12	
7	5	89	<— The value displayed by @LOOKUP in A3
8	7	1	
9	9	100	
10	11	14	

Figure 1.

	A	B	
1	6	7	<— The search value
2			
3	89	44	<— @CHOOSE (B1,55,B5. . .B10,(A3+1)
4			
5	1	45	
6	3	12	
7	5	89	
8	7	1	
9	9	100	
10	11	44	<— The seventh value in the list; displayed in B3

Figure 2.

Let's see an example of how this works. Enter the following into the indicated cells:

A3: @LOOKUP (A1,A5...A10)
A5: 1
A6: 3
A7: 5
A8: 7
A9: 9
A10:11
B5 through B10: random numbers (your choice)
A1: a value between 1 and 11 (the example below shows 5).

Your result should look like figure 1.

Here's what happened. Lookup took the value in A1, searched A5 through A10 until it found five, then looked in the cell to the right of the five (B7) and displayed the value it found there—in this case, eighty-nine.

To add flexibility to structuring your template, the range can be across a row. If the range in our example had been a row, Lookup would have searched across the row until it found the value it was looking for and then displayed the value from the cell directly below. Just remember: the results of a lookup will be taken from cells either to the right of or below the source range.

Now replace the five in A1 with a six. You'll notice that the number six doesn't appear in A5 through A10 and yet the value eighty-nine was still returned or displayed in cell A3 (we'll use the terms *return* and *display* synonymously). This is because Lookup searches the range until it finds a number greater than the value (v) in the function statement and then uses the preceding value for the lookup. In our example, it searched until it found seven—the first number in the table greater than six—and then it returned the number in the cell next to five.

Try some other values and notice the results. If you enter a value greater than eleven (the maximum value in the list A5. . .A10), the function will always return forty-four. Conversely, if you enter a value less than the lowest value (in this case, one), the function will return an NA.

If you are going to use Lookup in a template, you'll want to remember the latter idiosyncrasy. Any cell that refers to a cell holding the value NA will also return an NA. Any formulas that include a cell displaying an NA will not return a value, but will return NA. To prevent problems, you will want to make sure the lookup range includes maximum and minimum values that exceed any number likely to be used as the lookup value.

@Choose. *VisiCalc*'s lookup capability has been augmented with a second type of search function, called choose. Choose accomplishes much the same thing as *lookup,* but it requires only one column. Another difference between lookup and choose is that the former requires a range of contiguous values, whereas choose allows a list to be comprised of values, cell coordinates, and functions in an arbitrary order.

The function is written: @choose(v,list). Value (v) is the same as in lookup. It can be a value, a reference to a cell, a formula, or a function. A list is a combination of values and ranges.

Take our example for lookup and do the following:

In cell B3, enter @choose (B1,55,B5. . .B10,(A3+1)). In cell B1, enter 7.

Your screen should look like figure 2.

Choose took the value in Cell B1, which was 7, and, starting with the first value in the list, which was 55, counted seven values, returning the value 44 held in cell B10. If the search value at B1 were changed to 8, choose would go to the eighth item in the list, A3+1, and return the value 90 (since A3 holds 89).

If v is less than one or greater than the number of values in the list, choose NA will return an NA. Once again, you should carefully consider how this might affect your template.

The choose function is more flexible than lookup in many ways, allowing you to enter a list of values that aren't contiguous or in any particular order. Using choose, you can look up values in places in a template where there isn't room to use two columns or rows for a *lookup* table.

Other Functions. The other functions are a mix of mathematical tools that will save typing effort (@SUM, for example) or provide results otherwise unobtainable (for example, @LN, which returns the natural logarithm). As with lookup and choose, the argument of these mathematical functions—the value upon which they act—may be a range, function, formula, cell coordinate, or number. A list can be any combination of values.

Arithmetic and Trigonometric Functions. Here are some other functions whose purposes are not always clear.

@ABS(v) returns the absolute value of a number—its value without reference to sign. Both *@ABS(20)* and *@ABS(−20)* will return 20. The assumed positive sign of the first argument and the negative sign of the second are both removed, and the value is displayed as an absolute number.

One obvious use for @ABS is in *VisiCalc*'s graphic mode. The graphic mode will display a value entered in a cell as a number of asterisks equal to the value—or to the window width, if that is less than the value of the cell. In order to plot −3, you must first convert it to an absolute or positive value. @ABS makes that task simple.

@INT)(v) returns the integer portion of a value. If you take the *@INT* of a number like 3.9, you'll get 3. If you are developing a financial application that uses the eleven to twelve place precision of *VisiCalc*, you might get rounding errors in some calculations. To avoid such rounding errors, it's sometimes useful to make *VisiCalc* less precise, with the help of *@INT*. The formula *@INT((v)*100+.5))/100* reduces the calculating precision to two decimal places.

@COUNT(list) returns the number of values in list. If you were keeping track of a list of invoices by invoice number, *count* could total the number of invoices entered.

Net Present Value. *@NPV(dr,list)* calculates the net present value of future cash flows, based upon a discount rate. Wonder what that means in English? Let's take an example.

Suppose you have a choice between receiving $1,000 today or $100 a year for ten years. Intuitively you would probably choose to take the $1,000 today. That would be the smart thing to do—but do you know why? It may seem like greed, but mathematically it works this way:

Clear the *VisiCalc* screen, with /CY. In each cell from A1 to A10, enter the value 100. At B1, enter .10—the discount rate. At B3, enter @NPV(B1,A. . .A10). The screen would now look like figure 3.

The NPV is $614.4567. At a 10 percent discount rate (ap-

	A	B	
1	100	.1	<— discount rate
2	100	614.4567	<— NPV
3	100		
4	100		
5	100		
6	100		
7	100		
8	100		
9	100		
10	100		

Figure 3.

proximately equal to inflation) the $1,000 is really only worth $614 and some change to you at the end of ten years. Why? Because of the discount rate—inflation—the $100 you receive each year will buy less than $100 will buy today. The amount of groceries you could buy with the $100 you receive in the ninth year is no more than the amount of groceries you buy with $40.81 today.

You will find that NPV will be an extremely useful function for doing income investment analysis.

Let's stop for a second and think about what can be done with that $1,000. If you put it in a savings account at 5.75 percent interest (compounded yearly), how much would it be worth in ten years? Here's how to find the answer.

Clear the *VisiCalc* screen, with /CY.Enter /GF$ to set the spreadsheet to display in dollar-and-cents format. Enter /GOR to let *VisiCalc* calculate across rows.

Now enter >A10 and return to move the cursor to cell A10, and at that location enter .0575 (5.75 percent) and /FG. At A11, enter 1000, and at B11 enter the formula +A10*A11. The result of that formula—$57.50—is the interest you'll earn on your $1,000 in year 1.

At A12, enter the formula +A11+B11. That value—$1,057.50—represents your balance at the beginning of the second year.

At B12, enter +A10*A12, to get $60.81, the interest you'll earn in the second year. Replicate this last formula, in A12, into A13 through A21, and replicate the formula at B12 into B13 through B20. The replication entries are:

```
A12: /R RETURN
     A13.A21 RETURN
     RR
B12: /R RETURN
     B13.B20 RETURN
     NR
```

You can now determine your balance at the end of ten years. It is the $1,749.06 in cell A21.

This is one method of doing a simple interest calculation, with the interest compounded yearly. Change the interest rate and/or the initial $1,000 deposit and see what different interest rates and deposits can earn you. You can expand this template, using the replicate command, to see what your balance will be in any year.

Follow your intuition; take the $1,000 today and invest it.

Forward and Circular References. When we started to build the simple interest model, you entered the command /GOR. Why? When *VisiCalc* computes, it starts at the top left of the spreadsheet and either calculates down the columns (A1,A2,A3. . .BK254), or across the rows (A1,B1,C1. . .BK254). If you look at cell A12 in our simple interest calculation, you'll see the formula +A11+B11. By entering the command /GOR, we have made certain that *VisiCalc* will compute across row 11 before computing row 12. Any change in A11 or B11 will be completed before calculations start in A12.

Let's think about what happens when we change to calculating by columns (/GOC). *VisiCalc* calculates down the columns. It will calculate all of column A before calculating column B. In cell A12 there is a reference to A11 and B11.

If *VisiCalc* is calculating by columns, it will figure both A11 and A12 before it gets to B11. If there's a change that affects the value of B11, the formula +A11+B11 in cell A12 won't see the result of that change until the next time *VisiCalc* recalculates. This situation is called a *forward reference.*

In the situation just described, the value that the formula in A12 returns will be wrong until the next recalculation. Any other formula that uses the value in A12 will also be wrong until the next recalculation. Enter /GOC and change either the interest rate (now .0575) or the principal (now 1000) and see the result. The numbers change and all seems fine, doesn't it? If you press !, you will manually cause *VisiCalc* to recalculate. Try it, and you'll see that in this model, it takes six presses of the ! key to ensure that all the numbers are accurate.

There is another type of reference whose result is even more interesting. A circular reference does exactly what the name implies—it goes in circles. Enter >A50 RETURN to move to a clear area of the spreadsheet. At that cell, enter +B50+10, and at B50, enter +A50+10.

Can you see what is going to happen? Every time you press ! or enter a number somewhere else in the spreadsheet, these two numbers will change. Press ! a few times and watch the results. At every recalculation, each cell is taking the value in the opposite cell and adding 10 to it. This will continue indefinitely; the display on the screen will eventually show a series of > symbols, meaning that there are more digits in the value than spaces in the cell to display them. Any other cell that looks at these two cells will still use the value currently computed in the cell. You can see that this type of problem is to be avoided at all costs.

Remember, always test new models by pressing the ! key a few times with /GOC and /GOR, to determine which recalculation order works best and to check that you don't have any circular references.

/S# and /SQ. One note before we quit (/SQ). In *VisiCalc*, the # is the symbol for data interchange. This column will be a focal point for many ideas and applications on many products. In future columns, we will look at a number of different application templates, as well as discussing specific methods or algorithms for accomplishing different tasks.

That is our part of the data interchange (/S#). We need data

TASC™ The Applesoft* Compiler. It turns your Apple into a power tool.

Step up to speed. TASC, the Applesoft Compiler, converts a standard Applesoft BASIC program into super-fast machine code. By increasing program execution speed up to 20 times, Microsoft gives you a power tool for Applesoft BASIC programming.

Highest capacity available. TASC will compile and run larger programs than any other Applesoft Compiler. As a disk-based system, it doesn't require the simultaneous presence of compiler and program in memory. The memory you save allows you to compile significantly bigger programs.

Power without bulk. Code expansion of up to 100% severely restricts other compilers. TASC's special code compression schemes typically limit code expansion to only 25%. You'll really appreciate that with complex programs or programs that utilize Apple's hi-res graphic pages.

More BASIC power. TASC's powerful new commands increase Applesoft BASIC programming capability. Chain with COMMON allows compiled programs to share variables, so a main menu supports several programs in a single runtime environment. TASC's True Integer Arithmetic and Integer FOR...NEXT capabilities maximize the execution speed of compiled programs. TASC's near total compatibility with Applesoft speeds compilation of existing programs with little or no modification.

What about mistakes? You perfect your programs interactively with Applesoft. If something does slip by, TASC recovers from errors discovered in compilation and traps all runtime errors. It even permits graceful interruptions during compilation.

See for yourself. Ask for a demonstration of TASC at your Microsoft dealer. Discover the software package that turns your Apple into a power tool.

*Applesoft is a trademark of Apple Computer, Inc.

MICROSOFT
CONSUMER PRODUCTS

A Division of Microsoft Inc.
10700 Northup Way • Bellevue, WA 98004

interchange from you, too. As you discover modeling hints, new uses for commands and functions, different applications, and so on, please write us about them. We can all learn from sharing our ideas. ◨

As several readers of this column noted, the description of the IRA model in February's Ventures with VisiCalc had a few errors. The format at cell B5, where the interest rate per compounding period was displayed, should have been /FR; in dollar format the cell read 0.00. For neatness and consistency, B2 should also have been in flush right.

More seriously, the instruction, on page 29, to perform a /R:A14.A56:NNRN should have read, "With the cursor still at B13, type /R;B13.B56:NNRN. . .". Our apologies for the confusion this must have created.

One reader suggested that for the sake of accuracy the number of compounding periods per year should have been entered as 365.25 instead of 365; it turns out that change made a difference of about $21 to the future value of the investment at a forty-four-year term.

A number of readers accepted our challenge to discover a simpler means of calculating the future value of the investment assuming a constant annual flow of deposits. Some of the solutions were quite ingenious; a number of them provided valuable additional information to the model.

Jon Ruppert, W. J. Friesen, John Howsam, Samuel Abramson, Shirley Fisher, David Scannell, and Dan Moore suggested the following procedure: use column C to generate a running sum of the values in column B. To do this, first eliminate everything to the right of column B in the model as we described it. Second, enter @SUM(B12.B13) at C13. Then replicate that formula with /R:C14.C56:NR. The whole model thus fits into 4K.

Wolfgang Gunther offered an even more compact model, shown in figure 1. And Richard Moy showed how to accomplish the same results with an ingenious mix of forward, backward, and circular references. His model, shown in figure 2, requires that you first enter a 0 at line 8. Then enter a 1 at that cell and hit !. Each time you recalculate, the model will increment the term of the account and calculate the future value. When the model is reloaded from disk, you'll need to blank out B5 through B7 and then reenter the formulas. The easiest way to do that is to replicate the three cells, then delete them, then replicate them back into B5 through B7. Try it; you'll like it.

Enter Column A	Enter Column B	Numeric Result
1 PRINCIPAL $	:/F$ 2000	2000.00
2 ANNUAL INTEREST %	:14	14
3 YRS OF DEPOSIT	:44	44
4 INTRST PERIODS/YR	:365	365
5 ADD/SUBT ANNUAL $	:/F$ 2000	2000.00
6 "----------	:"----------	----------
7 ADD/SUBT MONTHLY	:/F$ +B5/12	166.67
8 INTEREST/PERIOD	: +B2/B4	.038356164383
9 PERIOD INT FACTOR	: 1+(B8/100)	1.0003835616
10 ANNUAL INT FACTOR	: +B9^B4	1.1502428583
11 COMPOUNDING FACTOR	: +B10^B3	472.86808318
12 "----------	:"----------	----------
13 ANNUAL YIELD $	::/F$ +B1*(B10-1)	300.49
14 COMPNDED CAPITAL	:/F$(B1*B11)+(B5*(B11-1) /(B10-1)):	7227140.67
15 AMORTIZATION $/YR	:/F$((B10-1)/(B11-1)+ B10-1)*B1:	301.12

Figure 1.

Enter Column A	Enter Column B	Numeric Result
1 CONTRIBUTION	:2000.00	2000.00
2 INTEREST	:14	14
3 PERIODS	:365	365
4 INT/PER	:00038356164383	.00038356164383
5 FUTURE VAL	: +B8*B6*((1+B4)^B3)	7225140.65
6 PRINCIPAL	: +B1+B5	7227140.65
7 TERM	: (1+B7)*B8	44
8 MULTIPLIER	: 0=RESET, 1=RUN	1

Figure 2.

Aye! Come aboard for the fight of your life

SWASHBUCKLER

Fantastic HI-RES sword fighting

You're in a duel to the death against a black-hearted pirate crew. Look out! Here they come swinging and slashing with their pikes, swords, and knives.

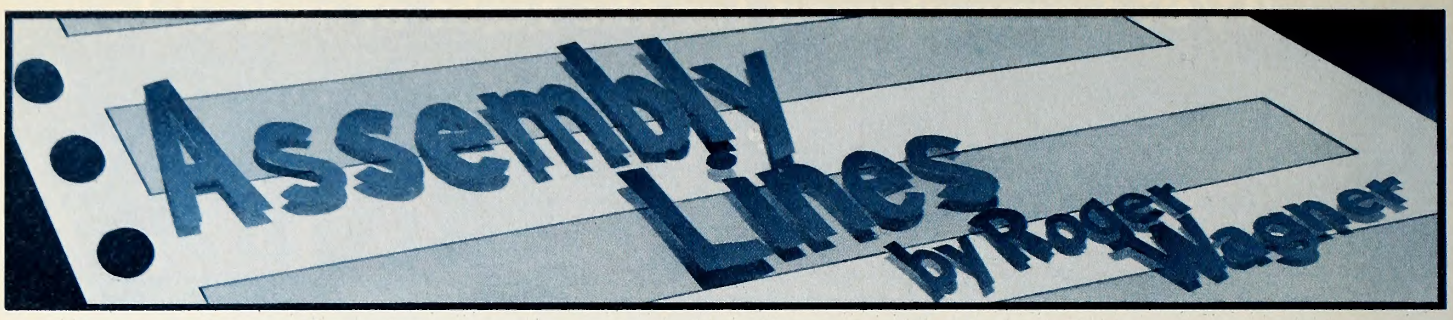
Pirates to your right, pirates to your left . . . poison snakes, spiders, scorpions and killer rats! Hear your sword ring as you parry, thrust, lunge . . . run them through! If you're swordsman enough to win below decks, you go topside, and the fight continues . . . with Blackbeard himself and the motley crew. What a game! What a test of skill against opponents who really attack and fight back! What a chance to become the swordsman of the year! Get your copy now - thrill to the most realistic HI-RES graphics ever!

SWASHBUCKLER. Only **$34.95** for the Apple II*. At computer stores or:

DATAMOST
9748 Cozycroft Ave.
Chatsworth, Ca. 91311
(213) 709-1202

VISA/MASTERCHARGE accepted.
$1.00 shipping/handling charge.
(California residents add 6% tax)

*Apple II a trademark of Apple Computer, Inc.



Everyone's Guide to Assembly Language, Part 19

Shortly before running out of space last month, we were discussing hi-res graphics and how to plot a bouncing hi-res ball. We had constructed a simple Applesoft program to illustrate the principles involved:

```
10  HGR
20  X=140:Y=80
30  XV=1:YV=1
100 REM DRAW LOOP
110 HCOLOR = 3:REM WHITE
120 HPLOT X,Y:REM DRAW OBJECT
130 HCOLOR = 0:REM BLACK
140 HPLOT X,Y:REM ERASE IT
200 REM MAKE IT MOVE!
210 X = X + XV:Y = Y + YV
220 IF X > 278 THEN XV = XV * (-1)
230 IF X < 1 THEN XV = XV * (-1)
240 IF Y > 158 THEN YV = YV * (-1)
250 IF Y < 1 THEN YV = YV * (-1)
260 GOTO 100
```

Note that this loop has a basic pattern of: draw -> erase -> calculate -> test -> (do it again . . .).

For the Applesoft program shown, this works fairly well, and is very understandable. There is one problem, however. That is the fact that very little time passes between the draw and erase stages, compared to the amount of time spent in the calculate and test sections. The result on the screen is a large amount of flicker, resulting from the dot spending more of its time black than white.

One solution to this is to make a small modification to the original Applesoft program, so that it appears as follows:

```
0   REM FP DOT DEMO PROGRAM
10  HGR
15  HCOLOR=3: HPLOT 0,0 TO 279,0 TO 279,159 TO 0,159 TO 0,0
20  X = 140: Y = 80
30  XV = 1 : YV = 1
100 REM CALC NEW POSN
110 TX = X + XV: TY = Y + XY

200 REM CHECK POSN
210 IF TX > 277 THEN XV = XV * (-1) : GOTO 110
220 IF TX < 2 THEN XV = XV * (-1) : GOTO 110
230 IF TY > 157 THEN YV = YV * (-1) : GOTO 110
240 IF TY < 2 THEN YV  = YV * (-1) : GOTO 110

300 REM ERASE OLD POSN
310 HCOLOR = 0: REM BLACK
320 HPLOT X,Y

400 REM DRAW NEW POSN
410 X = TX: Y = TY
420 HCOLOR = 3: REM WHITE
430 HPLOT X,Y
440 GOTO 100
```

This routine not only draws a nice border around the screen, but also follows this general pattern: calculate -> check -> erase old -> draw new -> (start over).

The advantage of this technique is that relatively little time is spent between the erase and redraw stages. Thus the dot is on the screen the majority of the time, and very little flicker is apparent.

Another new detail is the use of a set of temporary variables, TX and TY. These store the new position while the old one is being erased. The new one is then drawn and TX,TY are made "official" by being passed to the "real" X,Y variables.

As a minor point, also note that we have reduced the boundary test points in lines 200–240 so that the dot reverses direction before actually contacting the boundary we have drawn. Otherwise, the boundary would be erased by the dot passing through it on each bounce.

Now let's look at how to implement this program in machine language.

Signed Binary Numbers. A critical part in the Applesoft listing shown here is the ability to add a negative value to the speed factor of the object. So far in this series, we have only discussed how to represent whole numbers greater or equal to zero, using one or two bytes. A reasonable question then is, "How do we represent negative numbers?"

Negative numbers can be thought of as a way of handling certain common arithmetic possibilities, such as when subtracting a larger number from a smaller one, as in 3 − 5 = −2, and when adding a positive number to a negative number to obtain a given result, as in 5 + −8 = −3.

To be successful then, what we must come up with is a system using the eight bits in each byte that will be consistent with the signed arithmetic we are currently familiar with.

The Sign Bit. The most immediate solution to the question of signed numbers is to use bit 7 to indicate whether a number is positive or negative. If the bit is clear, the number is positive. If the bit is set, the number will be regarded as negative.

Thus positive 5 would be represented as 00000101, while −5 would be shown as 10000101.

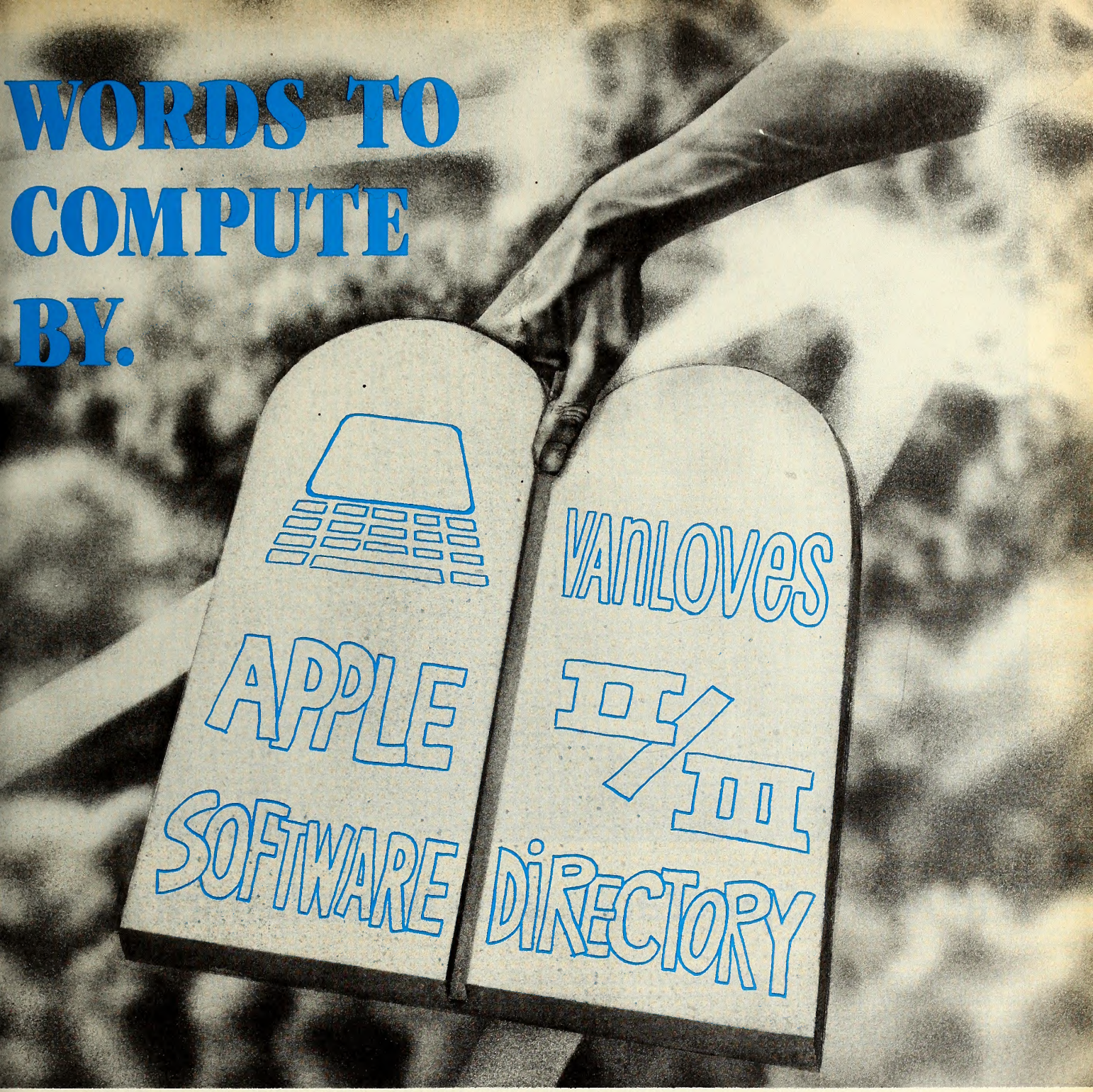
Note that by sacrificing bit 7 to show the sign, we've limited ourselves to values from −127 to +127. When using two bytes to represent a number—such as an address—we'll be limited to the range of −32,767 to +32,767. Sound familiar? If you've had any experience with Integer Basic, then you'll recognize this as the maximum range of number values within that language.

Although this new scheme is very pleasing in terms of simplicity, it does have one minor drawback—it doesn't work. If we attempt to add a positive and negative number using this scheme we get disturbing results:

```
    5        00000101
+  -8        10001000
-------     ----------
   -3        10001101 = -131
```

Although we should get −3 as the result, using our signed bit system we get −13. Tsk, tsk. There must be a better way. Well, with the help of what is essentially a little numeric magic, we can get something that works, although some of the conceptual simplicity gets lost in the process.

What we'll invoke is the idea of number *complements*. The



The complete word on Apple II and Apple III products comes from the folks at VanLoves. The 1982 issue will be available soon in a form less bulky than pictured above, with details on more current software and more current peripherals than you ever imagined were available for your Apple.

Here's a resource no Apple owner should be without. Hundreds of pages of brief descriptions on software and peripherals as well as a list of companies, a list of Apple clubs, and a list of books.

Added to this year's edition is a comprehensive index that will make using the directory even easier.

The plethora of products now available makes your Apple more exciting. Wouldn't it be heavenly to have VanLoves at hand to help make the best choices?

At computer stores and book stores everywhere, or order from:

VANLOVES

800-255-5119

Vital
Information Inc.

7899 Mastin Dr. Overland Park KS 66204

The Most Spectacular Extravaganza Ever..

At Applefest '82 hundreds of manufacturers, distributors and dealers will showcase the entire spectrum of Apple-compatible products including computers, components, peripherals, plug-in cards, publications, gifts, magazines, services, accessories and software for home, office and school.

Hands-on centers and multimedia presentations will demonstrate the newest applications for business, education and entertainment.

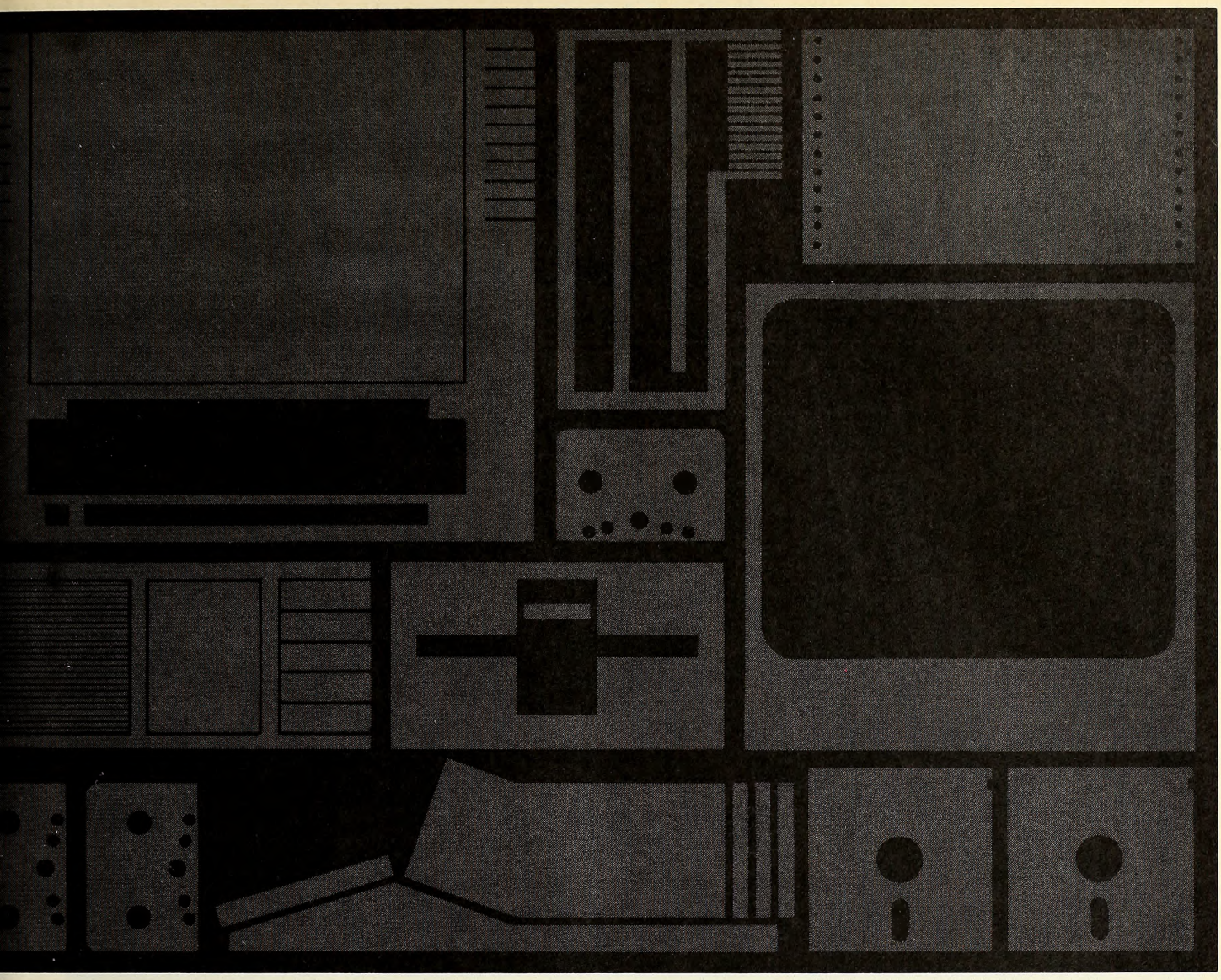
Seminars and workshops, conducted by the world's leading Apple authorities, will detail new uses to make your Apple more enjoyable and more useful than you ever imagined.

Applefest/Boston

Fri-Sun May 14-16, 1982
Hynes Auditorium
Show Hours: 11AM to 6PM Daily
Admission: $6 per day or $10 for 2 days,
$15 for 3 days

Applefest/Minneapolis

Thurs-Sun Sept 16-19, 1982
Minnesota Auditorium and Convention Hall
Show Hours: 11 AM to 6 PM Daily
Admission: $5 per day or $8 for 2 days,
$12 for 3 days, $15 for 4 days



For Apple Users

You'll meet thousands of other Apple owners and find the newest of everything for your Apple under one roof . . . and for sale at super show prices.

So if you use an Apple . . . or are thinking about buying one, you won't want to miss a minute of Applefest '82.

Ticket & Hotel Information
Send your check and a note indicating the specific show you wish to attend. Tickets and hotel information will be mailed back to you. Tickets can also be purchased at the show. Make all checks payable to Northeast Expositions Inc. 824 Boylston Street, Chestnut Hill, Mass. 02167 Tel: 617 739 2000.

Exhibitor Information
For specific exhibitor information on one or all of the Applefest '82 shows call Northeast Expositions at the telephone number above.

Applefest/Houston

Fri-Sun Nov 19-21, 1982
Albert Thomas Convention Center
Show Hours: 1PM to 10PM Daily
Admission: $5 per day or $8 for 2 days, $12 for 3 days

Applefest/San Francisco

Fri-Sun Dec 3-5, 1982
Moscone Center
Show Hours: 1PM to 10PM Daily
Admission: $5 per day or $8 for 2 days, $12 for 3 days

Applefest is produced by Northeast Expositions Inc. and is sanctioned by Apple Computer Inc. and The Boston Computer Society.

*Apple and Applefest are registered trade and service marks of Apple Computer Inc.

Professional proofreading on the APPLE II Computer is now a reality

THE APPLE SPELLER fills the void that has consistently kept the large variety of excellent word processing packages for the Apple II Computer from approaching the power of a dedicated work processor. Finally, the first professional quality spelling verification program is available for the Apple II. The Apple Speller will certainly be the standard against which all other similar programs are compared.

The Apple Speller interfaces to the most popular Apple word processors, including Applewriter, Apple Pie, Executive Secretary, Letter Perfect, Magic Window, and Superscribe II, just to name a few. In fact, The Apple Speller can analyze the output of any editor that writes a standard Apple binary or text file to a diskette. In addition to this flexibility, the performance of The Apple Speller will astound the microcomputer world.

The Apple Speller is supplied with a 30,000+ word dictionary on a single 5¼" diskette with additional space to easily add another 8,000 words to suit your individual needs.

The Apple Speller has built-in utilities to maintain the dictionary diskette. You can readily add words, delete words, and create an unlimited number of modified and/or new dictionaries for specific application.

The Apple Speller is unbelievably fast. The first pass reads your document and collects all the words it contains at a rate of 5,000 words per minute. Next, the words are compared to the dictionary for spelling errors at the incredible speed of 50,000 words per minute. Finally, all misspelled words are marked as such in your document with a rate of 1,000 words per minute. This translates to proofreading a 10 page document in 1 minute if there are no spelling mistakes and 2 minutes, 15 seconds for an unlimited number of spelling errors.

Numerous options are provided throughout the program to enable you to completely control all activities of The Apple Speller. These include the ability to ignore both control codes and formatting commands, an alphabetical listing of either mispelled words or all the words in your document along with usage frequencies, multiple options for the action taken with each misspelled word, and much, much more! A verification mode is provided to allow you to examine and dispense with misspelled words while viewing them in the actual context in which they appeared in your file.

The Apple Speller requires an Apple II/Apple II+ equipped with 48K, DOS 3.3, and 1 or 2 disk drives. Two disk drives are required to delete or add words to the dictionary.

The Apple Speller is being introduced at the incredible price of **$75.00!**

Sensible Software
6619 Perham Drive Dept. M
West Bloomfield, Michigan 48033
(313) 399-8877
Visa and Mastercard Welcome
Please add $1.25 postage and handling per diskette
Please specify program desired

simplest complement is what is called a *one's complement.* The one's complement of a number is obtained by reversing each 1 and 0 throughout the original binary number.

For example, the one's complement to 5 would be:

```
00000101 =  5
11111010 = -5
```

For 8, it would be:

```
00001000 =  8
11110111 = -8
```

This process is essentially one of definition, that is to say that we declare to the world that 11110111 will now represent −8 without specifically trying to justify it. Undoubtedly there are lovely mathematical proofs of such things that present marvelous ways of spending an afternoon, but for our purpose, a general notion of what the terms mean will be sufficient. Fortunately computers are very good at following arbitrary numbering schemes without asking, "But why is it that way?"

Now let's see if we're any closer to a working system:

```
   5          00000101
+ -8          11110111
------        --------
  -3          11111100 = -3
            (00000011 = +3)
```

Hmmm, . . . Seems to work pretty good. Let's try another:

```
  -5          11111010
+  8          00001000
------        --------
   3          00000010 =  2
              (plus carry)
```

Well our answers will be right half the time. . . . It turns out there is a final solution, and that is to use what is called the *two's complement* system.

The only difference between this and the one's complement system is that after deriving the negative number by reversing each bit of its corresponding positive number, *we add one.*

Sounds mysterious. Let's see how it looks:

```
For −5:                      For −8:
5 = 00000101                 8 = 00001000

                one's complement

11111010                          11110111

                  now add 1

 −5 = 11111011                −8 = 11111000
```

Now let's try the two earlier operations:

```
   5          00000101
+ -8          11111000
------        --------
  -3          11111101 = -3
sample number: (00000011 =  3)
one's complement: (11111100    )
                  (     +1     )
                  ---------------
two's complement: (11111101 = -3)
  -5          11111011
+  8          00001000
------        --------
   3          00000011 =  3
              plus carry
```

At last! It works in both cases. It turns out that two's complement math works in all cases, with the carry being ignored on one-byte additions.

Taking the Opposite of a Signed Number. All that we need now is a routine that will produce the opposite of a number given it, that is produce the two's complement of a positive number and also the positive version when given a negative value. To do this, we'll use the EOR command.

EOR is useful in creating a routine to convert between signed numbers because of its ability to reverse all the bits in a given byte. The conversion is done with two individual routines. In the examples below, the routines convert a constant value, #$34, back and forth. In a working version, the value would be passed via a register or a memory location, as will be shown later.

```
                        Positive to Negative
ENTRY   LDA   #$34  ;  %00110100 = +52
                    ;  TO BE CVRTD TO −52
        EOR   #$FF  ;  %11111111 TO RVRS BITS
                    ;  RSLT = %11001011
        CLC
        ADC   #$01  ;  RSLT = RSLT + 1
                    ;       = %11001100 = $CC
        STA   MEM   ;  STORE RSLT
DONE    RTS
                        Negative to Positive
ENTRY   LDA   #$CC  ;  %11001100 = $CC = −52
                    ;  TO BE CVRTD BACK
        SEC
        SBC   #$01  ;  ACC = ACC − 1
                    ;       = %11001011 = $CB
        EOR   #$FF  ;  REVERSE ALL BITS
                    ;  RSLT = %00110100 = $34 = +52
        STA   MEM   ;  STORE RSLT
DONE    RTS
```

Note that in this example the percent sign is used to indicate the binary form of the number. Some assemblers support this notation.

The Real Thing. We now have the necessary tools to construct the machine language version of the last Applesoft listing. Assemble and run this listing:

```
                        1   **********************
                        2   * HIRES ONE DOT PROG *
                        3   **********************
                        4   *
                        5   *
                        6           OBJ   $6000
                        7           ORG   $6000
                        8   *
                        9   X       EQU   $E0      ; $E0,E1
                       10   Y       EQU   $E2
                       11   XV      EQU   $06      ; $06,07
                       12   YV      EQU   $08
                       13   TX      EQU   $09      ; $09,0A
                       14   TY      EQU   $0B
                       15   *
                       16   PREAD   EQU   $FB1E
                       17   WAIT    EQU   $FCA8
                       18   HCOLOR  EQU   $F6F0
                       19   HGR     EQU   $F3E2
                       20   HPLOT   EQU   $F457
                       21   HPOSN   EQU   $F411
                       22   HLIN    EQU   $F53A
                       23   *
6000: 20  E2  F3       24   ENTRY   JSR   HGR
6003: A2  03           25           LDX   #$03     ; WHITE
6005: 20  F0  F6       26           JSR   HCOLOR
                       27   *
6008: A9  00           28   BOX     LDA   #$00     ; Y = 0
600A: A8               29           TAY
600B: AA               30           TAX
600C: 20  57  F4       31           JSR   HPLOT    ; PLOT 0,0
600F: A9  17           32           LDA   #23      ; 279 MOD 256
6011: A2  01           33           LDX   #01      ; 279/256
```

```
6013: 20 3A F5 34          JSR  HLIN   ; FROM 0,0 TO 279,0
                35 *
6016: A9 17       36       LDA  #23
6018: A2 01       37       LDX  #01
601A: A0 9F       38       LDY  #$9F   ; Y = 159
601C: 20 3A F5 39          JSR  HLIN   ; 279,0 TO 279,159
                40 *
601F: A9 00       41       LDA  #$00
6021: A2 00       42       LDX  #$00
6023: A0 9F       43       LDY  #$9F
6025: 20 3A F5 44          JSR  HLIN   ; 279,159 TO 0,159
                45 *
6028: A9 00       46       LDA  #$00
602A: A2 00       47       LDX  #$00
602C: A0 00       48       LDY  #$00
602E: 20 3A F5 49          JSR  HLIN   ; 0,159 TO 0,0
                50 *
6031: A9 00       51 SET   LDA  #$00
6033: 85 07       52       STA  XV+1
6035: A9 01       53       LDA  #$01
6037: 85 06       54       STA  XV     ; XV = 1
6039: 85 08       55       STA  YV     ; YV = 1
                56 *
6038: A2 8C       57 POSN  LDX  #$8C
603D: A0 00       58       LDY  #$00   ; X = 140
603F: A9 50       59       LDA  #$50   ; Y = 80
6041: 20 11 F4 60          JSR  HPOSN  ; SET CURSOR @ X,Y
                61 *
6044: 18          62 CALC  CLC
6045: A5 E0       63       LDA  X
6047: 65 06       64       ADC  XV
6049: 85 09       65       STA  TX
6048: A5 E1       66       LDA  X+1
604D: 65 07       67       ADC  XV+1
604F: 85 0A       68       STA  TX+1   ; TX = X + XV
                69 *
6051: 18          70       CLC
6052: A5 E2       71       LDA  Y
6054: 65 08       72       ADC  YV
6056: 85 08       73       STA  TY     ; TY = Y + YV
                74 *
6058: A5 0A       75 CHK   LDA  TX+1
605A: D0 09       76       BNE  CHK2
605C: A5 09       77       LDA  TX
605E: C9 02       78       CMP  #$02
6060: 80 03       79       BCS  CHK2
6062: 20 AE 60 80          JSR  RVRSX  ; X < 2
                81 *
6065: A5 0A       82 CHK2  LDA  TX+1
6067: C9 01       83       CMP  #$01
6069: 90 09       84       BCC  CHK3
606B: A5 09       85       LDA  TX
606D: C9 16       86       CMP  #$16
606F: 90 03       87       BCC  CHK3
6071: 20 AE 60 88          JSR  RVRSX  ; TX >= $116 (278)
                89 *
6074: A5 08       90 CHK3  LDA  TY
6076: C9 02       91       CMP  #$02
6078: 80 03       92       BCS  CHK4
607A: 20 D6 60 93          JSR  RVRSY  ; TY < 2
                94 *
607D: A5 08       95 CHK4  LDA  TY
607F: C9 9E       96       CMP  #$9E
6081: 90 03       97       BCC  ERASE
6083: 20 D6 60 98          JSR  RVRSY  ; TY >= $9E (158)
                99 *
6086: A2 00      100 ERASE LDX  #$00   ; BLACK = 0
6088: 20 F0 F6 101         JSR  HCOLOR
6088: A6 E0      102       LDX  X
608D: A4 E1      103       LDY  X+1    ; GET X,X+1
608F: A5 E2      104       LDA  Y      ; GET Y
6091: 20 57 F4 105         JSR  HPLOT  ; ERASE POINT
               106 *
6094: A2 03      107 PLOT  LDX  #$03   ; WHITE1=3
6096: 20 F0 F6 108         JSR  HCOLOR
6099: A6 09      109       LDX  TX
609B: A4 0A      110       LDY  TX+1   ; GET TX,TX+1
609D: A5 08      111       LDA  TY     ; GET TY
609F: 20 57 F4 112         JSR  HPLOT  ; PLOT POINT
               113 *
60A2: A2 00      114 DELAY LDX  #$00   ; PDLO
60A4: 20 1E FB 115         JSR  PREAD
60A7: 98         116       TYA
60A8: 20 A8 FC 117         JSR  WAIT
               118 *
               119 *
60A8: 4C 44 60 120 GO8ACK  JMP  CALC
               121 *
               122 *
60AE: A5 07      123 RVRSX LDA  XV+1
60B0: 30 12      124       BMI  NEG/POSX
               125 *
60B2: A5 06      126 POS/NEGX LDA  XV
6084: 49 FF      127       EOR  #$FF
60B6: 18         128       CLC
60B7: 69 01      129       ADC  #$01
6089: 85 06      130       STA  XV
60BB: A5 07      131       LDA  XV+1
60BD: 49 FF      132       EOR  #$FF
608F: 69 00      133       ADC  #$00
60C1: 85 07      134       STA  XV+1
60C3: 60         135       RTS         ; XV -> -XV
               136 *
60C4: A5 06      137 NEG/POSX LDA  XV
60C6: 38         138       SEC
60C7: E9 01      139       SBC  #$01
60C9: 49 FF      140       EOR  #$FF
60CB: 85 06      141       STA  XV
60CD: A5 07      142       LDA  XV+1
60CF: E9 00      143       SBC  #$00
60D1: 49 FF      144       EOR  #$FF
60D3: 85 07      145       STA  XV+1
60D5: 60         146 DONEX RTS         ; -XV -> XV
               147 *
               148 *
               149 *
60D6: A5 08      150 RVRSY LDA  YV
60D8: 30 0A      151       BMI  NEG/POSY
               152 *
60DA: A5 08      153 POS/NEGY LDA  YV
60DC: 49 FF      154       EOR  #$FF
60DE: 18         155       CLC
60DF: 69 01      156       ADC  #$01
60E1: 85 08      157       STA  YV
60E3: 60         158       RTS         ; YV -> -YV
               159 *
60E4: A5 08      160 NEG/POSY LDA  YV
60E6: 38         161       SEC
60E7: E9 01      162       SBC  #$01
60E9: 49 FF      163       EOR  #$FF
60EB: 85 08      164       STA  YV
60ED: 60         165 DONEY RTS         ; -YV -> YV
               166 *
```

When you run this routine, notice how much faster it executes and how the speed of the dot can be varied using paddle 0.

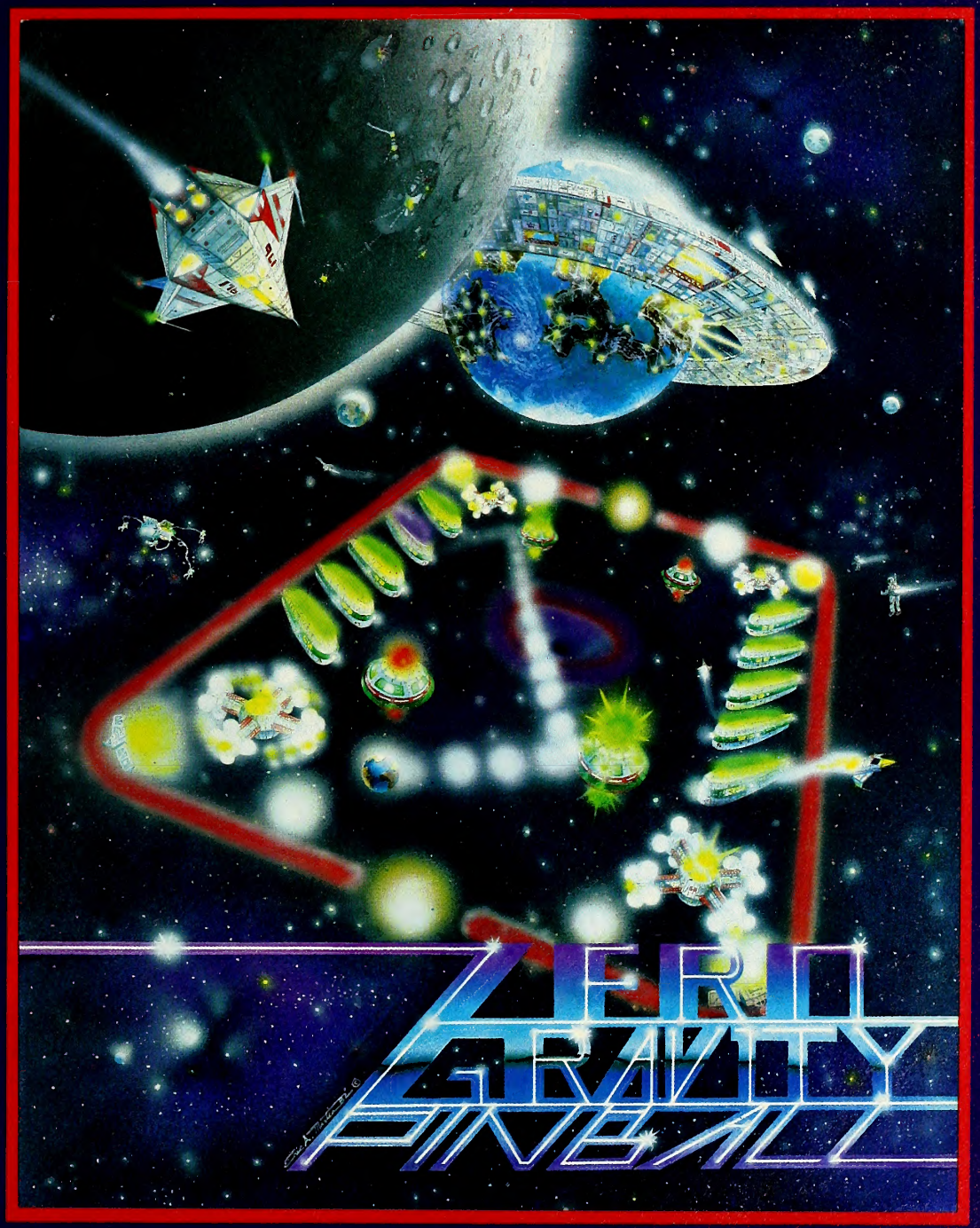
This routine essentially parallels the Applesoft routine shown earlier. Lines 24-50 clear the hi-res screen and draw the box around the screen. Lines 51-55 set the velocity components to '1'; lines 57-61 position the hi-res cursor in the center of the screen. This also conveniently loads $E0-E2 with the desired x and y coordinates of the dot. Remember that $E0/E1 and $E2 are the zero page locations used by the Applesoft hi-res routines for the x and y coordinates of its cursor.

Lines 62-73 calculate the new position of the dot by adding the respective velocity components to the x and y coordinates. Lines 75-98 check to see if this new position is still within the specified screen boundaries. If it has reached the edge, the appropriate velocity components are reversed for the next go-round's calculation.

Line 100 starts the erasing of the current dot position, immediately followed by a drawing of the new position. Note that the equivalent of the X=TX:Y=TY statement is apparently missing. In actuality, it is automatically accomplished by the JSR HPLOT on line 112. Remember that the contents of the Accumulator, X and Y registers are automatically assigned to $E0-E2 by HPLOT. Line 114 does a short delay by getting a value from paddle 0 to be used by the WAIT ($FCA8) routine. After the delay, a JMP CALC restarts the entire process.

PINBALL HAS GONE INTO ORBIT!

Now that you've conquered pool hall pinball, it's time to take your Apple into outer space*



ZERO GRAVITY PINBALL

A machine language arcade game with full-color hi-res graphics, and sounds like you've never heard before.

ZERO GRAVITY PINBALL **$29.95** plus postage

Requires game paddle 0, Apple II*, with Applesoft* ROM, 48K, DOS 3.2 or 3.3

ask your dealer or order direct

AVANT-GARDE CREATIONS
P.O. Box 30160
Eugene, OR 97403
(503) 345-3043

VISA/MasterCard accepted

*Apple II, Applesoft are registered trademarks of Apple Computer, Inc.

MONEY BUSINESS

Beat the I.R.S.

You'd always pay the minimum Federal income tax if you had the knowledge and time to examine 120 tax return options. Well, now you can do it, with Datamost's TAX BEATER.

Written by an I.R.S. Enrolled Agent, with 24 years of financial and tax planning experience, the TAX BEATER automatically evaluates up to 120 return options from your input. It searches and finds the best tax path for you . . . displaying up to 15, and ready to print out the optimum method so you pay the minimum!

The TAX BEATER can handle it all. From income averaging to loss carryovers to dividends, alimony, pension to special exclusions. It's perfect for the average taxpayer . . . indispensable for the financial professional. It's so complete, so logical that, especially in this interim year of unusual tax revisions, it can easily pay for itself many times over. Be ready for the I.R.S. . . . with TAX BEATER, the program that's updated for the 1981 tax laws!

$129.95 on disk for Apple II*. Complete with thorough, easy to follow documentation.

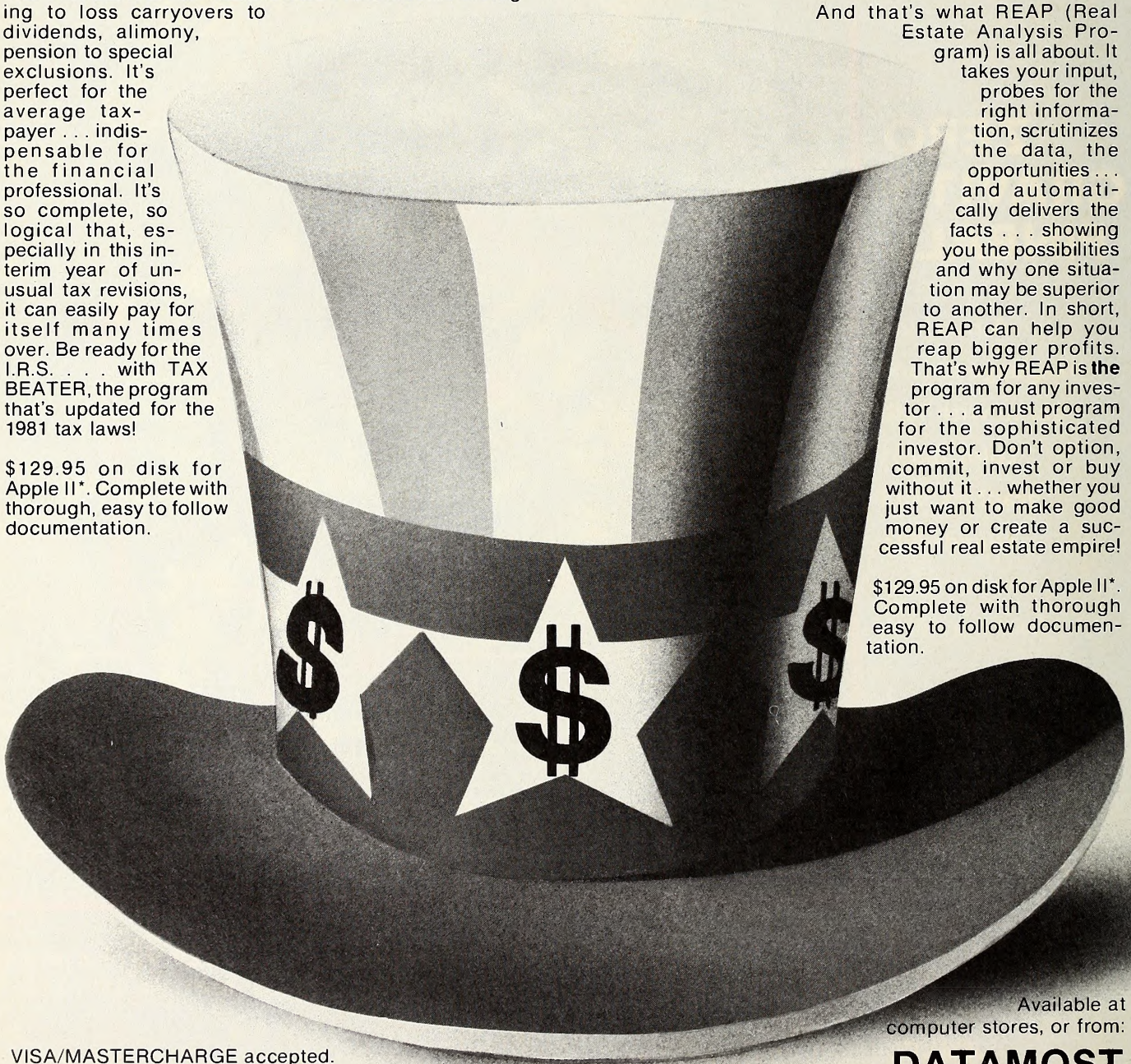
Beat the Real Estate game

Real Estate has always been the major wealth-builder. In fact, more millionaire dynasties owe their creation to investments in real estate than anything else. And, it doesn't matter . . . boom times or bust . . . inflation or deflation . . . the knowing investor can make money, even create an empire!

But, the key to real estate success is accurate analysis of opportunities, knowledge of cash flows, return on investment, current and future profitability, tax consequences and other interrelated information — all needed, all necessary to making the right decision.

And that's what REAP (Real Estate Analysis Program) is all about. It takes your input, probes for the right information, scrutinizes the data, the opportunities . . . and automatically delivers the facts . . . showing you the possibilities and why one situation may be superior to another. In short, REAP can help you reap bigger profits. That's why REAP is **the** program for any investor . . . a must program for the sophisticated investor. Don't option, commit, invest or buy without it . . . whether you just want to make good money or create a successful real estate empire!

$129.95 on disk for Apple II*. Complete with thorough easy to follow documentation.



VISA/MASTERCHARGE accepted.
$1.00 shipping/handling charge.
(California residents add 6% tax)

Available at computer stores, or from:

DATAMOST
19273 Kenya St.
Northridge, Ca. 91326
(213) 366-7160

* Apple II is a trademark of Apple Computer, Inc.

Lines 126-165 are applications of the sign reversal routines shown last month. Notice that RVRSY is a one-byte reversal, while RVRSX illustrates the reversal of a two-byte value. Similarly, CALC shows that the same addition routine is used for both signed binary (our current condition) and unsigned binary (as in previous articles).

Table-Driven Graphics. For graphics of any complexity—anything involving more than one dot—a little improvement on this routine is needed. One of the most common ways of doing this is to use a table of all the current points on the screen, and their corresponding velocities. Motion is then managed by sequentially scanning through the table and using the entire calculation, check and erase/plot section as a subroutine.

To convert the routine presented earlier, make the following changes to the source code (the hex data from the assembly is included to assist in error checking):

1) Add these lines to the end of the listing (new line numbers shown):

```
6154:  A2 00      241  SETUP  LDX  #$00
6156:  BD 62 61   242  LOOP   LDA  DATA,X
6159:  9D 00 10   243         STA  TABLE,X
615C:  E8         244         INX
615D:  E0 28      245         CPX  #40      ; 8 BYTES * NUM DOTS
615F:  90 F5      246         BCC  LOOP
6161:  60         247  DONE   RTS
                  248  *
6162:  8C 00 50   249  DATA   HEX  8C005000  ; X,Y(1) = 8C,50
6166:  01 00 01   250         HEX  01000100  ; XV,YV(1) = 1,1
                  251  *
606A:  8E 00 52   252         HEX  8E005200  ; X,Y(2) = 8E,52
606E:  01 00 01   253         HEX  01000100  ; XV,YV(2) = 1,1
                  254  *
6072:  90 00 54   255         HEX  90005400  ; X,Y(3) = 90,54
6076:  01 00 01   256         HEX  01000100  ; XV,YV(3) = 1,1
                  257  *
607A:  92 00 56   258         HEX  92005600  ; X,Y(4) = 92,56
607E:  01 00 01   259         HEX  01000100  ; XV,YV(4) = 1,1
                  260  *
6082:  94 00 58   261         HEX  94005800  ; X,Y(5) = 94,58
6086:  01 00 01   262         HEX  01000100  ; XV,YV(5) = 1,1
                  263  *
```

2) Rewrite line 120 (will end up as 190) as:

```
6113:  60         190  GOBACK  RTS
```

3) Rewrite the beginning of the source as:

```
                    1  ********************
                    2  * HIRES LOTS DOTS *
                    3  ********************
                    4  *
                    5  *
                    6          OBJ   $6000
                    7          ORG   $6000
                    8  *
                    9  TABLE   EQU   $1000
                   10  CTR     EQU   $0C
                   11  NUM     EQU   $05      ; FIVE DOTS
                   12  *
                   13  X       EQU   $E0      ; $E0,E1
                   14  Y       EQU   $E2
                   15  XV      EQU   $06      ; $06,07
                   16  YV      EQU   $08
                   17  TX      EQU   $09      ; $09,0A
                   18  TY      EQU   $0B
                   19  *
                   20  PREAD   EQU   $FB1E
                   21  WAIT    EQU   $FCA8
                   22  HCOLOR  EQU   $F6F0
                   23  HGR     EQU   $F3E2
                   24  HPLOT   EQU   $F457
                   25  HPOSN   EQU   $F411
                   26  HLIN    EQU   $F53A
                   27  *
6000:  20 E2 F3    28  ENTRY   JSR   HGR
6003:  A2 03       29          LDX   #$03     ; WHITE
6005:  20 F0 F6    30          JSR   HCOLOR
                   31  *
6008:  20 54 61    32  TABLESET JSR  SETUP
                   33  *
600B:  A9 00       34  BOX     LDA   #$00     ; Y = 0
600D:  A8          35          TAY
600E:  AA          36          TAX
600F:  20 57 F4    37          JSR   HPLOT    ; PLOT 0,0
                    "
                    "
                    "
```

4) Insert the code for the table look-up starting at new line 68:

```
6047:  A9 00       68  LOOKUP  LDA  #$00
6049:  85 0C       69          STA  CTR
604B:  A5 0C       70  GET     LDA  CTR
604D:  0A          71          ASL
604E:  0A          72          ASL
604F:  0A          73          ASL        ; X = CTR*8
                   74  *
6050:  AA          75          TAX
6051:  BD 00 10    76          LDA  TABLE,X
6054:  85 E0       77          STA  X
6056:  E8          78          INX
6057:  BD 00 10    79          LDA  TABLE,X
605A:  85 E1       80          STA  X+1
605C:  E8          81          INX
605D:  BD 00 10    82          LDA  TABLE,X
6060:  85 E2       83          STA  Y
6062:  E8          84          INX
6063:  E8          85          INX        ; Y+1 NOT USED
                   86  *
6064:  BD 00 10    87          LDA  TABLE,X
6067:  85 06       88          STA  XV
6069:  E8          89          INX
606A:  BD 00 10    90          LDA  TABLE,X
606D:  85 07       91          STA  XV+1
```

```
606F:  E8              92          INX
6070:  BD  00  10      93          LDA   TABLE,X
6073:  85  08          94          STA   YV
                       95    *
6075:  20  AC  60      96    SERVICE JSR   CALC
                       97    *
6078:  A5  0C          98    PUT   LDA   CTR
607A:  0A              99          ASL
607B:  0A              100         ASL
607C:  0A              101         ASL
607D:  AA              102         TAX
                       103   *
607E:  A5  E0          104         LDA   X
6080:  9D  00  10      105         STA   TABLE,X
6083:  E8              106         INX
6084:  A5  E1          107         LDA   X+1
6086:  9D  00  10      108         STA   TABLE,X
6089:  E8              109         INX
608A:  A5  E2          110         LDA   Y
608C:  9D  00  10      111         STA   TABLE,X
608F:  E8              112         INX
6090:  E8              113         INX   ; SKIP BYTE
                       114   *
6091:  A5  06          115         LDA   XV
6093:  9D  00  10      116         STA   TABLE,X
6096:  E8              117         INX
6097:  A5  07          118         LDA   XV+1
6099:  9D  00  10      119         STA   TABLE,X
609C:  E8              120         INX
609D:  A5  08          121         LDA   YV
609F:  9D  00  10      122         STA   TABLE,X
                       123   *
60A2:  E6  0C          124         INC   CTR
60A4:  A5  0C          125         LDA   CTR
60A6:  C9  05          126         CMP   #NUM    ; NUMBER OF DOTS
60A8:  90  A1          127         BCC   GET
60AA:  B0  9B          128         BCS   LOOKUP
                       129   *
                       130   *
                       131   *
60AC:  18              132   CALC  CLC
60AD:  A5  E0          133         LDA   X
60AF:  65  06          134         ADC   XV
                       "
                       "
                       "
```

Run this routine from the Monitor with a 6000G or from Applesoft with a *call* 24576. If calling from the Monitor, make sure you have entered the Monitor from Applesoft when you do the *call −151* to assure that the Applesoft ROM or RAM card bank is selected. Note that although the entire routine is in machine language, it does require the presence of the Applesoft hi-res routines in the $D000-F7FF range.

By using paddle 0 you can vary the speed of execution considerably. One drawback of using the WAIT routine is that 0 will be just as slow as 255 when adjusting the paddle. Otherwise, it should behave quite nicely.

To speed things up further, NOP out the JSR to WAIT on line 187. An even greater speed increase is achieved by similarly disabling the JSR PREAD on line 185, although with PREAD gone there is no longer any control over the speed. This will, however, give you an idea of the maximum speed possible for the five dots using standard Applesoft hi-res routines.

The main points to note in the new listing are the JSR to SETUP on line 32, the LOOKUP section on lines 68-128, and the table generator at the end on lines 241-263.

SETUP creates a data table starting at location $1000 that contains a number of eight-byte blocks, each of which contains the necessary information for a given dot. The block is made up of two four-byte subunits. The first four bytes give the location data for the x and y coordinates. Notice that the fourth byte is not used. Space in the table could have been saved by omitting this byte, but the eight byte length per entry allows us to use a few simple ASLs, as will be explained momentarily.

GOTO 169

You don't have to change
the way you do business,
the Data Factory adapts to you.

Do it your way!

The new version of the Data Factory is COMPLETELY ADAPTABLE. It includes a customized data entry system that allows you to position information anywhere on the screen. The flexible printout routine frees you from formatting constraints giving you pinpoint control.

A new Master Search routine will locate records by any criteria you wish . . . from one value to another, date-to-date, by the beginning of a field, a complete field, or within it, greater than, less than, equal to, not equal to, and any combination of the above. The sequence can be stored for future use.

The new sort feature flashes through 1,000 records in less than six seconds. Work with two, on-line disks giving you 225,000 bytes of data storage capacity. The expanded math routine has the most advanced math capabilities for an Apple data base, use it across all fields and store the answers on data disk. You may still add, delete, or change fields without re-entering information. Change up to 10,000 fields of data at one time.

Anyone can use the power and flexibility of the new Data Factory. Prompts guide you throughout. It's difficult to imagine what else anyone would need. The package includes two master disks and two back-ups. A one-year Extended Warranty is included in the price of the program that replaces blown disks and provides updated versions. Available now at $300. See it at your Micro Lab dealer, today. You'll need an Apple II-48K.

Another fine product
from the people who
brought you:

Invoice Factory
The Merger
VisiFactory
Language Plus
Tax Manager
Asset Manager
The Learning System
and many more.

micro lab
systems that work
2310 Skokie Valley Road,
Highland Park, Il. 312-433-7550

THE DATA FACTORY T.M.

SWORD THRUST™

All await you in SwordThrust™

by Donald Brown.

The King's Testing Ground (Master diskette) $29.95
All other adventures $29.95

Available now at your local computer store or from

CE Software
801 73rd St.
Des Moines, Iowa 50312
(515) 224-1995

1. The King's Testing Ground
2. The Vampyre Caves
3. Kidnapper's Cove
4. The Sultan's Pearl
5. The Green Plague
6. The Eternal Curse
7. The Hall of Alchemie

SwordThrust requires a 48K Apple II+® or Apple III® with Applesoft in ROM and one disk drive.

The Most Popular Program of 1981: RASTER BLASTER!

In a landslide vote that overwhelmed the other competitors, much as *VisiCalc* stands in a class by itself each month on the Top Thirty, *Raster Blaster* was elected by *Softalk*'s readers as the most popular software program for the Apple computer for 1981.

Anyone who didn't think Bill Budge's *piece de resistance* would place high hasn't been paying attention to the Apple market of late. But the margin of victory startled even the poll monitors, who halfway through the tabulation were predicting a head-to-head battle between *Apple Panic* and *Olympic Decathlon.*

While the monitors weren't looking, *Castle Wolfenstein* snuck into second place with *Apple Panic* third and *Olympic Decathlon* fourth.

That *Raster Blaster* is the overwhelming choice of Apple users is indicated by the fact that it was mentioned on 113 more ballots than was *Apple Panic,* which received the second most raw votes.

Votes were tabulated on a weighted basis. Each elector could name as many as ten programs in order. The votes were then weighted ten points for a first place mention, nine for a second place nod, and so on down to one point for a tenth place vote.

In the matter of the number of raw votes received, *Raster Blaster* and *Apple Panic* were trailed by *Gorgon,* the fifth place finisher overall; *Alien Rain,* the sixth place program; *Space Eggs,* the ninth place disk, and *Olympic Decathlon* tied; *Castle Wolfenstein;* and *Snoggle.*

Programs generally getting their first big marketing push during November and December of 1980 and all of 1981 were eligible for election. Questionable areas were resolved in favor of the program, so that *DB Master, Personal Filing System,* and *Magic Window*—all released prior to November 1980, but not truly in general release—were admitted to the rolls, while such programs as *VisiCalc, VisiFile, Supertext II,* and *Data Factory,* which underwent extensive revisions but had

much longer marketing postures, were omitted from final consideration.

Even though *VisiCalc* was excluded from the list of eligible programs, it garnered sufficient mentions that it would have placed sixteenth, and the probability that it would have contended seriously with *Raster Blaster* for first place had Apple users been free to include it is indicated by examining the weighted averages.

The weighted average indicates the strength of feeling of the user base for a particular program. Clearly 10 was the highest possible score, with anything over 7 indicating that the program significantly enhanced the library of the user.

In the top one hundred programs, only fourteen managed averages higher than 7, and only three of those were games. Weighted averages can be looked at as an indication of most valuable, although the fact that many of the most useful programs were not eligible for this year's balloting puts such assumptions on shaky terrain.

The surprising leader in weighted averages was High Technology's *Information Master,* which snared an 8.88 rating. Following were *Wizardry,* its 8.49 rating testifying to the way it's catching on throughout the country; *Executive Secretary,* 8.38; *Nibbles Away,* 8.25; *Data Factory,* 7.91; *WordStar,* 7.86; *Z-Term,* 7.78; *Zork II,* 7.73; *VisiCalc,* 7.73; *DB Master,* 7.37; *Magic Window,* 7.34; *Castle Wolfenstein,* 7.29; *Locksmith,* 7.27; and *VisiFile,* 7.00.

In contrast, *Raster Blaster* obtained a weighted average of 6.21.

The results are tabulated in the same format as the Best-sellers, with breakouts for each category customarily treated individually in that section.

In addition, there's a list of all programs that received first place ballots. This list may serve as a priority shopping guide, in that at least one member of the Apple community recommends it highly.

Individual programming honors, Budge's vote count not-

Strategy 5

1. **Castle Wolfenstein,** Silas Warner, Muse
2. **Robot War,** Silas Warner, Muse
3. **Warp Factor,** Paul Murray and Bruce D. Clayton, Strategic Simulations
4. **Hi-Res Football,** Jay Sullivan and Ken Williams, On-Line Systems
5. **Computer Baseball,** Charles Merrow and Jack T. Avery, Strategic Simulations

Adventure 5

1. **Zork,** Infocom
2. **Hi-Res Adventure #3: Cranston Manor,** Harold DeWitz and Ken Williams, On-Line Systems
3. **The Prisoner,** David Mullich, Edu-Ware Services
4. **Zork II,** Infocom
5. **Hi-Res Adventure #0: Mission: Asteroid,** Roberta and Ken Williams, On-Line Systems

Fantasy 5

1. **Wizardry,** Andrew Greenberg and Robert Woodhead, Sir-tech
2. **Ultima,** Lord British, California Pacific
3. **Hellfire Warrior,** Automated Simulations
4. **Akalabeth,** Lord British, California Pacific
5. **Dragon Fire,** Rodney Nelson, Dakin5/Level-10

Business 10

1. **DB Master,** Alpine Software/Stanley Crane and Jerry Macon; and Barney Stone, Stoneware
2. **Personal Filing System,** John Page, Software Publishing Corporation
3. **VisiTrend/VisiPlot,** Micro Finance Systems/Mitch Kapor, Personal Software
4. **BPI General Ledger,** John Moss and Ken Debower, Apple Computer
5. **VisiDex,** Peter Jennings, Personal Software
6. **VisiPlot,** Micro Finance Systems/Mitch Kapor, Personal Software
7. **BPI Accounts Receivable,** John Moss and Ken Debower, Apple Computer
8. **Data Reporter,** Robert Clardy, Christopher Anson, and Michael Branham, Synergistic Software
9. **PFS:Report,** John Page, Software Publishing Corporation
10. **Datadex,** Information Unlimited Software

withstanding, must go to Silas Warner, Muse's prodigious programmer. Not only did he program runnerup *Castle Wolfenstein,* he had the twentieth and twenty-fourth rated programs in *Robot War* and *ABM.*

Nasir placed *Gorgon* fifth and *Space Eggs* ninth. Ken Williams shared programming honors on *Threshold,* ranked twenty-seventh, and *Hi-Res Adventure #3: Cranston Manor,* ranked twenty-ninth.

On-Line Systems placed the most programs in the Top Thirty—five—but all were in the bottom ten. Sirius Software had four programs in the Top Thirty, Nasir's two and *Sneakers* achieving the top ten. Broderbund and Muse each placed three programs in the Top Thirty and Apple itself had two programs.

In addition to *VisiCalc*'s projection of sixteenth had it been eligible, three other oldtimers placed higher than those listed. *Super Disk Copy* would have been eighteenth; *Program Line Editor* and *Supertext II* also placed ahead of *Superscribe II.*

The breadth of the Apple market is indicated by the fact that 386 different programs received votes.

In only one of the specialized lists was the issue clear-cut.

Home 10

1. **Data Capture 4.0,** David Hughes and George McClelland, Southeastern Software
2. **Graphtrix,** Steven Boker, Data Transforms
3. **ASCII Express,** Bill Blue, Southwestern Data Systems
4. **Z-Term,** Bill Blue, Southwestern Data Systems
5. **The World's Greatest Blackjack Program,** Special Delivery Software, Apple Computer
6. **MasterType,** Bruce Zweig, Lightning Software
7. **Goodspell,** Henry G. Baker, Special Delivery Software, Apple Computer
8. **Personal Finance Manager,** Jeffrey Gold, Special Delivery Software, Apple Computer
9. **VisiTerm,** Tom Keith, Personal Software
10. **Home Money Minder,** Bob Schoenburg and Steve Pollack, Continental Software

Hobby 10

1. **DOS 3.3,** Apple Computer
2. **DOS Tool Kit,** Apple Computer
3. **Locksmith,** Omega Microware
4. **The Inspector,** Omega Microware
5. **DOS Boss,** Bert Kersey and Jack Cassidy, Beagle Bros
6. **Multi-Disk Catalog,** Chuck Hartley, Sensible Software
7. **Expediter II,** Stewart Einstein and Dennis Goodrow, On-Line Systems
8. **E-Z Draw,** Jerry Jewell and Nasir, Sirius Software
9. **Complete Graphics System,** Mark Pelczarski, Penguin Software
10. **TASC,** James M. Peak and Michael T. Howard, Microsoft

No older programs managed to crack the Fantasy 5, where *Wizardry* held forth, followed by *Ultima, Hellfire Warrior, Akalabeth,* and *Dragon Fire.*

As with each month's poll, the Word Processors 5 list saw heated competition. *Magic Window* received the highest vote count, followed by *WordStar, Superscribe II, Executive Secretary,* and *The Correspondent.* If older programs had been allowed, *Supertext II* would have been second, *Apple PIE* and *Apple Writer* would have followed *Superscribe,* and *Letter Perfect, Easy Writer, Apple Writer III, Word Handler,* and *Magic Wand* all received staunch support.

With *VisiCalc* disqualified from consideration, *DB Master* reigned in the Business 10 list. *Personal Filing System* was second, followed by *VisiTrend/VisiPlot, BPI General Ledger, VisiDex, VisiPlot, BPI Accounts Receivable, Data Reporter, PFS:Report,* and *Datadex.*

A caveat here, of course, is that many of the truly fine business products were ineligible for inclusion in the list of first-year software. *VisiCalc* would have been third; *Data Factory, Information Master,* and *VisiFile* would have followed *VisiPlot,* and *Apple Plot* would have followed *Data Reporter.*

Apple Computer had a lock on the Hobby 10 with *DOS 3.3* in first and *DOS Tool Kit* in second. *Locksmith* was third, followed by *The Inspector, DOS Boss, Multi-Disk Catalog, Expediter II, E-Z Draw, The Complete Graphics System,* and *TASC.*

Had older programs been admitted to the list, *Super Disk Copy* would have followed *DOS Tool Kit, Program Line Editor* would have followed *Locksmith, LISA* would have trailed *Expediter II,* and *Bill Budge's 3-D Graphics Package* would have followed *E-Z Draw.*

The Home 10 was dominated by communications packages, with *Data Capture 4.0,* thirty-first in the overall rankings, leading. *Graphtrix* was second followed by *ASCII Express, Z-Term, The World's Greatest Blackjack Program, MasterType, Goodspell, Personal Finance Manager, VisiTerm,* and *Home Money Minder.*

Oldsters that would have cracked this list were Howard Software's *Tax Preparer* ahead of *Graphtrix, Typing Tutor* following *Z-Term,* and *Dow Jones Portfolio Evaluator* following *The World's Greatest Blackjack Program.*

The Adventure 5 balloting provided the biggest upset. Although On-Line System's hi-res adventures continually dominate sales charts, *Zork* got the nod as the most popular adventure program. *Hi-Res Adventure #3: Cranston Manor* was second, *The Prisoner* was third, *Zork II* fourth, and *Hi-Res Adventure #0: Mission: Asteroid* was fifth.

Oldster *Hi-Res Adventure #2: The Wizard and the Princess* would have been third had it been eligible. *Creature Venture*

Word Processors 5

1. **Magic Window,** Gary Shannon and Bill Depew, Artsci
2. **WordStar,** MicroPro
3. **Superscribe II,** David Kidwell, On-Line Systems
4. **Executive Secretary,** Sof/Sys
5. **The Correspondent,** Roger Wagner, Southwestern Data Systems

The Top Thirty

	Index	
1.	160.13	**Raster Blaster,** Bill Budge, BudgeCo
2.	97.64	**Castle Wolfenstein,** Silas Warner, Muse
3.	93.70	**Apple Panic,** Ben Serki, Broderbund Software
4.	87.80	**Olympic Decathlon,** Tim Smith, Microsoft
5.	84.42	**Gorgon,** Nasir, Sirius Software
6.	82.36	**Alien Rain,** Tony Suzuki, Broderbund Software
7.	79.55	**Wizardry,** Andrew Greenberg and Robert Woodhead, Sir-tech
8.	75.33	**DOS 3.3,** Apple Computer
9.	72.43	**Space Eggs,** Nasir, Sirius Software
10.	69.15	**Sneakers,** Mark Turmell, Sirius Software
11.	68.68	**Ultima,** Lord British, California Pacific
12.	67.09	**Snoggle,** Jun Wada, Broderbund Software
13.	65.78	**DOS Tool Kit,** Apple Computer
14.	64.93	**DB Master,** Alpine Software/Stanley Crane and Jerry Macon; and Barney Stone, Stoneware
15.	54.25	**Personal Filing System,** John Page, Software Publishing Corporation
16.	49.94	**Pool 1.5,** Don Hoffman, Howard de St. Germain, and Dave Morock, Innovative Design Software
17.	48.07	**Sabotage,** Mark Allen, On-Line Systems
18.	45.35	**Zork,** Infocom
19.	44.04	**Magic Window,** Gary Shannon and Bill Depew, Artsci
20.	42.54	**Robot War,** Silas Warner, Muse
21.	42.26	**Locksmith,** Omega Microware
22.	41.23	**Gobbler,** Olaf Lubeck, On-Line Systems
23.	37.01	**Falcons,** Eric Varsanyi and Thomas Ball, Piccadilly Software
24.	36.73	**ABM,** Silas Warner, Muse
25.	35.32	**Epoch,** Larry Miller, Sirius Software
26.	35.04	**Asteroid Field,** Jim Nitchals, Cavalier Software
27.	28.39	**Threshold,** Warren Schwader and Ken Williams, On-Line Systems
28.	26.51	**WordStar,** MicroPro
29.	25.86	**Hi-Res Adventure #3: Cranston Manor,** Harold DeWitz and Ken Williams, On-Line Systems
30.	24.74	**Superscribe II,** David Kidwell, On-Line Systems

Programs that Received a First-Place Vote

ABM
A.C.E.
Adventure in Time
Adventureland
Air Traffic Controller
Akalabeth
Alien Rain (Apple Galaxian)
Alien Typhoon
Animals
Apple Adventure
Apple Panic
Apple Pascal
Apple Pie
Apple Pilot
Apple Writer
Applesoft Carpenter
ASCII Express II
Asteroid Field
Astroscope
Autobahn
Big MAC
Blister Ball
BPI General Ledger
Brain Surgeon
Bridge Partner
Bug, The
Bug Attack
Castle Wolfenstein
Classic Adventure
Complete Graphics System
Computer Quarterback
Copts & Robbers
Correspondent, The
Creature Venture
Crop Duster
Crossfire
Cyber Strike
Data Capture
David's Midnight Magic
DB Master
Decision Master
DOS 3.3
DOS Boss
DOS Tool Kit
Dragon Fire
E-Z Draw
E-Z Ledger
Easy Writer
Enhanced MX-80 Graphics
Enhanced Paper Tiger Graphics
Epoch
Expediter II
Falcons
Firebird
Flight Simulator
Galactic Trader
Galaxy Gates
Ghost Town
Gobbler
Gorgon
Grafpak
Graphtrix
Hadron
Hellfire Warrior
Hi-Res Adventure #0: Mission: Asteroid
Hi-Res Adventure #3: Cranston Manor
Hi-Res Adventure #4: Ulysses and the Golden Fleece
Hi-Res Cribbage
Hi-Res Football
Hi-Res Golf
Higher Text
Home Accountant, The
Inspector, The
International Gran Prix
Invasion Orion
Lisa
Locksmith
Magic Window
Market Charter
Micro Apple
Micro Painter
Missile Defense
Multi-Disk Catalog
Napoleon's Campaigns
Odyssey
Olympic Decathlon
Oo-Topos
Orbitron
Outpost
Paddle Graphics
Pegasus II
Personal Filing System
Phantoms Five
Planetoids
Pool 1.5
Prisoner, The
Pro-Golfer
Program Line Editor
Pulsar II
Quick & Dirty
Raster Blaster
Reversal
Robot War
Sabotage
Sargon II
Shattered Alliance
Shuffleboard
Snack Attack
Sneakers
Snoggle
Softporn
Soft Seventy
Southern Command
Space Eggs
Space Invaders
Space Raiders
Space Warrior
Star Cruiser
Star Thief
Stellar Trek
Street Life
Super Disk Copy
Superscribe
Supertext II
Sword Thrust
TASC
Tax Planner
Tax Preparer
Tellstar
Temple of Apshai
Thief
Threshold
Transforth II
Tuesday Night Football
Ultima
Utility City
VisiCalc
VisiTerm
Votrax
Warp Factor
Wizardry
World's Greatest Blackjack Program
Z-Term
Zork
Zork II

missed the list by an eyelash.

The Strategy 5 list held few surprises. *Castle Wolfenstein* and *Robot War* made the top of the list Silas Warner's private domain. *Warp Factor* was third, followed by *Hi-Res Football* and *Computer Baseball.*

Flight Simulator would have been fourth, had it been permitted on the list, and *Sargon II* was pushing *Computer Baseball.*

As with last year's poll, the awards were made at the West Coast Computer Faire by *Softalk* editor Margot Comstock Tommervik. Bill Budge was present to accept as author and Ellen Beritzhoff of BudgeCo accepted as publisher.

This was Beritzhoff's first software publishing venture and she was somewhat nonplussed when asked what she would do to top her debut. "You've got to have a good game. I guess Bill will have to program another one."

A New Age Has Begun

SDS Presents "The Professional" Series

For the first time ever, three packages encompassing all major Apple operating systems:

- ASCII Express "The Professional"™ for Apple DOS
- P-Term "The Professional"™ for Apple Pascal
- Z-Term "The Professional"™ for Apple CP/M

FORGET EVERYTHING YOU'VE EVER HEARD ABOUT COMMUNICATIONS SOFTWARE.
SDS now offers three new communication packages like no others on the market today. These packages are designed to operate alike, to look alike, and to be completely conversive with each other. An operator familiar with any one of these packages can easily operate the other two with minimal instruction. Absolutely no other packages can equal these outstanding features:

- All three packages are totally conversive with each other. ANY type of file may be transferred completely error-free within these or compatible systems utilizing the Christensen Protocol.
- Transfers may also include any ASCII based dial-up host computer. These flexible transfer capabilities are made possible by a wide variety of send options unique to SDS software.
- Support of ALL Apple-compatible modems and ALL display systems including any 80-column board or external terminal; 40-column too.
- Terminal emulation — allows your 80-column board or external terminal to "emulate" screen control codes of other terminals. Ideal for screen-oriented editors or data bases.
- Fully unattended answer mode features password only access and error-free transfer of files, as well as standard ASCII transfer modes.
- All packages support up to 1200 baud. Z-Term and ASCII Express support up to 4800 baud when interrupt-driven.
- Greatly enhanced macro system now includes programmable "hand shaking" macros — entire log-on sequences may be automatically executed. Also, many internal program commands may be initiated from macro control. Multiple macro libraries may be stored for use with a variety of hosts.

A NEW STANDARD OF EXCELLENCE
"The Professional" series is so innovative and superior to other packages in the field that it is truly in a class by itself. Our comparison chart gives a detailed point-by-point analysis of these and other communication packages. Write for it. You'll be convinced — when it comes to communication software, SDS stands alone.

SDS
southwestern data systems™

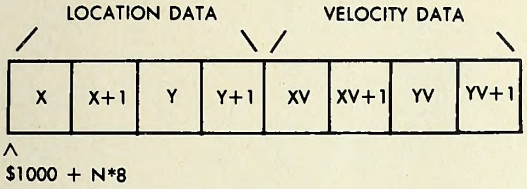
P.O. Box 582-S
Santee, CA 92071
714-562-3670

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

Assembly Lines:

from page 160

The second four bytes hold the velocity data, again in an x,y format, with byte four being unused.



LOOKUP basically does three things. First it retrieves the data for a dot, and puts it in the current X,Y,XV,YV bytes. Second, it feeds these to the calculate/plot routine. Third, when CALC/PLOT returns, the new location and velocity values are stored back in the table.

Examining the code starting at GET, you can see CTR is used to keep track of which dot we're currently processing. This is multiplied by 8 to get the base address of the data for that dot. Remember that ASL can be used to multiply easily by a power of two, depending on the number of ASLs you use. Each ASL is equivalent to multiplying by 2.

Once the base address offset is determined, this is put in the X register and the data retrieved via a series of LDA/STA operations.

After returning from CALC/PLOT, the process is reversed to store the new data.

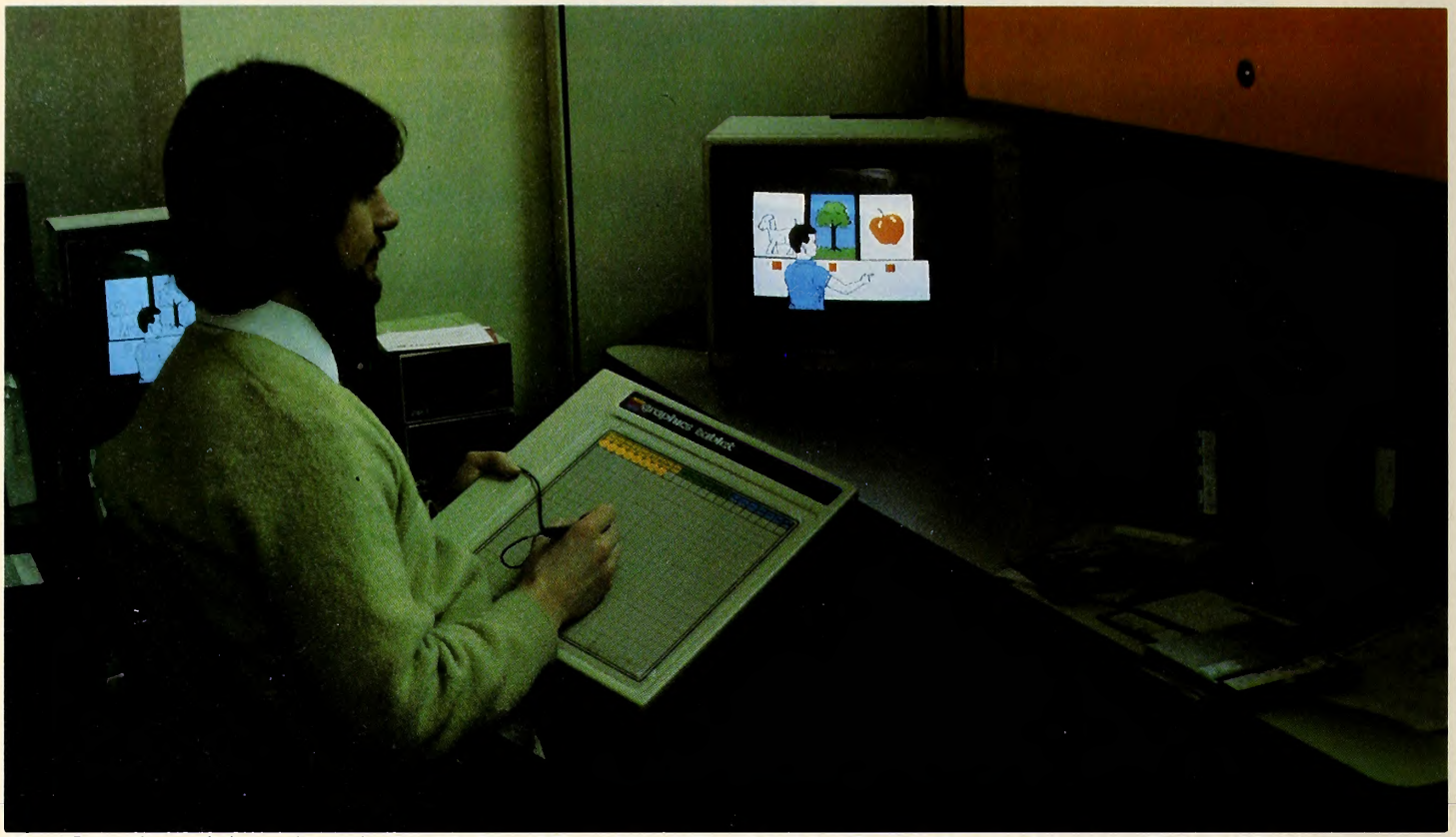
Wrapping It Up. Hi-res is an involved topic, and it's challenging to try to present the right mix of clarity and in-depth explanation. My goal is to present enough of the basics to give you the springboard to pursue your own interests.

In general, the principles provided in this, and last month's article are the foundation of most animated graphics programs. Tables are especially worth your consideration as they provide a straightforward way of managing a larger number of screen points.

By now it should also be evident that even in machine language, the Applesoft routines themselves are still the most restraining portion in terms of speed and execution. In all fairness to Applesoft though, realize that their speed is sacrificed for simplicity and convenience of operation.

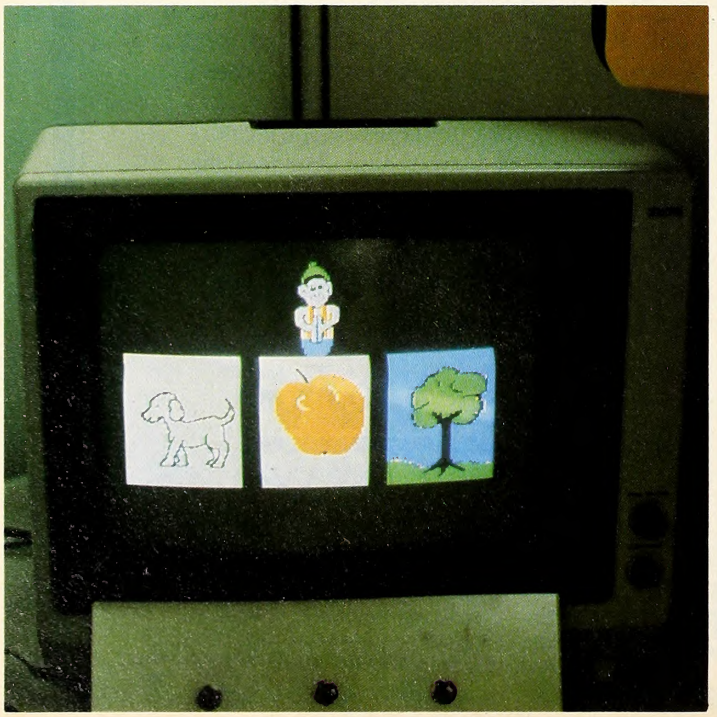
Next month's topic will be the layout of the hi-res screen itself, and how certain dedicated routines can be created to get a little more out of the ol' Apple.

See you then!



Above, Fox working with the graphics tablet. Top right, Bernard Fox and Mary Wilson with students from the Courseware Challenge program: Jim, Lakshmi, and Wendy at the computer. Bottom right, the vocabulary training program with the Blob. Below, Receptive Syntax and Vocabulary Testing and Training research program (American Speech and Hearing Association demonstration).

Maple Sugar and Apples:



BY DAVID DURKEE

When computers first became available for use in schools and other educational settings, a new field in education was born: Computer Assisted Instruction (CAI). At first CAI was met with the kind of hi-tech enthusiasm that many new developments in computer applications enjoy, but after a while, once the novelty wore off, the enthusiasm waned.

One survivor from this period is the Plato system based in Minnesota. This is a time-sharing system with an extensive bank of courseware (educational software). Plato is accessed by smart terminals over telephone lines, and while the variety of courseware available makes the system a useful educational tool, the cost of long-distance phone calls at day rates makes it prohibitively expensive for most potential users.

Getting It To The People. Since the advent of the microcomputer there has been a second wave of research in CAI. Microcomputers are within the range of even an elementary school's budget, and some commercial courseware is available.

Mary Wilson and Bernard Fox of the University of Vermont are pioneers in the development of microcomputer courseware for special education. Both are speech-language pathologists in the Department of Communication Science and Disorders at the University of Vermont, trained to help people who have problems in using and understanding language. Although millions of people have such problems, Dr. Wilson ob-

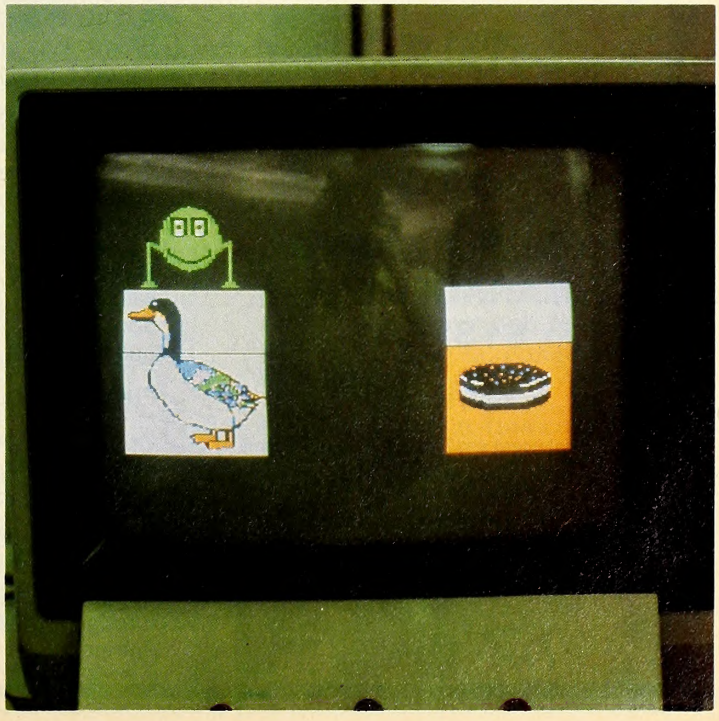
A Yummy Way to Learn

served in 1979—the beginning of the microcomputer boom—that these people's needs were not being addressed by the new developments in CAI. "We wanted to see that the handicapped were included rather than excluded from the microcomputer revolution," Wilson says now when describing the thrust of their work.

She and Fox, who had previous computer experience, set out to address this need. Their first major step was to acquire an Apple II with university funding. Wilson recalls, "When we first began developing audible microcomputer courseware, our colleagues questioned whether young children would even interact with the computer, so we designed a study to address this question."

For purposes of this study, they designed a Receptive Language and Syntax Testing and Training program. Although not a fully developed piece of courseware, the program contained many of the elements of one. It used a two-step process to teach the prepositions "in" and "on," and to test for comprehension of those prepositions and "under," which was used as a control variable. It taught with a bombardment technique and tested with a feedback technique.

As Dr. Wilson explains it, a normal child can pick up the correct use of these and other constructions by the examples provided in normal conversation. The bombardment technique is more like formal drilling: "The ball is under the table. Look at the ball under the table. See the ball under the table?" The repetition helps clarify the relationships to handicapped



learners. The testing phase actually teaches through feedback as well; when an incorrect answer is given, the system corrects it immediately and tells the student the right answer.

To be effective, this program relies upon both visual and aural components. Obviously students with these kinds of learning difficulties cannot read text on the computer screen, so a Mountain Hardware Supertalker was called into play. The Supertalker digitizes sounds, stores them in binary memory, and "speaks" in a voice that resembles a human one. A visual component was created using an Apple Graphics Tablet and a machine language routine that takes a rectangular subsection of the graphics screen and moves it to any other screen location. The routine was developed by Leslie Smith, then a senior in UVM's Computer Science department.

In the program, the hi-res screen takes the place of picture flash cards. Three pictures, each showing the same two objects in different configurations, appear on the screen and the voice says, "Show me the ball on the box." The learner responds by pressing one of three buttons on a special board connected to the Apple's I/O port. If the child answers correctly, a clown appears on the screen, claps its hands, and says "Good!" If the child's answer is wrong, the correct answer is indicated and the question is repeated.

The findings of this research indicated that handicapped students would indeed interact with the computer. In fact, all but one of the children participating in the test preferred the computer test over the flash card version. In November of 1980, Wilson and Fox presented the program as the focus of a scientific exhibit at a convention of the American Speech and Hearing Association. In ASHA's national competition, the exhibit took first prize for excellence of presentation and second for scientific merit.

By winter of 1981 Wilson and Fox were ready to share what they had learned about CAI for the handicapped, and to put some of its principles to actual use. They submitted proposals for two related projects: the first, submitted to UVM's Living Learning Center was called the courseware challenge; the second, submitted to United Cerebral Palsy of Vermont, was a telecommunications project. Both proposals were accepted.

Design for Learning. The Living Learning Center is an educational and residential environment in which undergraduates live together in *programs.* Each program is designed to teach a particular subject in a way that may or may not include lectures. Programs usually feature non-classroom learning; such learning may take the form of group or individual projects, discussions, field trips or other events.

The Center currently houses faculty and student designed programs on theatre, unspoken language, parapsychology, emergency medicine, Irish studies and holography. The Courseware Challenge proposed to teach the history and techniques of CAI to a group of UVM students with a variety of academic backgrounds.

When the Courseware Challenge program got off the ground this fall, ten men and women were enrolled. The students live in two suites; Apples, printers, disk drives, modems and other peripherals, including a graphics tablet, the *Supertalker,* and the *Echo II* reside nearby in a two-room office. Students in this program learn firsthand about CAI and help write and modify programs for Wilson and Fox's projects.

The Spice of Life. Wilson and Fox wanted their students to be a productive part of the courseware development process. They value varied background and imagination as much as they do computer expertise, so they selected program members with such varied majors as electrical engineering, computer science, education and communication science and disorders.

The projects the students are working on are as varied as their majors. Mark Biamonte is working on a touch-sensitive overlay for the computer screen designed to interface with the Apple and tell it where a user is pointing. By relating that to what is displayed on the screen, the computer would be able to accept simple pointing as input.

THE NEW STEP BY STEP

A Programming Course For Beginners

What is the best way to learn how to use an **APPLE®** computer?

You could enroll in a class — arranged at someone else's convenience, given by a technician who may not be a very good teacher. Or you could read a book — written by a programmer who may not be a very good writer. Or you could learn in your own home, on your own **APPLE®**, at your own convenience, using the course that has become a standard of the industry.

Introducing THE NEW STEP BY STEP, a programming course for beginners

In 1978, PDI introduced the original *Step by Step*. The student learned by interacting with his or her computer. Thousands of students learned how to use an **APPLE®** computer and the **Applesoft®** language with *Step by Step*. Now, using feedback from students, teachers and learning specialists, we are introducing *The New Step by Step*.

The New Step by Step, in addition to sound and effective teaching, includes computer graphics, animation, sound effects and a voice track.

How does THE NEW STEP BY STEP work?

The computer program shows screen displays or sample program outputs, while the audio cassette explains to the student what is being shown. After each instructional segment, the student is asked a question or asked to solve a problem. The computer checks the student's work.

When a lesson has been completed, the student turns off the cassette and goes to the *Step by Step Workbook* to review and

practice the material covered in the lesson. After the practice assignment has been completed, the student takes a quiz.

A final exam is included.

What does THE NEW STEP BY STEP teach?

There are about twenty hours of instruction. Topics covered include:

- **PRINT** instruction
- Writing simple programs
- **INPUT**
- Loops **GOTO**
- **IF-THEN**
- Program counters
- Library functions such as **INT** and **RND**
- Screen formatting, including **TAB, VTAB** and **HTAB**
- Subroutines
- **READ-DATA** statements
- **FOR-NEXT** loops
- One-dimensional arrays
- Nested loops
- Low-resolution graphics
- Multiple-statement lines
- Program logic
- Floating point notation

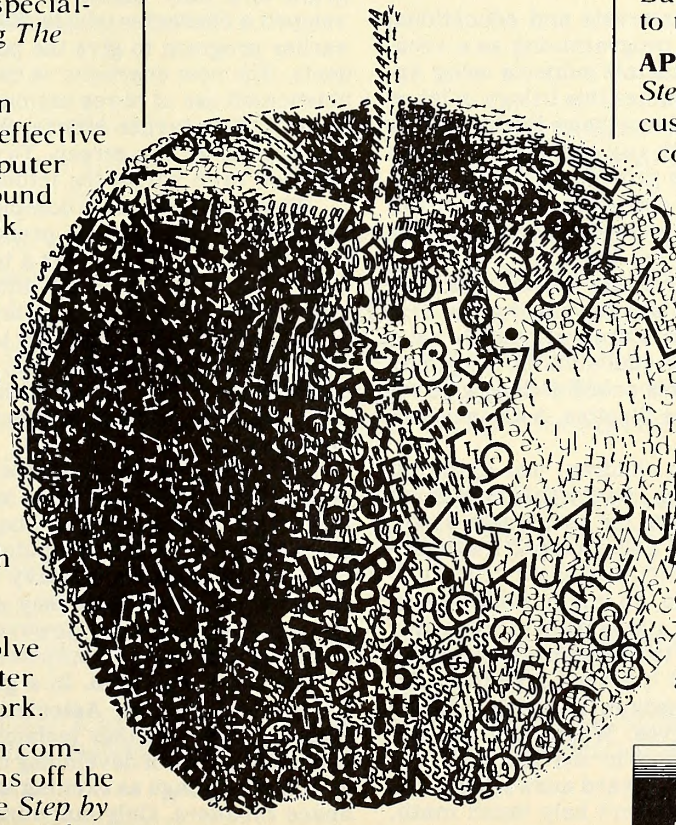
How can THE NEW STEP BY STEP be used?

Schools can use *Step by Step* for individualized instruction in programming in **BASIC** and in computer literacy programs.

Families can use *Step by Step* to ensure that all family members can use the family computer.

Businesses can use *Step by Step* to train their personnel.

APPLE® dealers can use *Step by Step* to introduce prospective customers to the **APPLE®** computer.



What are PDI's plans for the future?

We are now in the process of developing *Step by Step Two*, which will teach intermediate **BASIC** programming, disk operation and hi-res graphics. *Step by Step Three* will teach machine language and assembler.

Available at fine computer stores. Or, directly from **PDI**, for $79.95 plus $3.00 shipping and handling.

*Apple is a trademark of Apple Computer Corp.

Program Design, Inc.
11 Idar Court
Greenwich, CT 06830
203-661-8799

Since the first impuse of many learning-impaired students in answering a question is to point to the correct picture when answering a question, this capability would be quite valuable in connection with the syntax training and testing courseware. Children have to be taught to equate the buttons with the picture. In some cases, children would point to the correct picture but press the wrong button. Biamonte plans to use either two overlaying sheets of clear plastic with conductors embedded, or a matrix of infrared light which would be interrupted by the user's hand. Although there are touch-sensitive screens available for the Apple, they are very expensive and have a much higher resolution than is necessary for these applications. The goal here is to be able to develop a screen suited to their special needs for under one hundred dollars.

John Wilson developed a special joystick that enables a student who has poor coordination in her hands to use her elbow instead, but some victims of cerebral palsy find even this device difficult to use.

Jamie Gump, another program participant, is planning to develop eyes of a kind for the Apple. Using a pair of distance sensors of the sort Polaroid uses in their newest cameras, he hopes to enable severely handicapped students to use their heads in the manner of joysticks. The two distance sensors would be aimed at the user's head from different sides to triangulate on its position. The sensors plug into the paddle circuits of the I/O port, making them, in effect, a joystick. In combination with the necessary interpretive software, such a device makes selecting a program from a catalog or an option from a menu as easy as moving your head from side to side.

Other students are involved in creating games. These too have a courseware purpose: they are a simple enjoyable way to train handicapped users in the motor skills they'll need to operate their special interface devices. They can also help handicapped users overcome the social barriers imposed by their handicaps.

Unlocking Communication. The United Cerebral Palsy project proposal involves tutoring people with cerebral palsy at home using Apple computers. United Cerebral Palsy provides the eight participants with Apples in their homes on a rotating basis.

Tutoring is geared to students' interests and educational levels. Some students are tutored in programming as a vocational skill; others are tutored in academic subjects using Apple Computer's *Shell Games*. What makes this trilogy of interactive matching, multiple choice, and true/false testing games so useful is that they can be set up with any sort of database an instructor wants to use. The database provides space for questions, multiple possible answers, and comments on the subject to be added after a question has been answered correctly. This means instructors can not only test, but teach.

Among the cerebral palsy programs that Wilson, Fox, and their associates have developed and are developing are programs to help the users communicate. Cerebral palsy is not a learning handicap, but it does hinder education by hindering communication. Cerebral palsy affects coordination; as a result, individuals' handwriting may be illegible, both to others and to themselves.

Ann Peery, a mother with grown children, just appeared one day to join Wilson and Fox's group. Aware that communication in writing can be a real problem for people who suffer from cerebral pasly, Peery developed a scratch-pad math program for them.

This program allows students to do math problems in "longhand" on the computer screen. It allows them to move over to the side to do related problems, or to move up to the top of a problem to do a carry or borrow, but it doesn't give them the answers; it just provides a legible medium in which students can work out problems for themselves. When finished, students can print out problems to hand to the teacher. The scratch-pad program goes a long way toward answering some educators' criticisms that computers don't help teach math, they just give the answers.

Students in the UCP program also use *Apple Writer* as a communication tool. *Apple Writer* does for them in English what the scratch-pad does in math—it allows them to communicate legibly. It has an advantage over even the best typewriter, in that it makes editing simple. Apples do not so much teach students to write effectively as allow them to show that they can communicate in writing. And both the scratch-pad and *Apple Writer* enable students to do their work much faster.

Fox is also working on a program to help the speaking disabled to communicate. He is constructing a database of words with both normal and phonetic spelling, and devising a program that can access any word with a few keystrokes. The program builds the words into phrases and sentences, and then speaks them with Street Electronics' *Echo II* speech synthesizer. Fox hopes to put as much of the program as possible under joystick control. The real challenge, he says, is organizing information and instructions on the text screen to make possible the easiest and fastest use.

Embarrassment of Riches. In addition to their current active projects, Wilson and Fox have some extensive plans for the future. One of these plans is to expand their current work in training special educators to use microcomputers as teaching devices. They have letters of recommendation from educators throughout Vermont that speak out strongly for this need; where Vermont was once a state with more cows than people, it now seems to have more microcomputers available for special education than it has special educators trained to use them. A special education coordinator put it most succinctly: "We are equipment rich and training poor."

The proposal is a three-year plan extending from June of 1982 through May of 1985, that includes periodic training sessions, workshops, modules, and full courses, on such subjects as the different computers available, available hardware and software for those computers, how to adapt commercial hardware and software to special uses, and special techniques in using microcomputers with the handicapped. They also plan to provide newsletter updates and phone consultation on a continuing basis.

Another of their proposals is to expand their vocabulary and syntax testing and training software from a research program to a fully usable piece of courseware. They have developed a character who is more active than the clown in their earlier program to give the positive reinforcement to the students. The new character is called the Blob. Making more sophisticated use of hi-res animation techniques that they developed in their earlier efforts, the Blob walks, talks, rolls, and jumps all over the screen. To make things more interesting, the Blob even varies the intonation of his voice.

When his database is complete, the Blob will be able to train students in prepositions, pronouns, who and what questions, verb tenses, and more, for a total of thirty-four different constructions, three hundred different questions, and over one thousand high resolution pictures. The program makes full use of the Apple's 48K of memory to minimize disk loading time; a memory map shows that only about three hundred bytes remain unused. Fox explains that it is necessary to minimize disk I/O time because students lose interest when they are kept waiting.

The Blob currently uses the *Supertalker* to speak, but they would like to convert it to use with the *Echo II*, which takes up much less memory space. The problem is that they are wary of teaching children a computer accent. Studies show that children of parents with a heavy foreign accent develop normal American intonations if they are also sufficiently exposed to nonaccented speakers. However, no studies have been done to show the effects of a computer accent on a developing child.

Unlimited Potential. In a guest editorial for the American Speech and Hearing Association Magazine, Wilson and Fox stated, "Today neither technology nor the cost of technology stand in the way of developing creative courseware that makes learning language as exciting as any arcade computer game of *Space Invaders*. Only our mind's imagination limits us."

The enthusiasm inherent in that statement shows in all of their work. ■

A TRULY EXPENSIVE FREEBIE!

Softalk's been mailing more than sixty thousand magazines to Apple owners each month. There's nothing remarkable about that; Reader's Digest mails two hundred times as many.

What's escaped notice is that Softalk is the most expensive widely distributed magazine in the world. After all, you've got to have paid more than two thousand dollars for an Apple before you're eligible to receive Softalk.

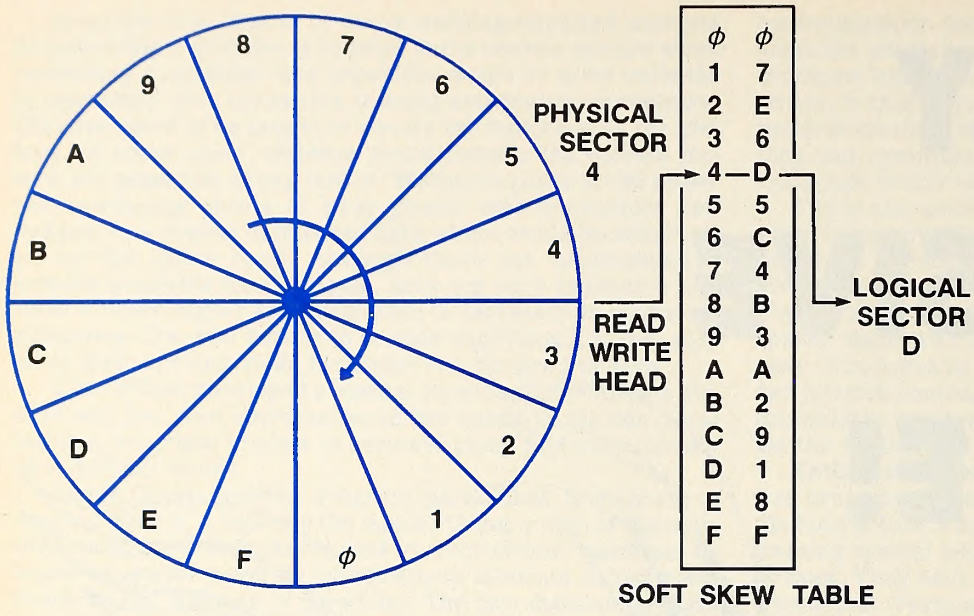
Once you've taken the plunge, bought your Apple, and received a free trial subscription to Softalk, the magazine costs $24 per year—less if you are a sponsored subscriber. That's a cost of about one percent a year to keep your Apple current with the state-of-the-art.

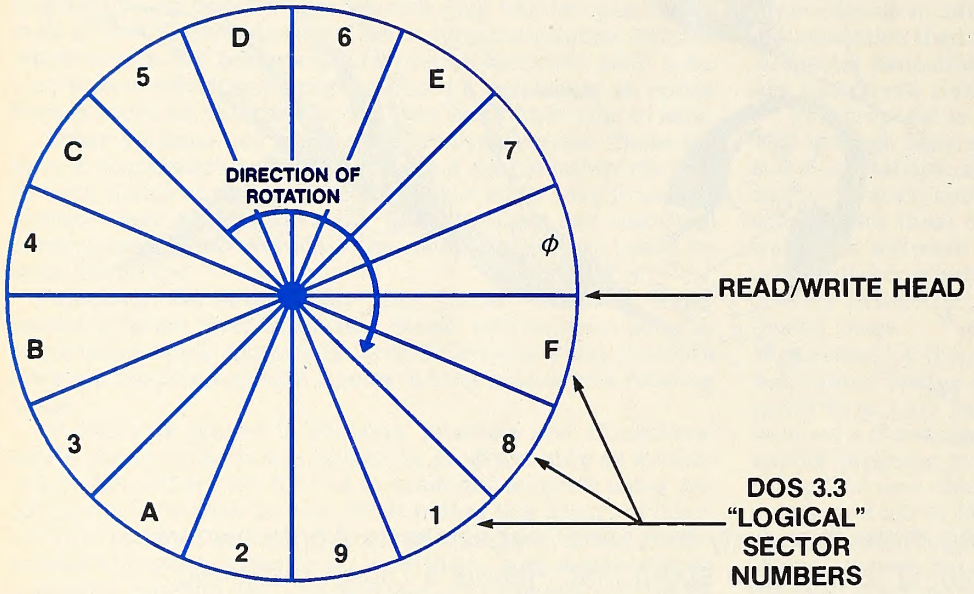
Keeping up with the Ataris is no longer a trivial task, considering the rapidly changing state of the Apple market. But Softalk stays abreast of the most current developments through its Marketalk News and Reviews columns as well as various applications columns.

Making your personal computer a personal tool is not as simple as it sounds. Let Softalk help.

Free trial subscription to all Apple owners who have never before received Softalk. Subsequently: $24 per year for unsponsored subscriptions; $18 per year for sponsored subscriptions. Mail to:

Softalk Circulation
11021 Magnolia Boulevard
North Hollywood, CA 91601





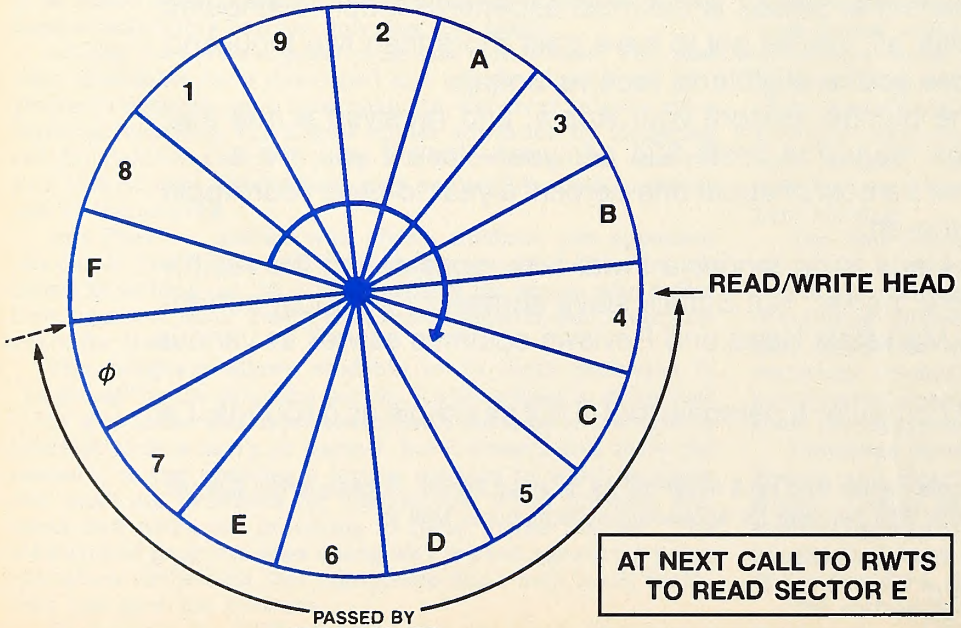


Figure 3.

BY DON WORTH AND PIETER LECHNER

Sector skewing, or interleaving, is a method by which the time required to access a disk sector can be reduced. Whenever you initialize a disk, either by using the DOS *init* command or by using the *copy* program to create a new disk, DOS formats each track by arranging the sectors in a "physical" order.

In DOS 3.3 this order is sequential, from physical sector zero to physical sector fifteen. When DOS 3.3 reads a sector it translates the physical sector number into a "logical" sector number using a soft skewing table, located in the read/write track/sector portion of DOS. Most programs and most of DOS refer to sectors by their logical sector number. When DOS 3.3 was written, the soft skewing table was established so that the logical order in which sectors were arranged on a track could be controlled merely by modifying the soft skew table. (Earlier versions of DOS set up sector skewing when the physical sectors were placed on the disk at *init* or copy time—in fact, under DOS 3.2 and DOS 3.1, *init* and copy each used a different skew pattern!) Figure 1 presents a diagram of the process used by DOS 3.3 to reference sectors.

Going around with a Disk. To understand skewing, it is enlightening to look at the series of events that occurs whenever a sector is read from or written to the disk. Figures 2 through 4 describe what happens when a program is read from a standard DOS disk. To simplify this discussion, these three figures show the *logical* sector numbers. In actual fact, the physical order is that of figure 1.

Shown in figure 2 is the standard skewing for DOS 3.3. There are different skews for Pascal/Fortran and CP/M. The reason for this should become apparent as this discussion proceeds. When DOS 3.3 was designed, a skewing wash was chosen that optimizes access during booting. Since sector reads occur fairly rapidly during the boot process and sectors are read in reverse order (sector F first, then E, then D, and so on), a "2 descending" skew was chosen. This means that the next lower sector is (nearly) always two sectors away from the last one. For example, sector 6 is two sectors beyond sector 7. Although this standard DOS skewing allows disks to boot in at optimal speed, it's a very poor skewing for almost anything else, such as loading or running programs *(bload, load, brun, run)*.

In figure 2, the read/write head is positioned where it would be immediately after reading Sector F. At this point, control returns from RWTS and the file manager processes the data it has just read, determines which sector must be read next, and calls RWTS again.

Missing the Boat. At the same time all of this is going on, the disk continues to

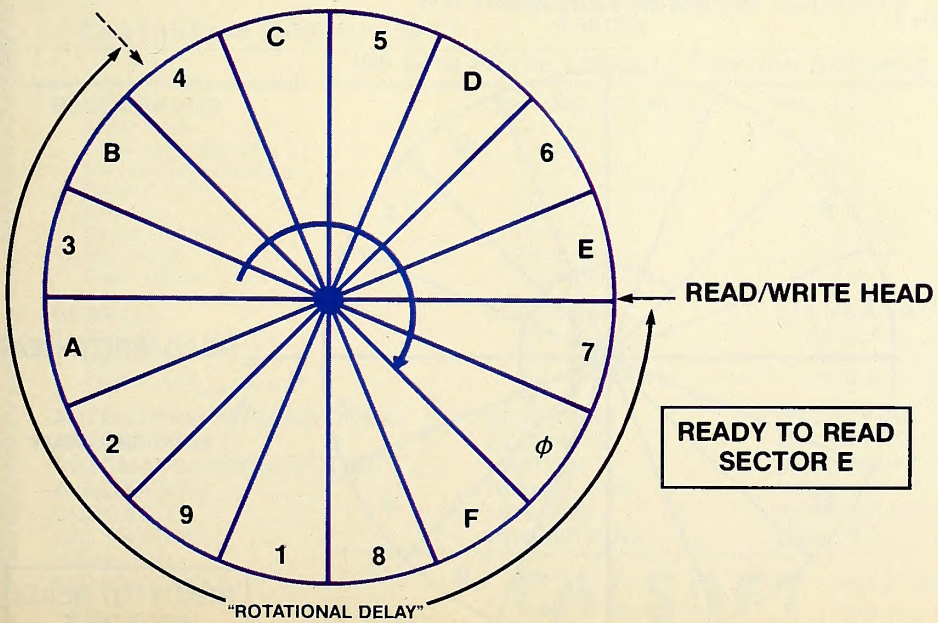


Figure 4.

spin in the drive. Thus, several sectors pass beneath the read/write head before the file manager is ready to request another read. In the usual case, the file manager will want to read the next lowest numbered sector on the track (sector E, in this case), and this sector will already have passed by. Figure 3 shows the position of the disk relative to the read/write head when the file manager is ready to read sector E.

Notice that one of the sectors passed over was, in fact, sector E. RWTS will now read each sector on the disk until it again finds sector E. The period of time it takes to find the desired sector after the read has been requested is referred to as the *rotational delay* for the disk access. Figure 4 shows the read/write head in position to read sector E and identifies the rotational delay. This delay is wasted time.

You may have noticed that, had the file manager processed sector F faster letting only one sector pass by in the meantime, it could have returned to the disk in time to read sector E as it passed

under the read/write head. This is apparently what the DOS designers intended and, in fact, this is what occurs during the boot process. The example described, however, was that of a *bload* operation (*bload, load, brun,* and *run* have similar patterns of behavior, using identical code within DOS).

Making Up Time. Obviously, unless you spend most of your time booting disks, you'd appreciate the reduced rotational delay for other operations as well. One solution would be to change the soft skew table in RWTS to a more appropriate pattern. This works, but it has two major disadvantages. One disadvantage is that, once this table is changed, any disk created using the standard skew table is not accessible using the modified soft skew table, and vice versa. For example, what standard DOS thinks is sector 7, the modified DOS may think is sector 5. The second disadvantage is that this scheme applies to every track on the disk, including the boot image of DOS on tracks 0, 1, and 2, making a *bload* command run faster at the expense of the time required to boot the disk.

Luckily, there is another way to deal with the problem. Instead of changing the soft skew table, the sectors can be physically rearranged on the track such that, when run through DOS's boot-optimized skew table, they will be in the optimal pattern for *load* and *bload*. Using this method, it's possible to have different physical arrangements on different tracks. Thus, tracks 3 through 34 can be optimized for *loading* while the boot tracks are optimized for booting. Also, disks created by physically rearranging the sectors may be read by any DOS (of the appropriate version).

It turns out that during a *bload* operation the file manager is "out to lunch" processing the previously read sector for about the time it takes eight sectors to pass beneath the read/write head. Thus a "9 descending" skew seems a good choice (9 descending provides eight sectors of padding between descending sequential sectors). Figure 5 shows the logical arrangement of sectors on the disk when a 9 descending skew is used.

Using this new skewing, after the file manager has finished processing sector F, the read/write head is positioned very near to sector E. Figure 6 shows the position of the disk at this time. The actual rotational delay is less than one sector long, as shown in figure 7.

What You Gain. You might ask just how significant a contribution rotational delay makes to the overall access time of a disk sector. Within about a 10 percent tolerance, a Disk II spins at three hundred revolutions per minute. This means that every sector on the track passes beneath the read/write head three hundred times per minute. If rotational delay amounts to waiting for ten sectors to go by before reading the desired sector, as in the standard DOS skewing with

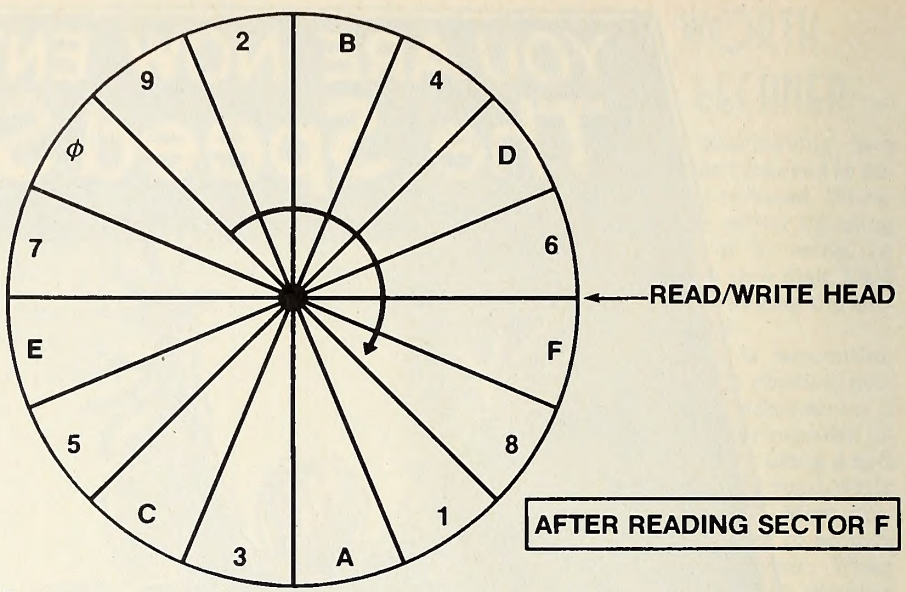


Figure 5.

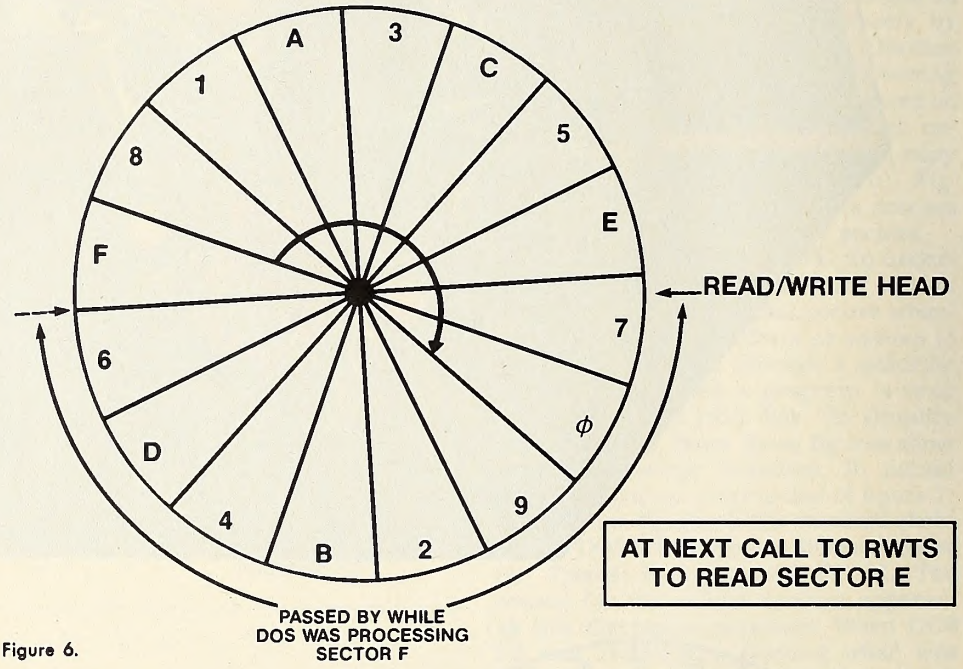


Figure 6.

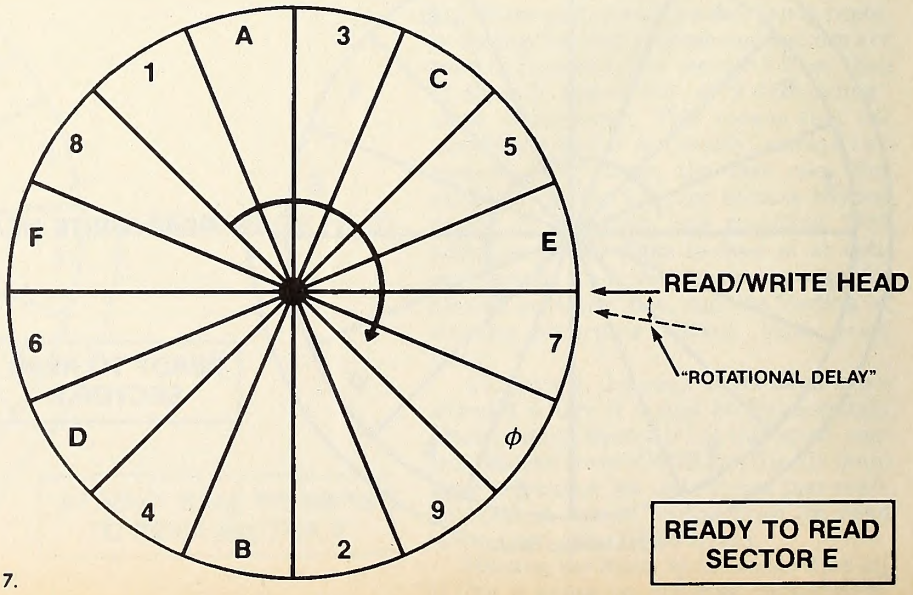


Figure 7.

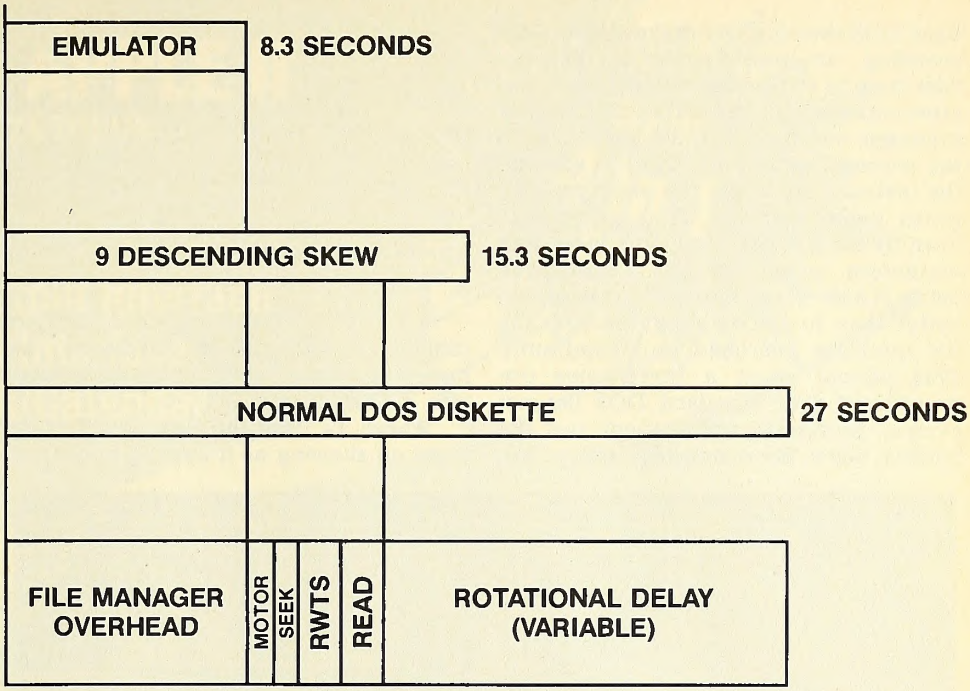
bload, and sixteen sectors are read on every track, the time wasted amounts to:

1 min × 60 sec × 10 lost sectors × 16 reads × 1 rev
300 rev min read track 16 sector

= 2 lost seconds per track

This value is theoretical and is not completely accurate, since rotational delay varies, depending upon factors such as random placement after moving to a new track, variations in DOS response, and so on. However, experimental measurements have show it to be within 10 percent of the actual delay in most situations. If an Applesoft program stored in sixty-four sectors on the disk is loaded, this rotational delay amounts to eight seconds. With the nine descending skew, the delay is reduced to less than half a second. Obviously, the more sectors accessed the greater the rotational delay. The chart in figure 8 shows the times required to *bload* a binary program that is $7000 bytes in length ($70 sectors or 112 decimal). Using both experimental and theoretical methods, the component delays to a complete disk access have been found.

Figure 8 shows that rotational delay amounts to more than 50 percent of the time required to load the file with stan-



MOTOR — Delay waiting for disk motor to come up to speed
SEEK — Time spent moving disk arm from track to track
RWTS — Read/Write Track/Sector overhead: Postnibble etc.
READ — Time spent reading data sectors

Figure 8.

dard DOS skewing! By reskewing to 9 descending, an improvement in disk access time of 43 percent was realized. Another interesting point is that the DOS file manager overhead is quite high. This is the processing time required to execute the instructions in the file manager program itself, and this time is required even by the commercially available disk emulators, which simulate a disk drive using RAM. Thus, emulators can do no better than to cut the delay back to the file manager overhead as a minimum. This effects about a three-times improvement over standard DOS but not even a two-times improvement over the optimal skew. By reskewing, with no additional investment in hardware, we have gained almost 2/3 of the advantage of a disk emulator.

Time in Seconds

Activity	2 Desc. Skew	9 Desc. Skew	% Difference
Boot	7.0	6.6	6
Boot/load lang card	19.5	13.3	32
LOAD Basic File	8.0	5.0	38
SAVE Basic File	12.7	11.3	11
BSAVE X,A$800,L$7FFF	44.2	39.6	10
BLOAD X	31.7	17.4	45
Read Text File	7.9	8.4	−6
Write Text File	10.2	9.0	12
CATALOG	2.7	2.6	−4

Figure 9.

Where To Find the Key. One further word on skewing as it applies to nonprogram files is in order. The pattern of access of a basic program to its data files varies drastically from program to program. Frequently an application will not return to the disk to read the next sector for many revolutions—it might even allow the drive to turn itself off! In this case, sector skewing becomes a consideration of relatively low importance. Since access can also be affected by the number of records in each sector, it becomes almost impossible to identify a repeating computational delay. The best that can be said is that it does not matter much which skew is used for data files and 9 descending will probably work as well as any other.

The table in figure 9 shows the difference between the standard 2 descending skew and a 9 descending skew for typical disk activities. It should be noted that these values may vary, because a number of variables can affect the outcome, including hardware and even the brand of disk used.

With information given in chapter 8 of *Beneath Apple DOS* and a disassembly of the read/write track/sector portion of DOS, an assembly language programmer can create a program that will format an entire disk, changing the normal order of the physical sectors on each track so that skewing is arranged to his preference. Although somewhat difficult technically, this task is fairly straightforward. A simple modification to DOS's *init* code yields a method for reskewing every track of a blank disk, but this does not address the need for using a different skewing for tracks 0 through 2 than for the remainder of the disk nor does it permit the user to update an existing disk. These last two features are provided by the *Init* program that comes with the disk utility *Bag of Tricks*. *Init* will format, or reformat in place, one or more tracks with any desired skewing. ◨

Don Worth graduated from UCLA with a computer science degree and has been a systems programmer on large-scale IBMs for the past ten years. Worth is also the author of Beneath Apple Manor *and* Linker. *He wrote* Beneath Apple DOS *with Pieter Lechner. Lechner, whose background includes aerospace engineering and construction, started working with Apple computers in 1977. He is now employed in software development at Quality Software in Reseda, California.*

NEW...FOR APPLE II[1]

FROM PROMETHEUS

VERSAcard

1. Serial Input/Output Interface
2. Parallel Output Interface
3. Precision Clock/Calendar
4. BSR Control

NEW 4 in 1 CARD

ALL ON ONE CARD
with true simultaneous operation
. . .at a down-to-earth price!

CHECK THESE FEATURES: True simultaneous operation! VERSAcard's unique hardware selection circuitry make your Apple "think" that separate cards are installed. This allows VERSAcard interface ports to be compatible with existing software such as APPLE PASCAL, Microsoft Softcard[2], and most other Apple software.

The Serial Input/Output port is RS-232C standard with crystal controlled baud rates from 19.2K to 110 baud. You can connect with a CRT terminal, line printer, or virtually any RS-232 modem. VERSAcard's "Terminal" mode, along with modem, connects you with the *Source* or a variety of other computers. "Terminal" mode also allows you to output to both printer and display simultaneously.

The versatile **Parallel Output Interface** is configured for the Centronics standard and is easily configured for other standards. Simply hookup your **Epson, Okidata, IDS,** or other printer and begin printing.

Precision Clock/Calendar. With all standard real-time clock/calendar functions and one second resolution. Battery back-up supplied. And, best of all, it's **software compatible with Thunderclock.** This makes numerous other software packages immediately usable. Interrupts are supported.

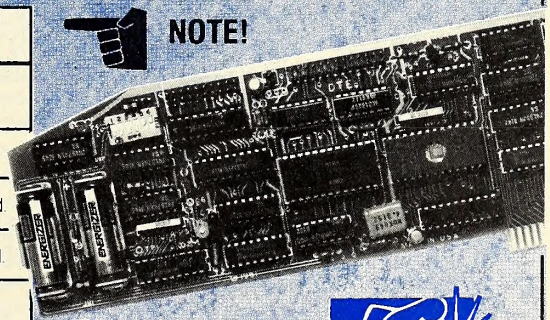
BSR Control. Add an ultrasonic transducer, and VERSAcard provides you with remote control through your BSR X-10 Home Control System.

Onboard firmware provides optional auto line feed, video, paging, terminal mode, lower to upper case, 8th bit set or clear ... and more. These versatile device drivers permit connection to virtually any standard peripheral. The Prometheus engineering staff is available to help. With any peripheral and special device interface problems. Call us, we'd like to hear from you.

VCD-1 VERSAcard @ $249.00 each.

Compare Key Features	VERSAcard	CPS MulliFunction Card
Parallel, Serial and Realtime Clock Interfaces?	YES	YES
BSR Control Interface?	YES	NO
Compatible w/ Apple Pascal?	YES	NO. Special Pascal disk required.
Compatible with CP/M?	YES	NO. Special CP/M disk required.
Interrupts?	YES	NO
Apple Modification?	NO	YES
"**NEW** . . . Applications/Diagnostic disk—available 1st quarter '82"		

NOTE!



Dealer inquiries invited.

VERSAcard—the only Apple peripheral card you're likely ever to need!

All Prometheus boards are completely tested and burned-in prior to shipment. A one-year warranty covers parts and labor.

Registered Trade Marks: (1) Apple Computers, Inc. (2) Microsoft Consumer Products

PROMETHEUS

PROMETHEUS PRODUCTS INCORPORATED

45277 Fremont Blvd. • Fremont, CA 94538 • (415) 490-2370

Ask about the new PROMETHEUS 128K RAM card.

PROGRAMMING THE APPLE II JUST GOT EASIER!

ANIX v1.0

A UNIX-like DOS 3.3 Compatible Operating System for the Apple II

ANIX 1.0 is a "Tiny-UNIX™" that is completely file structure upwards compatible with DOS 3.3. ANIX incorporates a healthy subset of UNIX'S™ capabilities and a superset of CP/M's™ capabilities. In addition to the ANIX operating system, Lazer provides over 30 utility programs on the ANIX disk. Included are: ADU (ANIX disk utility), DUPDSK, EXPUNGE, DREN, STAT, TYPE, PUTDOS, FREE2, UNDEL, MAKE, SIZE, MON, NOMON, WC (word count), LC (line count), PIE.PATCH, PRINT, BUILD, FIND, BTOT (binary to text conversion), INIT, MAKEAUTO, PAG3, ED, COPY, SETCLK, SETPRTR, SETLOWER, SETINVMODE, TIME, and more! Many of the programs found on the ANIX disk are selling alone for between $30 and $60. Also included are the source listings for several of the utilties. For advanced machine language programmers, the source listing of the entire operating system is available separately. ANIX supports powerful screen editing commands and will automatically time and date stamp files saved to disk (if a clock card is installed). For a complete description of ANIX's capabilities, call or write Lazer.

ANIX is the start of a complete line of system software tools available from Lazer MicroSystems, Inc. All new languages and applications programs available from Lazer will run under the ANIX operating system. Lazer Pascal is available now. Other languages and systems are in the works. Productive programmers are already using ANIX, are you?

Lazer Pascal

The Elegance of Pascal-- The Power of "C"

Lazer Pascal is a unique systems programming language for the Apple II. It combines features found in Pascal and is extended to include several features found in the "C" programming language. The Lazer Pascal compiler is very fast (1500-2000 lines/minute) making the system very easy to use. No longer will the systems programmer or game programmer be forced to use assembly language, Lazer Pascal is here!

Lazer Pascal supports BYTE, CHAR, BOOLEAN, INTEGER, LONG, pointer, string, array, static, and dynamic data types. Lazer Pascal was created to replace 6502 machine language as the choice of systems and game programmers. Included with Lazer Pascal are several sample programs demonstrating the use of Lazer Pascal including: ANIX.P, TSTPARMS.P, LINECOUNT.P, WORDCOUNT.P, CHARCNT.P, EXPAND.P, COMPRESS.P, CRYPT.P, and TRANSLIT.P. Better yet, the source listings for the compiler, P-code interpreter, and other utilities are included.

D/ASM

A High-Powered 6502 Disassembler for the Apple II

DISASM/65 produces a 6502 assembly language source listing from machine code and a set of input commands. Only DISASM/65 supports all the commonly used data types found in machine language programs. We used DISASM/65 to disassemble DOS 3.3 for our popular DOSOURCE 3.3 product-- that should describe DISASM/65's power! DISASM/65 is provided with our popular LISA V2.5 assembler. Several users, however, have reported considerable success using DISASM/65 with the Toolkit assembler, the SC Assembler, TED, and others; so we are offering DISASM/65 separately for these users.

p-SOURCE

for Apple Pascal users

The Internals of the Apple P-code Interpreter Explained

p-SOURCE is a technical manual that describes the internal operation of the Apple Pascal P-code interpreter. Included are descriptions of programming techniques used within the interpreter, hints on how to speed up the interpreter, add your own routines to it, and incorporate hardware floating point. p-SOURCE is absolutely essential to the Pascal programmer.

ANIX, Lazer Pascal, p-SOURCE and DISASM/65 were all written by Randy Hyde, the author of "USING 6502 ASSEMBLY LANGUAGE", LISA, SPEED/ASM, DOSOURCE 3.3, and other fine software products. Additional information on Lazer's software products can be obtained by calling or writing Lazer MicroSystems, Inc.

Lazer Software Products:

ANIX:	$49.95
LAZER PASCAL:	$39.95
DISASM/65:	$29.95
p-SOURCE (book):	$19.95

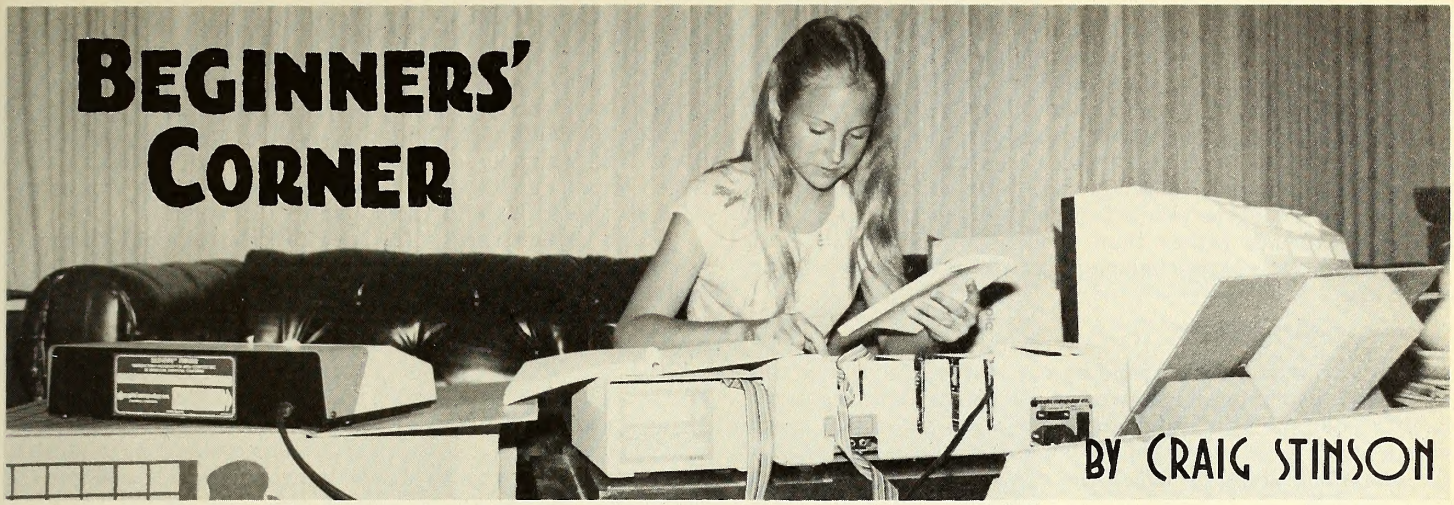
Other Fine Lazer Products:

Lower Case +Plus II	: $29.95
Lower Case +Plus	: $64.95
Keyboard +Plus	: $99.95
Character Set +Plus	: $24.95

UNIX™ Bell Labs.
CP/M™ Digital Research, Inc.
APPLE II™ Apple Computer, Inc.
ANIX™ Lazer MicroSystems, Inc.

Lazer MICRO SYSTEMS INC.
1791-G Capital
Corona, CA 91720
(714)735-1041

BEGINNERS' CORNER



BY CRAIG STINSON

If you were to examine the contents of your Apple's 65,536 memory addresses, you'd see an enormous string of eight-bit binary numbers that, taken in conjunction, constitute the program and data running within your computer at the moment of your inspection.

Program in this context is a broad term that includes the system Monitor, a Basic interpreter, some version of the Disk Operating System (assuming you have that loaded into memory), and, perhaps, an application program of your own making. The term *data,* in this context, means the binary representation of whatever numbers and words your program may be operating upon or producing for your edification.

Right Place at the Right Time. The computer doesn't distinguish between program and data except by context. You could say that it's all data to the 6502; and, just as with us, a bit of information may mean one thing in one setting and something else entirely in another. So it is with your computer. A given number at a certain memory position may cause the processor to perform some action; at another time and place it may be merely that which the processor processes.

So how do all these numbers get into the computer in the first place?

Some of them, like the ones that make up the system Monitor and the Basic interpreter, and like certain values stored in the low end of memory, are implanted automatically when you turn on your machine. The Monitor and interpreter are stored in ROM and are hence immovable residents of the upper extent of memory; certain instructions stored within the Monitor cause other values to be written into what's called the zero page of memory—the addresses from 0 to 255.

The Disk Operating System gets transferred into memory automatically, provided that you have a disk drive connected to the drive 1 pins of a controller card that's placed in one of your peripheral slots, and provided that that drive holds a disk that has DOS stored on it. Certain instructions in the Monitor cause the 6502 to scan the Apple's peripheral slots as soon as you turn the machine on, starting at slot 7 and moving downward. If a controller card is found, the 6502 is ordered to read the contents of the disk's outer three tracks and to transfer that information, by way of the drive 1 pins, into memory. Once within, DOS takes up quarters in the highest available area of RAM, which, on a 48K machine, is directly below the ROM addresses.

The Disk Operating System, on the Apple, resides in RAM, rather than ROM, so, unlike the Monitor, it can be altered—if you feel like doing that (see (DOSTalk, p. 33). In fact, altering the operating system is one means by which software publishers often make their disks hard to copy. In order to run the *Copy* or *Copya* programs on your System Master disk, you have to have the standard version of DOS loaded into memory. But if the disk you wish to copy was formatted under a slightly altered operating system, the Apple's copy programs, running under normal DOS, will not be able to read that disk, so you'll get an I/O error when you attempt to make the copy.

Speaking the Language. Whatever programs are in your machine—whether ROM-resident, loaded from disk, or whatever—have to have been entered manually at somebody's computer sometime. And almost without exception, the person who has done this work will have done it with the help of some kind of programming language.

We say "almost without exception" because it is theoretically possible to program certain computers directly in machine-readable binary code. Certain computers older than your Apple are equipped with switches on the outside of the cabinet by which a person can enter binary numbers directly into specified memory locations. Another switch on this kind of machine tells the computer to begin executing the program so stored.

If there is such a thing as programming in machine language, this kind of laborious procedure would be it.

On your Apple there is no way of entering binary numbers quite that directly. But you can come almost as close to talking to the processor in its native tongue by going into the Monitor and entering hexadecimal numbers.

To get an idea of what that kind of programming is like, do a *call −151* and enter the following:

```
300:A9 0C 20 A8 FC AD 30 C0 88 D0 F5 60
```

followed by a return. You've just written a tiny program in (almost) machine language. Exactly what you've done is enter a specific hexadecimal value at each of the twelve addresses from hex 300 to 30B.

If you now type 300G and hit return, you'll tell the processor to run the program starting at location 300. *Eureka,* the program sounds the familiar Apple chime. You can make the program repeat rapidly by typing 300 and bunch of Gs and then hitting return.

If you had a mind to do it, you could program a database manager, a word processor, or an arcade game in just this fashion—by entering two-digit hex values at the proper memory locations one by one, letting the Monitor translate them into their binary equivalents and using the Monitor's G command to make the program run. This would be monstrously difficult to do, however, for several reasons.

First, it's tough to remember which numbers mean what to the 6502. Second, even if you had all the instruction codes memorized, it would take you forever to program anything meaningful. Third, and most important, you'd never get it right the first time (programmers seldom do) and you wouldn't wish upon an enemy the task of correcting, or debugging, such a stream of undocumented hex code.

Hansel and Gretel Strike Back. Even if by some quirk you did manage to get something complex to run the first time, this sort of direct programming would still be a lousy way to go. If you ever wanted to change your program more than a day after you'd finished it, you'd find that going back into that code to figure out how it all worked was worse than being lost in the forest without any bread crumbs.

Programming languages were invented partly to let human beings command their computers with verbal symbols rather than merely with numbers and partly to provide a crumb trail to facilitate program debugging and maintenance.

Programming languages are often described as being high-level or low-level, with numerous gradations of altitude between. What makes a language high in level, by this way of thinking, is the degree to which it permits the programmer to command results rather than processes. For example, in Basic, a relatively high-level language, it is possible to write an order like

```
PRINT  TAB(12)  SQR(169)
```

and the computer will respond by calculating the square root of 169 and displaying its answer on the screen, starting at the twelfth column. You could accomplish the same end result with a lower level language, but you'd have to specify more of the individual steps by which the computer would carry out your request.

As a consequence of high-level languages being more result-oriented than process-oriented, they tend to bear more resemblance to English than low-level programming languages do. Just as we tend to think in terms of actions, like "walk across room," and leave to automatized internal processes the mechanical minutia of which muscles to contract when, so with a high-level language like Pascal we can use a command like *write* and let the computer worry about what individual steps it must take.

Suiting the Word to the Deed. Basic, Pascal, and other high-level languages commonly use words taken directly from English—or other so-called natural languages—although the manner in which they require natural words to be used may seem a little formal and stilted to us at first. Computers, at this stage of their development, are much more exacting in their syntactic requirements than any natural language.

Another consequence of the result-orientation of high-level languages is that they are sometimes developed for a specific category of application. Cobol, for example, is a language best-suited to business programming, while Pilot is ideal for developing tutorial dialogues for classroom settings. Fortran was written to facilitate mathematical and scientific computation. Basic was designed as a general-purpose high-level language, which is one reason why it has become the most commonly used language in the personal computer world.

At the low-level end of the spectrum is assembly language. In assembly language the programmer must manipulate each individual machine action, much as he would if he were programming directly in binary or hexadecimal code, but the procedure is vastly simpler than direct programming.

To begin with, in assembly language, instructions are specified in the form of mnemonics instead of numbers. A typical instruction would look like JSR $FDED, where the JSR is the instruction or opcode, telling the computer in this case to jump to a subroutine (a sequence of code located somewhere else in the program), and the $FDED is the argument, telling the computer where to jump to. To program that instruction directly in hex, you would store the values 20, ED, and FD at the appropriate three consecutive memory addresses.

Writing JSR $FDED may not seem a whole lot simpler than writing 20 ED FB, but it is some improvement. It's easier to remember what JSR means if you come back and look at your code later than it is to know what 20 means, particularly since that 20, out there in the middle of a stream of hex numbers, could just as easily be a piece of data as an opcode.

The second big advantage of assembly language is that it allows the programmer to identify certain addresses or certain segments of code by labels of his own choice. For example, $FDED is the starting address in the Monitor of the routine that handles the display of characters on your screen. If your program made reference to that set of Monitor code, as it very likely would, you could define $FDED as, say, COUT (for character output—the name chosen by the programmers who

southwestern data systems™

PRESENTS

NORAD

A HIGH SPEED STRATEGY GAME

NOT "JUST ANOTHER" ABM-LIKE GAME
NORAD'S WORLD MAP SHOWS ALL THE ACTION
"IN A TOTALLY FRESH, NEW WAY"
(According to THE SOFTALK REVIEW, December 1981)

FUNPAX™

NORAD is the first in the "FUNPAX" series of entertainment software from SDS and is available through dealers nationwide.

alert!

ENEMY MISSILES BEARING NUCLEAR WARHEADS ARE HURTLING TOWARDS THE UNITED STATES!

alert!

AUTOMATIC ABM GUIDANCE SYSTEMS HAVE FAILED AND MUST NOW BE CONTROLLED MANUALLY!

alert!

YOUR APPLE II* HAS DETERMINED THE I.C.B.M. TARGETS TO BE THE **TEN** MOST POPULATED CITIES IN AMERICA!

THE FATE OF MILLIONS RESTS IN **YOUR** HANDS!

alert!

YOUR INTENSE PERSPIRATION IS SOAKING YOUR CLOTHING...
DON'T JUST SIT THERE...
KEEP READING!

Strategic planning as well as quick reflexes are needed to survive! You must launch, control and detonate your interceptors from all ten launch sites!

If you hold out long enough, even Congress helps by approving mobile MX and submarine-based reinforcements!

Buy it. It's good!

$39.95

"NORAD is licensed by Western MicroData Ent., Ltd."

*APPLE II is a trademark of APPLE COMPUTER, INC.

wrote the Monitor) or as any other name you liked, and then just call the routine by name instead of by number. If you've been trying to handle 20 ED FD or even JSR $FDED, JSR COUT starts to look downright friendly.

The third big advantage of assembly language over direct programming is that it allows you to put comments into your program. Comments are the crumb-trail that tells you, or some other programmer who needs to decipher your code, why you did what you did—in plain English. There are no spelling or syntax rules governing what you put in a comment.

Programming in assembly language is essentially a two-step process. First you write what's called a *source code,* which includes your label definitions, your instructions, and your comments. Second, another program, called an *assembler,* translates your source code into something the computer can read and execute; that something is usually known as *object code.*

Most assemblers are equipped to store both your source code and the resultant object code as separate files on disk. That way you can run the object code and still have the source listing for purposes of debugging and subsequent alteration. The results of the assembler's labor, the object code, are just what you would get if you programmed directly, through the Monitor; JSR COUT gets broken back down into 20 ED FD, and so on.

This brings up a point of nomenclature. You'll frequently see programs written in assembly language described as being written in machine language. This is technically incorrect. The programs in question are written in assembly language and translated by an assembler into machine language.

Moreover, everything that runs on your computer, whether originally coded in Pascal, Basic, Fortran, or Algol, must run in machine language; the 6502 responds to nothing else. The only kind of true machine language programming would be, as we said earlier, the direct entry of binary numbers into specified memory locations by way of external switches.

It's a minor point, perhaps, but it may contribute to your understanding of what's going on. Programs labeled "machine language" were written, presumably, in assembly language. The import of that label to you is that neither Basic interpreter is required, so you can run such a program on either an Apple II or a II Plus.

The Source of the Object. It may occur to you to wonder, since source code has to be translated by an assembler into object code, how one writes an assembler program. One way to do it is to use someone else's assembler. Another way, of course, is to punch the code in barehandedly, number by number, through the Monitor. There are also such things as cross-assemblers, by which code originally written for some other computer (with the help of its assembler) can be converted into 6502 object code.

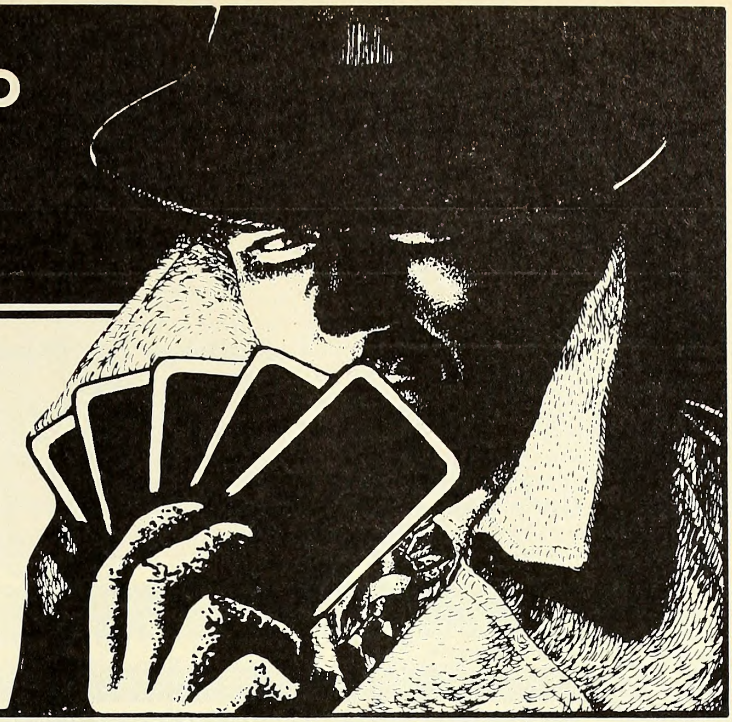
Here's a sample of what an assembly language listing looks like. This is a small portion of the System Monitor (the autostart version), as listed in the *Apple II Reference Manual,* pp. 136 to 154.

```
FB9F:  20 97 FB  53          JSR  ESCOLD  ; DO THIS CURSOR MOTION
FBA2:  20 0C FD  54          JSR  RDKEY   ; AND GET NEXT
FBA5:  C9 CE     55 ESCNEW   CMP  #$CE    ; IS THIS AN N?
FBA7:  B0 EE     56          BCS  ESCOLD  ; N OR GREATER DO IT
FBA9:  C9 C9     57          CMP  #$C9    ; LESS THAN I?
FBAB:  90 EA     58          BCC  ESCOLD  ; YES SO OLD WAY
FBAD:  C9 CC     59          CMP  #$CC    ; IS AT A L?
FBAF:  F0 E6     60          BEQ  ESCOLD  ; DO NORMAL
FBB1:  D0 E8     61          BNE  ESCNOW  ; GO DO IT
```

Actually what you see here is the source code side by side with the object code it generates; this is the way that assembly listings are commonly printed in books and magazines. The first four columns on the left represent object code; the remainder is source code.

The first column of the source code, the one that in this example contains the numbers 53 through 61, represents the current line number; a program would normally begin with line 1 and proceed upward by consecutive whole-number steps.

Graphics are no mystery when you hold...



...The Winning Card

Computer Station™ Presents
STATION MASTER™
for the Apple II® or Apple II Plus®

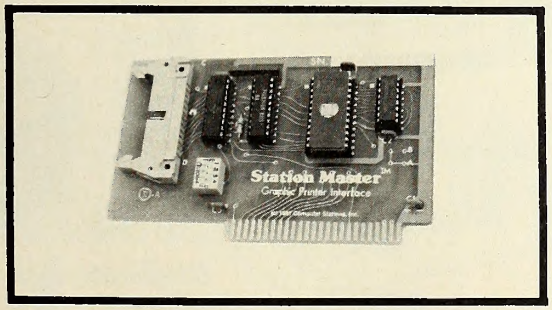
Computer Station has long been known for its graphic developments for the Apple Computer. The Dithertizer II™ is a video digitizer board which loads the hi-res screen with a video camera. Combined Enhanced Graphics Software (CEGS) dumps the contents of the hi-res screen onto paper for a large variety of printers by means of a low-cost software package.

We now introduce the Station Master™. The Station Master is a universal parallel card with the added capability of being able to put the hi-res screen on paper with simple keyboard commands. The Station Master may be used with ANY one of FOUR different printer types. (See model selection below.)

As with our CEGS, there are options for dumping page one or page two, normal or expanded size, picture or plot, and horizontal positioning.

The Station Master card is supplied with printer cable so you just plug in and go. We also include a diskette with practice pictures and plots as well as drivers for use with Visiplot™.

The Station Master offers compatibility with BASIC, Pascal 1.1. CP/M and application software. The documentation includes helpful information for first-time users about hi-res graphics and how to use the board with progrms such as Applewriter™, Apple Plot™, Visicalc™ and Visiplot™.



The Station Master including cable - **$175.00**

Model #7017 accommodates ANY of the printers listed below.

- Epson MX-80 with graphics option
- Epson MX-100
- Anadex 9501/9500
- Datasouth DS 180
- Centronics 739
- NEC PC8023

Computer Station

11610 Page Service Drive St. Louis, Missouri 63141

(314) 432-7019

Direct orders will incur $3.00 shipping and handling charge and sales tax where applicable.

Hard Disk Storage for your System *PLUS* Your Answer to Backup.

New! 5 or 10 MByte, 5¼" hard disk, and 10 MByte Removable 8" Winchester Cartridge.

Here's a cost effective, large capacity storage alternative for your Apple III or II, and IBM Personal Computer, to name just a few.

Use any CP/M® program with your IBM and the SCS series storage systems.

Or add Microsoft's Softcard to your Apple and run any CP/M program.

Storage for Most Computers

In addition to Apple and IBM, this same low-cost system interfaces to the most popular computers, including the TRS-80 series, Altos, S-100 systems, LSI-11, Multibus, and Motorola 6800.

Technical Excellence

Our Error Correction Coding (ECC) gives you a unique internal hard disk backup system, so media errors are corrected and become transparent to your computer. These advantages assure you of reliable performance for years to come:

- Reliability 10,000 hours MTBF.
- No preventative maintenance required.
- 2 hours MTTR
- Expandable through daisy chaining

Features	Apple ProFile	Corvus 5¼"	SCS-5 5/10 MByte	SCS-10R 10 MByte
Integrated Backup	None	None	5¼" Disk w/1 MByte Floppy	10 MByte Removable Disk built in
Error Correction	No	No	Yes	Yes
Capacity Formatted	5 MBytes	5 MBytes	5 or 10 MBytes	10 MBytes
Avg Seek Time	95 ms	125 ms	70 ms	60ms
Suggested List	$3495	$3750	$3495/$3995 *Add $550 for backup floppy*	$7310

Backup Storage a Dilemma?

We have the solution with the SCS-5/F. It's our combination 5¼" hard disk *and* 1 MByte 5¼" backup floppy in one compact cabinet.

No one but Santa Clara Systems answers backup problems with flexible storage solutions—not even Apple or IBM.

Supports Systems and Applications Software including CP/M.

For Apple, run DOS 3.3, SOS 1.1, Pascal, all CP/M programs, plus DB Master, Compu-Law, Accounting Plus, Peachtree Accounting, SuperCalc, Medical Management, Stockbroker, Real Estate Manager, Word Star, and Farmplan. For IBM use IBM DOS or CP/M 86. On TRS-80 run TRS-DOS and Oasis. CP/M runs on all S-100 and Multibus systems. And use RT-11 3B & 4 on your LSI-11.

Shipped Complete

It's all there: 5¼" Winchester, 8" removable hard disk subsystem, or combined hard disk and floppy backup, sophisticated controller, host adapter, operating software, power supply, cable, cabinet, and easy-to-follow operating manual.

Risk Free

You may return any SCS model storage system if the workmanship is not what we guarantee. That's how certain we are that the SCS-5, and SCS-10R systems are the best hard disk subsystems you can own.

Santa Clara Systems produces the largest selection of storage systems available. If you have storage problems, call us for the solution.

Shown below: Removable 8" 10 MByte Winchester cartridge drive.

Shown below: 5MByte 5¼" hard disk with floppy backup.

SAVE UP TO $200

Offer valid until 4/30/82.

I'm interested. I want to save ☐ $100 on the SCS-5 ☐ 5 ☐ 10 MByte with 1MB floppy backup ☐ $200 on the SCS-10R

My check is enclosed for $ _____ (Calif. res. add 6.5% sales tax). Checked below is the host adapter I require (check one only. Additional host adapters $550).

☐ IBM ☐ Apple II ☐ Apple III ☐ TRS-80 I ☐ TRS-80 II ☐ TRS-80 III ☐ S-100 ☐ Multibus ☐ LSI-11 ☐ Motorola 6800 ☐ Altos

☐ Please send more information. ☐ Have your salesman call me.

☐ Where is my nearest dealer?

Name _____ Phone _____

Title _____ Company _____

Street _____

City _____ State _____ Zip _____

Mail today to:

SANTA CLARA SYSTEMS, INC. • 560 Division Street • Campbell, California 95008

Santa Clara Systems, Inc.

560 Division Street • Campbell, CA 95008

(408) 374-6972

The next column to the right, the one in which **ESCNEW** appears on line 55, is a place for the programmer to label the current line. Other instructions in the program will be able to refer to a labeled line by its name.

The third column in the source list holds the three-letter opcode mnemonic. There are fifty-six different elemental operations that the programmer can ask the 6502 to perform (including one that tells it to do nothing but mark time for an instant); each one has its own command mnemonic. Together these fifty-six mnemonics constitute what's called the instruction set of the microprocessor.

The column to the right of the mnemonics holds the arguments, or operands. These are the items of information upon which the 6502 is instructed to act. The argument field can hold one of three things: a value, an address, or a label (by which the processor finds its way to an address). Or this column may be blank; certain instructions—like **INX**, which tells the 6502 to increment by one the value currently held in the **X** register, do not require an argument. In our sample listing, the arguments in lines 55, 57, and 59 are numeric values; the others are all labels.

Finally, the column on the right, the one that starts with a semicolon, contains the programmer's comments. These are notes to himself and other programmers; the assembler will ignore them when it converts the source file to object code.

The four columns on the left in our listing represent the object code. The first column, always terminated by a colon, holds addresses. The second column holds opcodes, and the third and fourth columns are for the operands. Note that certain opcodes require two bytes of operand, while others take only one. Still other opcodes, not shown in this list, would appear by themselves, with no operand following.

If you have an Apple with the autostart version of the system Monitor (an Apple II Plus or a standard Apple II equipped with an Apple Language System), you can go into the Monitor by typing *call −151* and verify the accuracy of our listing. Type FB9F.FBB2 from the asterisk prompt, and when you hit return the Monitor will respond by printing the values stored at the designated range of memory addresses. If you compare them to our list, you'll see they match. If your Apple has the so-called old Monitor ROM—the one that doesn't automatically boot your disk drive when you turn the power on—then your listing will differ entirely. The complete listings for both versions of the Monitor are published in Appendix C of the *Apple II Reference Manual.*

Try one more experiment while you're in the Monitor. Type FB9FL and hit return. The Monitor's L command produces what's called a *disassembled list.* This process is somewhat the opposite of what an assembler does. In disassembling, the Monitor tries to translate the values held in a range of memory addresses into source code. Where the range of memory in question holds some kind of executable program, the disassembly will produce something resembling a programmer's source code, although it will not show any line numbers, labels, or comments. If you try to disassemble some other range of memory, however, like an area that holds the last letter you wrote on your word processor, the Monitor will most likely produce a lot of question marks in the mnemonic field; that's its way of saying, "This does not compute."

We Never Promised You a ROM Garden. If by now you've drawn the conclusion that assembly language programming is an arcane art, practicable only by whiz kids and madmen, you're not alone in your judgment. Few would deny that it's more difficult than programming in Basic or Pascal.

There are, however, compensating advantages to assembly language. The one most frequently cited is speed; programs written in assembly language usually can be executed by the computer at a faster pace than programs written in Basic or other high-level languages.

Two factors account for the greater speed of assembly language programs. First, at the time they are run they don't have to be converted into executable binary code; they're already there. Programs in both Basic and Pascal have to be interpreted as they're run—translated line by line—into machine code. Second, the fact that the assembly language programmer has to specify each individual action of the microprocessor, while it makes programming more complex, also affords a greater degree of control. By writing in assembly language, a programmer can both minimize inefficiency in the code and accomplish certain things that would be awkward or even impossible using the more general-purpose routines provided by high-level languages.

Programs that need a lot of quick graphic action, like most games, are almost always written in assembly language. For many other kinds of programs, like those that go to disk frequently or that require a lot of input from the user, execution speed may be less critical, and so there may be no reason for a programmer to forego the amenities of a higher-level language.

Next month we'll begin looking at some of those amenities. ■

COLLEGE BOARD SAT* PREP SERIES

TRS-80, APPLE, PET, OSI, ATARI, CP/M, PDP-11

Each program confronts the user with a virtually limitless series of questions and answers. Each is based on past exams and presents material on the same level of difficulty and in the same form used in the S.A.T. Scoring is provided in accordance with the formula used by College Boards.

S.A.T., P.S.A.T., N.M.S.Q.T. — Educator Edition set includes 25 programs covering Vocabulary, Word Relationships, Reading Comprehension, Sentence Completion, and Mathematics. **Independent tests of S.A.T. series performance show a mean total increase of 70 points in students' scores.** Price **$229.95**

GRADUATE RECORD EXAM Series - Educator Edition includes 29 programs covering Vocabulary, Word Relationships, Reading Comprehension, Sentence Completion, Mathematics, Analytical Reasoning and Logical Diagrams. Price **$289.95**

COMPETENCY PROFICIENCY EXAM PREP SERIES

This comprehensive set of programs consists of simulated exam modules, a thorough diagnostic package, and a complete set of instructional programs. It is designed to teach concepts and operations, provide drill and practice and assess achievement levels through pre and post testing. The Competency Exam Preparation Series provides a structured, sequential, curriculum encompassing mathematical, reading and writing instruction.

This program is designed for individual student use or use in a classroom setting. Programs provide optional printer capability covering worksheet generation and performance monitoring. C.E.P.S. are available in three software formats.

Special editions available for California Proficiency Assesment Test and New York Regents Competency Tests. Call for Prices.

Krell's M.I.T. Logo for Apple* $179.95

Complete 4 disk set includes Library Disk, 20 program Tutorial Disk, **Alice In Logoland** and **2 COPIES OF M.I.T. LOGO FOR APPLE***

Odyssey In Time

This spectacular adventure game adds a new dimension of excitement and complexity to **Time Traveler.**

Odyssey In Time includes all the challenges of **Time Traveler** plus 10 additional eras. Each game is different and may be interrupted and saved at any point for later play. **$39.95**

Time Traveler

The best of the adventure games. Confronts the player with complex decision situations and the demand for real time action. Using the Time Machine, players face a challenging series of historical environments. To succeed you must build alliances and struggle with the ruling power. Each game is unique. **$24.95**

Isaac Newton +

Perhaps the most fascinating and valuable educational game ever devised - **ISAAC NEWTON** challenges the players (1-4) to assemble evidence and discern the underlying "Laws of Nature" that have produced this evidence. **ISAAC NEWTON** is an inductive game that allows players to intervene actively by proposing experiments to determine if new data conform to the "Laws of Nature" in question. Players may set the level of difficulty from simple to fiendishly complex.

F.G. Newton

Full Graphics Newton. This version of Isaac Newton presents all data in graphic form. Because data is graphic rather than symbolic, this game is suitable for very young children. Players may select difficulty levels challenging to the most skilled adults. **$49.95**

☆ NEW ☆ ☆MICRO-DEUTSCH☆

Micro-Deutsch set includes 24 grammar lessons, covering all material of an introductory German course. Four test units also included. Grammar lessons use substitution transformation drills, item ordering, translations and verb drills. Drill vocabulary based on frequency lists. Suitable for use with any high school or college textbook. Extensively field tested at SUNY Stony Brook. Available for Apple II and PET/CBM. (PET version includes a special foreign language character chip.) Also available soon: MICRO-FRANCAIS, MICRO-ESPANOL, MICRO-IVRIT, MICRO-YIDDISH, MICRO-CHINESE, MICRO-JAPANESE. **$179.95**

☆ NEW ☆ ☆Pythagoras and The Dragon☆

Mathematics in a fantasy game context. Based on **The Sword of Zedek, Pythagoras and The Dragon** introduces Pythagoras as a mentor to the player. When called on for aid, Pythagoras poses math questions, and depending on the speed and accuracy of the player response, confers secret information. With Pythagoras as an ally, the quest to overthrow Ra, The Master of Evil, assumes a new dimension of complexity. Depending on the level chosen, problems range from arithmetic through plane geometry.

32K **$39.95**

Free Bonus with purchase of $300.00 or more: Applesoft Tutor Series

Krell Software Corp. has no official ties with the College Entrance Examination Board or the Educational Testing Service. Krell is, however, a supplier of products to the E.T.S.

PROGRAMS AVAILABLE FOR
TRS-80, APPLE II, PET & ATARI

N.Y.S. residents add sales tax.

All programs require 16K ● TRS-80 programs require LEVEL II BASIC ● APPLE programs require Applesoft BASIC

*Trademark Apple Computer Corp.

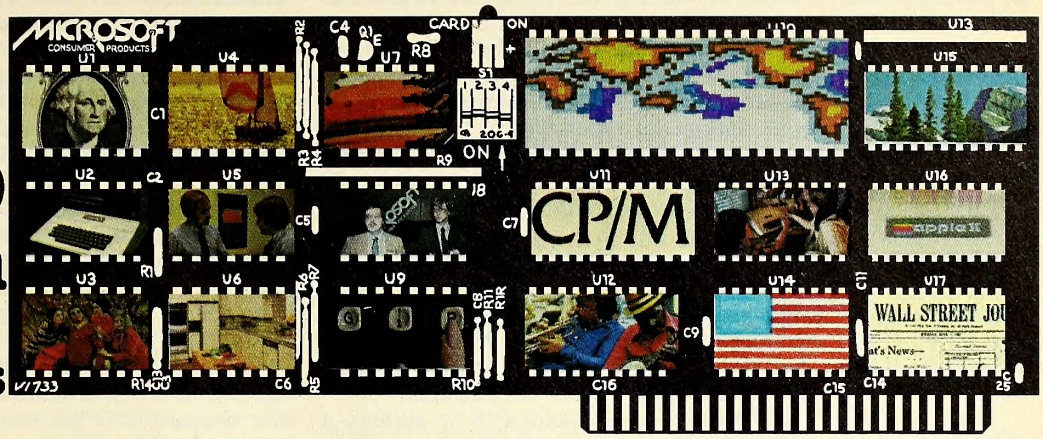
KRELL SOFTWARE CORP.

"The State of the Art in Educational Computing"

Dept. C, 21 Millbrook Drive, Stony Brook, N.Y. 11790 (516) 751-5139

SOFTCARD Symposium

by Greg Tibbetts



Over the past couple of months, these columns have been somewhat technical. As something of a break, this month we are going to continue with our discussion of the utilities, but our subjects will be the miscellaneous utilities—FORMAT.COM, COPY.COM, RW13.COM and CPM56.COM. Beginning next month, we'll go on to tackle more complex ones, such as ED and ASM.

The four programs just mentioned can be classified as maintenance utilities. They are the workhorses of the system, and FORMAT and COPY, at least, will come into play constantly. Those of you who still have access only to thirteen-sector disks will not be familiar with the RW13 and CPM56 programs. If this applies to you, it is recommended that you upgrade your system to sixteen sectors via the DOS 3.3 upgrade kit available at your dealer. As more and more companies discontinue thirteen-sector support, software usable in that format will become harder and harder to find. In any case, the additional disk storage space available with the sixteen-sector format is worth the price by itself.

Before we proceed, a further recommendation is in order. The 16K ramboards available now are also a good value. While their use with Apple DOS is pretty much restricted to giving you the alternate language—Applesoft or Integer Basic, whichever you don't have in ROM—with CP/M they give you additional memory that you can use not only with .COM files but with MBasic and GBasic as well. Although their cost is certainly not negligible, they return a sizeable increase in the usefulness of your system. In many instances the additional memory will make the difference between being able to run a program or not.

The first utility we will examine is FORMAT.COM. This program is the general-purpose disk formatter and is similar to the *init* command used under Apple DOS. When you buy a disk, you get it in an unformatted condition. What this means is that there is no information stored on the disk that your system is capable of reading. Basically, there are two types of information stored on a disk by the system.

The first type is obviously the data that you yourself have instructed the system to store. As we've discussed in past columns, the Apple CP/M system handles disk data in 256-byte blocks called sectors. In reality, because of the way the information is encoded to be written on the disk, your 256 bytes actually take up more than that number of bytes when stored there. Since this process is essentially invisible to us, we'll ignore it and simply say that the 256 bytes we put out there in a sector comprise the data area for that sector. The second type of information the system stores on the disk is of a housekeeping nature. It is the information the system needs to locate the sector—and hence our data—when we call for it. This information is comprised of several things: the address field that contains the track, sector, and volume numbers; certain gaps between sectors that are used as buffer zones so that the next sector will not get overwritten when we rewrite the current one, as might happen if the disk drive were running slightly faster than before; and, finally, some special bytes called *sync bytes* that the drive electronics uses to insure that it is in exact synchronization with the bytes stored in the address and data fields. In reality, then, for every byte of data stored on the disk, several bytes of housekeeping information are also stored.

Obviously, this housekeeping information must be written first for the entire disk, so that the system can find the proper sector when it needs to store a block of data. This is the job done by the formatter or initializer program. These programs or routines usually take care of initializing the directory; certain of them, for example, the Apple DOS *init* command, put the operating system on the disk as well. FORMAT, like its counterparts, sets up each of the thirty-five tracks of the disk with the address information for each sector. In addition, it also writes hexadecimal E5 bytes into every sector of the disk as data. These bytes are really only necessary for the directory portion of the disk and even then are only really needed in a few bytes of the directory. By writing the entire disk however, FORMAT has a chance to check each of the sectors for validity of the address information; it will report an I/O error if there are any media defects. No attempt is made by FORMAT to do a complete media verification, however.

The syntax for FORMAT is either:

FORMAT
or
FORMAT d: (where d is a valid system drive designator)

In the case of the first syntax above, FORMAT will prompt the user for the drive designator and when finished, will prompt again for another drive. If a return only is given at the prompt, the program cancels and returns to the CCP (remember, this is the Console Command Processor, responsible for the CP/M prompt). In response to the second syntax shown, FORMAT will do the format on the disk indicated and return immediately to the CCP. In either case, the program prompts the user to insert the blank disk to be formatted in the selected drive and, before beginning the process, checks to see if the disk has already been formatted. If so, it returns a warning that the disk contains data and prompts the user to go ahead or abort the format operation. Whenever a format is done, the FORMAT program always erases a track before formatting it. This is to insure that nowhere in those gaps we mentioned is there a chance that a valid address mark or sync byte field still exists that might cause the drive to read data erroneously.

FORMAT checks for proper drive designators and will attempt to verify that the drive named is actually connected to the system. As we've mentioned before, CP/M is aware of whether a disk controller card is in the appropriate slot for the corresponding drive, but it has no way of verifying if there are two drives connected to the card. Consequently, FORMAT will produce an error message for nonexistent drives in either situation.

The second program we'll discuss is COPY.COM, the disk copy utility. COPY fulfills several functions. It is the full disk

copier; it is the method of placing CP/M on the first three tracks of a disk (replacing SYSGEN found on most other systems); and, finally, in conjunction with DDT.COM, it is the method that Microsoft has used to enable owners to install changes in their CP/M (replacing MOVCPM's similar function on other systems).

COPY requires that the destination disk be newly formatted or that it previously contained data (used with an Apple system, of course). By far the safest method is to format a disk before copying to it, but if you like to live dangerously COPY will properly transfer the contents of the source disk to the target disk, whether the target was a previously used CP/M disk or an old Apple DOS disk. Since each and every sector of each and every track is transferred with COPY, it does not matter what information was previously there, as it will be entirely overwritten. The safest way, however, is to reformat.

Like FORMAT, COPY may be invoked with or without a command line as follows:

```
COPY
```

or

```
COPY d1:=d2:
```

(where d1 and d2 are drive designators and may be the same drive)

In the former case, the COPY program will prompt you for a response to indicate the source and destination drives. In this case, the source and destination are then typed, separated by an equals sign (=). As with all CP/M utilities, the equals sign acts just as it does in Basic; that is, the quantity (in this case the data) in the symbol on the right is placed into the symbol on the left. Destination is therefore on the left and source on the right. At this point, COPY will prompt you to insert the disk into the named drives and hit return. When you do, the copy procedure will commence. If only one drive is being used, COPY will prompt you to insert the appropriate disks into the drive at the proper times.

In addition to copying the entire disk, COPY also allows you to copy only the three system tracks onto the target disk. This allows you to put CP/M on the disk and make it bootable. A disk with CP/M on it is called a *system disk*, for obvious reasons. This is done using a /S after the destination and source drive names. There should be no spaces in this command line. Unlike Apple DOS, CP/M does not have the capability of using the first three tracks for the storage of data and therefore no storage space is lost by putting the system on your disk. *Note*, however, that your copy of CP/M is serial numbered and registered with Digital Research. Any disk that you use to give your own programs to friends and associates must not contain your serial-numbered CP/M. To do so is a violation of the license agreement you have with Digital Research and with Microsoft. If the programs you wish to give are on a system disk, use the PIP utility to put them on a nonsystem-formatted disk. With COPY, there is no way to avoid copying the system if the original disk contains it.

The next utility, RW13.COM, is a program to allow sixteen-sector systems to read and write thirteen-sector disks. With that in mind, this discussion and the program itself will be of limited use to those of you with thirteen-sector systems. Its actual purpose is twofold: first, it allows people with thirteen and sixteen sector systems to exchange their text files and programs with a minimum of effort; second, it allows companies to distribute certain software in a thirteen-sector single format, knowing that both types of systems can use it—either directly or by converting it with RW13 and PIP. From the user standpoint, the program simply assigns one drive in the system to be the thirteen-sector drive, and from then on, all disk reads and writes with that drive will be in thirteen-sector format.

Functionally, the program is a self-relocating image of the thirteen-sector RWTS (Read-Write-Track-Sector) routines. On execution, it loads into memory at the normal TPA (Transient Program Area), starting at 100 hex, and then it moves itself to high memory just below the CCP. The vectors for the appropriate drive are patched to point to these routines instead of the standard disk drivers. RW13 also alters the address field of the jump BDOS instruction at location 05H (location 06H is the low-order byte, and 07H the high-order byte; all are Z80 addresses), to point to its own beginning. This jump instruction and the corresponding address field are what tell programs the location of the beginning of BDOS and consequently the top of the user program memory. In this way, RW13 effectively protects itself from inadvertent overwriting by any programs running in the system. In order to keep the BDOS calls functional, the old address field from locations 06H and 07H is placed into a jump instruction in the first three bytes of RW13. The address that is patched into the disk vector to point to RW13 takes this into account and actually points to the fourth byte in the RW13 routine.

Only one drive may be converted to thirteen-sector at any one time, and this drive may not be the boot drive (drive A:). To invoke RW13 the command is:

```
RW13 d:
```

The system will respond with "Drive d: converted to 13 sector operation". Attempts to make another drive thirteen sector while one is already converted will result in the error message, "Must RW13 X: first". This is the format for cancelling the RW13 command, and is typed as follows:

```
RW13 X:
```

The system will respond with "Drive d: returned to 16 sector operation".

The final program on our list to be examined is CPM56.COM. This utility is the means used to increase your available program memory by taking advantage of the 16K ramboard. It obviously cannot be used unless a ramboard is installed in the system. As shipped, the version of CP/M included on your sixteen-sector disk is designed for 44K of available RAM memory. Note that if your Apple is standard and has 48K of RAM, only 44K are available for use by CP/M. The other four are needed for the Apple zero page scratchpad locations, the 6502 stack, and the Apple screen memory area.

Of the 44K usable, the combination of BDOS and BIOS—or CP/M itself, in other words—takes approximately 5K, leaving about 39K for user program space. This is somewhat better than the approximately 36K available under Apple DOS, but since the CP/M Basics must also reside in RAM, the area for user Basic programs becomes somewhat limited.

Using a ramboard and CPM56, the additional 12K of user RAM available on the ramboard can be used for programs, bringing the total to 56K and the user available space to 51K. The additional 4K unaccounted for is unfortunately needed in the Apple system to take care of memory-mapped I/O, the method used to address the ROMs on cards in the Apple slots. Even so, this is a significant increase, both in space and in the usability of the system for serious programming purposes.

CPM56.COM is basically another version of CP/M designed to run at a different and higher address in memory. In this way it is analogous to the program MOVCPM, which in many other systems enables you to move CP/M's operating location in memory so that you can adjust it for the particular size system you are using. Such a wide variation was not available on the Apple, so there was no need to provide other than the two versions. CPM56 can be invoked by the following command:

```
CPM56 d:
```

(where d is the drive designator you wish the new system to be written to.)

The system will prompt you when you are to insert the disk to be written to in the drive you selected.

Be aware that the two versions are mutually exclusive to the extent that you cannot place a 56K disk in the boot drive and do a warm boot if you are currently running 44K, and vice versa. To do so will cause the system to hang, requiring a cold boot to recover. You should also be aware that many programs perform the equivalent of a warm boot at their conclusion and that the effect of this will be exactly the same. The

disk and file formats of course are identical and, with the exception of the above, can be interchanged at will.

This completes our discussion of the miscellaneous utilities. Before we leave the subject, however, we will talk about a utility that you can enter at your keyboard that will also become a valuable utility for you to have at your disposal. The following program, entitled BOOT.COM, is a general-purpose reboot program that, in essence, does the equivalent of a cold boot on the controller card in slot 6. In addition, it also gives you the capability of booting either thirteen-sector or sixteen-sector disks from a sixteen-sector system. It is supplied in hex dump form, and is entered as follows:

Procedure for Entering BOOT.COM

1. Invoke DDT.COM by typing simply DDT when at the A> prompt.
2. When DDT signs on and gives you its prompt, a minus sign (−), type S100 followed by a return. DDT will respond with the address 0100, a space, and the contents of that address as a two-byte hexadecimal number; then it will leave the cursor one space to the right, awaiting your entering a new value for that location.
3. Each time you enter a value followed by a return, or just a return, DDT will display the next address and its contents. In the former case, it will have replaced the current contents of the last address with the one you typed, and in the latter case, it will have left them as is. Typing a period (.) followed by a return will cancel the S command and return you to the DDT prompt.
4. As you have no doubt surmised, your mission (should you decide to accept it) is to take the bytes from the hex dump that follows and, beginning at 100 hex, enter them all, using DDT and its S command. Once done, you should be very thorough in checking your work, either by going back through with the S command and just hitting returns or by using the command D100 to generate a table to compare to the one that follows.
5. When you have finished and are sure the data has been entered correctly, insert the disk you wish to save the program to in the currently logged drive and type control-C. When the CP/M prompt returns, type SAVE 1 BOOT.COM. This completes the procedure for installing BOOT. Accuracy cannot be stressed strongly enough, since erroneous bytes in the program may have any number of nasty side effects.

BOOT.COM

```
0100  11 2F 01 0E 09 CD 05 00 3E 77 32 0B 00 0E 01 CD
0110  05 00 21 60 01 11 00 50 01 00 01 ED B0 21 00 60
0120  FE 33 28 02 26 C6 22 D0 F3 2A DE F3 C3 0B 00 0D
0130  0A 0A 0A 3C 33 3E 3D 31 33 20 73 65 63 74 6F 72
0140  2C 20 3C 43 52 3E 3D 31 36 20 73 65 63 74 6F 72
0150  3A 20 24 A9 A9 8D 1F 03 A9 60 8D 20 03 4C 01 03
0160  A2 20 A0 00 A9 03 85 3C 18 88 98 24 3C F0 F5 26
0170  3C 90 F8 C0 D5 F0 ED CA 8A 99 00 08 D0 E6 20 58

0180  FF BA A9 60 48 A9 0C 0A 0A 0A 85 2B AA A9 D0 48
0190  BD 8E C0 BD 8C C0 BD 8A C0 BD 89 C0 A0 50 BD 80
01A0  C0 98 29 03 0A 05 2B AA BD 81 C0 A9 56 20 A8 FC
01B0  88 10 EB A9 03 85 27 A9 00 85 26 85 3D 18 08 BD
01C0  8C C0 10 FB 49 D5 D0 F7 BD 8C C0 10 FB C9 AA D0
01D0  F3 EA BD 8C C0 10 FB C9 B5 F0 09 28 90 DF 49 AD
01E0  F0 1F D0 D9 A0 03 84 2A BD 8C C0 10 FB 2A 85 3C
01F0  BD 8C C0 10 FB 25 3C 88 D0 EE 28 C5 3D D0 BE B0

0200  BD A0 9A 84 3C BC 8C C0 10 FB 59 00 08 A4 3C 88
0210  99 00 08 D0 EE 84 3C BC 8C C0 10 FB 59 00 08 A4
0220  3C 91 26 C8 D0 EF BC 8C C0 10 FB 59 00 08 D0 8D
0230  60 A8 A2 00 B9 00 08 4A 3E CC 03 4A 3E 99 03 85
0240  3C B1 26 0A 0A 0A 05 3C 91 26 C8 E8 E0 33 D0 E4
0250  C6 2A D0 DE CC 00 03 D0 03 4C 53 11 4C 2D FF 00
0260  0B
```

To invoke BOOT, simply enter its name followed by a return. The program will prompt you with a command to type either a 3 to boot thirteen sector or a return to boot sixteen sector.

Until next month . . .

THE THIRD BASIC

by

Taylor Pohlman

Exploring Business Basic, Part 8

Greetings, Basic fans. This month's column will be long on content and somewhat short on the usual verbiage and explanations. This is because of the interest that last month's column stirred up. As the faithful among you will recall, we began a fairly simple discussion of the graphics capability of the Apple III and the specific workings of the .GRAFIX driver and the BGRAF invokable module. If you haven't read that discourse, it would be a good idea to get a copy before tackling the treatise below. If that's not possible, then get a firm grip on your Basic and *Standard Device Drivers Manual.*

Last month's article had as its main feature a program to draw circles and arcs efficiently, using the line-drawing capability of BGRAF. After some examination (and some comments!), that routine could use some tweaking. To see what can be done and to bring yourself up to date, refer to program 1 for the initialization and circle-draw subroutines:

```
894   REM circle drow subroutine
895   REM  r=rodius, scolefoc=ospect rotio * relotive
      density
896   REM xcen= x coordinote of center
897   REM ycen= y coordinote of center
900   xscole=r*scolefac
905   xvol=xcen+.5:yvol=ycen+.5
907   density=(mode=2)+2*(mode<2)+
      3*(mode=3)
910   firstx=xcos(0)*xscole+xvol
915   PERFORM moveto(%firstx,%yval)
920   stepomt=INT(20*(5-density)/r)+density
930   IF stepomt>6 THEN stepomt=6
940   FOR i=stepomt TO 119 STEP stepomt
950     PERFORM lineto(%(xcos(i)*xscole+
        xvol),%(ysin(i)*r+yvol))
960     NEXT i
970   PERFORM lineto(%firstx,%yvol)
980   RETURN
995   REM Initiolize grophics ond tables
996   REM
1000  OPEN#1,".grafix"
1010  INVOKE".d1/bgraf.inv"
1020  DIM xcos(119),ysin(119),xdot(3),
      srch%(20,3)
1025  vol=6.2832/120
1030  FOR i=0 TO 119:xcos(i)=COS(val*i):
      ysin(i)=SIN(vol*i):NEXT i
1040  xdot(0)=280:xdot(1)=280:xdot(2)=
      560:xdot(3)=140
1050  orotio=1.3
1055  zip%=1
1060  PERFORM initgrofix
1070  RETURN
```

Program 1

Those readers from last time will notice that to improve the symmetry of the circles, the cosine and sine tables (line 1030) have been adjusted to 120 points, which happens to divide each quadrant of the circle into equal parts (given a maximum stepsize of 6, as used in line 930).

Rather than go into any further detail about these routines, we'll plunge into the material, which will make good use of this stuff later.

The major input from last time was to take the skeleton program and make it into something useful that would show off more of the graphics capabilities of the Apple III. One of the most reasonable approaches would be to expand the set of functions in the program and build a beginning graphics editor—which you could then expand to your heart's content. The actual functions of such a program are easy to define. We have already implemented circles, and to that we can add lines, points, rectangles, text, the ability to fill in areas easily, the ability to erase objects and points, and, finally, the ability to store images on disk for later recall. To this list you can add lots of other features of your own design, within the framework that will be described. The key to all this is the power of the BGRAF module. The benefits (an important word in marketing parlance) range from drawing up organization charts to creating interesting cartoons. So let's get to it!

The first thing needed for a screen editor is some way to locate the cursor. Since it will be difficult to distinguish the cursor from any arbitrary dot on the screen, it will be convenient for it to blink. To do that we need a way to alternate the colors of the dot that will be our cursor while waiting for input from the user. The interrupt capabilities of the Apple III really come in handy here, because Basic implements the *any keypress event* with SOS. This causes an interrupt to occur in Basic, to which your program can react. The following short program illustrates:

```
1   ON KBD GOTO 10
5   PRINT".";
6   GOTO 5
10  OFF KBD
30  IF KBD=13 THEN STOP
40  PRINT KBD
45  ON KBD GOTO 10
50  RETURN
```

This program will print lots of dots, stopping only to print out the ASCII value of the key you press on the keyboard. If you are not familiar with how this works, check the section in the manual that describes the *on kbd* statement. This one statement is going to make our graphics editor really easy to implement.

As was said, the creation of a blinking cursor entails alternating colors and printing a succession of these alternating dots to the screen at the cursor location. There are lots of possibilities as to which color can be at a given screen location, and we want to move the cursor freely without worrying about destroying the image. To do this efficiently, we'll use a feature in the .GRAFIX driver called the *transfer option*. This option is really eight options in one, but we'll use *inverse replace*, which will alternate colors each time a dot is plotted. Combining that with the *on kbd* statement and adding the initialization section results in program 2.

```
10    GOSUB 1000:REM initiolize
20    HOME:PRINT"Design progrom"
35    INPUT"Grophics mode: ";mode$
36    IF mode$="" THEN 180
37    mode=CONV(mode$)
40    PERFORM grofixmode(%mode,%1)
50    INPUT"pencolor,fillcolor: ";pen,fill
60    PERFORM pencolor(%pen)
70    PERFORM fillcolor(%fill)
75    INPUT"cleor screen? ";o$
80    a$=MID$(o$,1,1):IF o$="y" OR o$="Y" THEN
      PERFORM fillport
82    horiz=xdot(mode)/192
85    scolefoc=(1/orotio)*horiz
87    PERFORM grofixon
```

```
89    ON KBD GOTO 300
90    PERFORM xfroption(%4)
91    savecolor%= EXFN%.xycolor
92    color%= EXFN%.xycolor:PERFORM
      pencolor(%color%):PERFORM
      dotrel(%0,%0):GOTO 92
```

Lines 10 through 87 call the initialization subroutine, prompt for some key values, and then turn on the graphics screen. The really interesting stuff starts on line 89 where the *on kbd* is set up to go to line 300 on any keypress. Line 90 sets up the transfer option (see your manual for more details on what this does), and line 91 saves the current color for restoring later. Line 92 is a loop that can only be interrupted by a keypress. It picks up the current color, sets the pen to that color, and then plots the dot. Because of the transfer option, the actual color plotted will be a logical alternate color to the original one. Plotting the alternate color through the same option restores the original color; thus the blinking effect. Now for the fun. We would obviously like to do more than stare at a blinking cursor. At the least, we should be able to move it around. That can be handled by a routine at line 300:

```
300   OFF KBD:PERFORM xfroption(%0)
305   PERFORM pencolor(%savecolor%):
      PERFORM dotrel(%0,%0)
310   IF KBD>31 THEN GOSUB 360:GOTO 340
320   IF KBD=27 THEN POP:GOTO 170
325   xinc%=zip%*(( KBD=21)-( KBD=8)):
      yinc%=zip%*(( KBD=11)-( KBD=10))
330   PERFORM moverel(%xinc%,%yinc%)
340   savecolor%= EXFN%.xycolor:
      PERFORM xfroption(%4)
345   ON KBD GOTO 300
350   RETURN
360   REM commands go here
```

This one little routine does quite a bit. Lines 300 and 305 just restore the state of the screen and turn off the keyboard interrupt to insure that the next statements are properly executed. Line 310 checks to see if the key pressed was a printable character; if it was, line 360 starts a series of routines to process the command that the letter represents. If the character is a control character, it is checked for *escape*. Escape is used to signal quitting the screen and going back to main level options. Thus the *pop* statement is used to jump out without leaving our *gosub* hanging. If the character is not an escape, line 325 does some clever processing: through the use of logical statements, the character is checked for one of the arrow keys and the appropriate x or y coordinate is incremented. Note that the variable *zip%* is used to multiply the effect of the increment, to enable large cursor moves. This is initialized in the subroutine at line 1000. If line 325 is confusing to you, take a moment to study its effect. Try out various values of *kbd* to see how it works. These logical statements (not available in many Basics) can replace a lot of clumsy and lengthy *if* statements. Line 330 moves the cursor appropriately, and the rest of the routine cleans up and returns to line 92. To wrap up this routine and the one above it, we need to add a couple of lines:

```
170   TEXT:GOTO 35
180   PERFORM release:PERFORM release
185   CLOSE:INVOKE
190   END
```

Just a few more statements and we'll have a fully functional program! As was mentioned earlier, line 360 begins a subroutine that handles commands. We already have a way to move the cursor around on the screen, and the following lines implement the simple functions of drawing a dot, erasing a dot, speeding up the cursor, and returning the cursor movement to normal. We'll use the letters *D*, *E*, *Z* (for zip!), and *N* to describe those functions. The statements look like this:

```
360   IF KBD>95 THEN key$=CHR$(KBD-32):
      ELSE key$=CHR$(KBD)
366   IF key$="D" THEN PERFORM pencolor(%pen):
      PERFORM dotrel(%0,%0):RETURN
367   IF key$="E" THEN PERFORM pencolor(%fill):
      PERFORM dotrel(%0,%0):RETURN
368   IF key$="Z" THEN zip%=zip%*2:RETURN
370   IF key$="N" THEN zip%=1:RETURN
399   RETURN
```

The first line, 360, just makes sure that lower-case letters are upshifted, and then, mostly for ease of reading, the value is converted to an ASCII character. From there on, *if* statements test the value and perform the functions. Notice that drawing and erasing is as simple as changing the pencolor and drawing a dot (relative plotting is used to save the trouble of getting the coordinates). The *Z* command just doubles the movement of the cursor each time it's pressed. This comes in handy, especially on the 560-by-192 screen. The return on line 399 just traps any invalid commands that might be typed and returns with no effect.

With this in mind, it's easy to add features:

```
372   IF key$="H" THEN PERFORM
      moveto(%0,%0):RETURN
```

This just homes the cursor, in case you get it lost off the screen. Yes, that's right; you can move the cursor anywhere in that −32768 to 32767 space!

Now, since we already have a circle-draw routine, we can add that simply:

```
374   IF key$="C" THEN INPUT r:GOSUB 450:
      GOSUB 900:GOSUB 460:RETURN
```

Notice that we have added references to two new *gosubs*: 450 and 460. These are used to save and restore the state of the cursor, since the circle-draw routine always leaves the cursor on the circle. They are simple and look like this:

```
450   xcen= EXFN%.xloc:ycen= EXFN%.yloc
455   cres= EXFN%.xycolor:PERFORM
      pencolor(%pen):RETURN
460   PERFORM moveto(%xcen,%ycen):PERFORM
      pencolor(%cres):RETURN
```

The other new thing about the routine at line 374 is that it asks for input. The prompt will be displayed on the text screen, so you won't see it unless you choose to add a *text* command to switch back before the *input*. You would then need to *perform* grafixon to get the screen back.

Okay, this should be enough to make for an interesting display. Now would be a good time to type this program in and debug it (it is possible to make typing mistakes!).

Okay, now that you're back for more, try adding the following:

```
376   IF key$="B" THEN INPUT w,h:GOSUB
      450:GOSUB 500:GOSUB 460:RETURN
```

That's right, *B* stands for *Box* and uses a small subroutine at 500:

```
500   w=w*scalefac
510   PERFORM linerel(%w,%0):PERFORM
      linerel(%0,%h):PERFORM linerel(%(-w),%0):
      PERFORM linerel(%0,%(-h)):RETURN
```

Are you beginning to get the idea of how easy it is to add features to this package? You probably are starting to get some ideas of your own, but here are a few simple ones while you are thinking:

```
380   IF key$="T" THEN SWAP pen,fill:PERFORM
      fillcolor(%fill):PERFORM pencolor (%pen):
      RETURN
382   IF key$="R" THEN xrem= EXFN%.xloc:yrem=
      EXFN%.yloc:RETURN
384   IF key$="L" THEN GOSUB 450:PERFORM
      lineto(%xrem,%yrem):GOSUB 460:RETURN
386   IF key$="X" THEN PERFORM fillcolor(%fill):
      PERFORM fillport:RETURN
```

Line 380 lets you toggle between *fillcolor* and *pencolor*. This is handy to erase something you just drew (like a circle or a box), by toggling the pencolor. Then you can repeat the previous command, and it will magically disappear! Line 382 simply "remembers" a point. It is used in conjunction with line 384, which draws a line from that point to wherever the cursor is located. Line 384 creates the *X* command to erase the viewport completely. That's a good idea for a command you can add to reset the current graphics viewport.

A couple more small ones, and then we'll get to the last two biggies. Here are two that permit you to save and load the graphics screen to disk. This not only is useful for making a permanent copy but can be used before any major sequence of commands. In case you don't like the results, simply reload the old contents, and the screen is magically restored.

```
388   IF key$="S" THEN PERFORM gsave."picture":
      RETURN
390   IF key$="P" THEN PERFORM gload."picture":
      RETURN
```

By now, you've probably got so many commands in the program that you have

to write them down. That's great. All huge programs were once tiny subroutines.

This next routine deserves some study. One of the things that would be nice, especially for charts and graphs, would be to put text on the graphics screen. Fortunately, the .GRAFIX permits that to be done easily. However, we can make this much more sophisticated with just a little programming effort, to wit:

```
392  IF key$="W" THEN GOSUB 450:GOSUB 600:
     GOSUB 460:RETURN
```

Subroutine 600 is used to *write* on the screen (program 3). See how all these command letters make sense? After a while, the toughest part of the program is making up new commands that don't use any of the letters already taken.

```
600  charcnt=0:line$=""
605  GET o$
610  IF ASC(a$)<32 THEN 640
615  line$=line$+a$
620  PRINT#1;o$;
625  chorcnt=charcnt+1
630  GOTO 605
640  chr=ASC(a$)
645  IF chr=13 THEN RETURN
650  IF chr<>8 OR (chr=8 AND charcnt=0) THEN
     605
655  SWAP pen,fill
660  PERFORM pencolor(%pen)
665  PERFORM maverel(%-7,%0)
670  PRINT#1;RIGHT$(line$,1);
673  PERFORM maverel(%-7,%0)
675  chorcnt=chorcnt-1
678  line$=LEFT$(line$,charcnt)
680  SWAP pen,fill:PERFORM pencalar(%pen)
685  GOTO 605
```

Program 3

The nice thing about the screen-writing subroutine is that it not only writes on the screen as you type but also allows you to use the back-arrow to erase mistakes. To permit this, the variable *line$* keeps track of what you have typed, and, if a back-arrow is encountered, the routine picks off the last character in the string, puts the pen in fill mode, and reprints the character on top of the original character, thus erasing it. Note that this backing up is done with the *perform moverel* command, and −7 is used, because that is a standard character space. Note also that you have to back up after erasing, too, since the graphics routines still think you were writing an ordinary character.

A carriage return terminates the write mode, and a check in line 650 ensures that you are not allowed to back up past the beginning point.

The last routine is the most complex, and the one in the worst shape—meaning that it works fairly well but could stand enormous improvement. It was meant as a beginning, and any help, suggestions, modifications, etcetera, would be appreciated. The routine is the promised-from-last-time *area fill* subroutine. It is integrated into the package as follows:

```
394  IF key$="F" THEN GOSUB 450:GOSUB
     1300:GOSUB 460:RETURN
```

The subroutine at line 1300 does the work, as shown in program 4.

```
1300 target=pen:startx%= EXFN%.xlac:starty%=
     EXFN%.yloc
1302 filled=0:inc=0:flog=0
1305 GOSUB 1400
1307 IF filled=1 THEN 1350
1310 PERFORM maveta(%startx%,%starty%)
1315 GOSUB 1430
1330 PERFORM linerel(%-(rxprev%-
     lxprev%),%0)
1335 startx%=(rxprev%-lxprev%)/2+
     lxprev%+.5:starty%=storty%-1
1340 PERFORM maveto(%startx%,%starty%)
1345 GOTO 1305
1350 IF inc=0 OR flag=1 THEN RETURN
1351 flag=1
1352 incval=inc
1355 FOR ivol=1 TO incval
1360   stortx%=(srch%(ival,2)-srch%(ival,1))/2+
       srch%(ivol,1)+.5
1365   storty%=srch%(ival,3):lxprev%=
       srch%(ivol,1)
1370   filled=0:PERFORM
       moveto(%startx%,%starty%):GOSUB 1305
1375   NEXT ival
1380 RETURN
1400 IF EXFN%.xycalor<>target THEN 1410
1401 If flag=0 THEN inc=inc+1:srch%(inc,1)=
     stortx%:srch%(inc,2)=rxprev%:srch
     %(inc,3)=storty%
1402 IF startx%-lxprev%<=2 THEN
     filled=1:RETURN
1404 stortx%=(stortx%-lxprev%)/2+lxprev%:
     PERFORM maverel(%-(startx%-
     lxprev%),%0)
1406 IF EXFN%.xycalar=target THEN 1402
1410 FOR i=1 TO stortx%
1415   PERFORM moverel(%-1,%0):IF
       EXFN%.xycalar=target THEN lxprev%=
       EXFN%.xlac:RETURN
1417   NEXT i
1420 lxprev%=0:RETURN
1430 FOR i=startx% TO xdat(made)
1435   PERFORM moverel(%1,%0):IF
       EXFN%.xycalor=target THEN rxprev%=
       i:RETURN
1438   NEXT i
1440 rxprev%=xdat(mode):RETURN
```

Pragrom 4

Messy, right? Right. Unfortunately there is no easy way to do general area fill. The principle is that you must first locate the cursor in the uppermost part of the figure to be filled. The routine then searches down and across for a match for its current pencolor. When that's found, the cursor returns to the starting point and searches to the right until a match is found. A line is then drawn from the right-hand point to the left. Subroutines at 1400 and 1430 do the left and right scanning.

Of course, if this was all it did, the routine would be a great deal simpler. The additional sophistication lies in an algorithm designed to enable the filling of complex shapes that contain other shapes. The simplest example is that of a circle within a circle. If you want to create something that looks like a donut, you could draw one circle inside the other and fill the space between the two circles. This routine will handle most of those cases, along with circles inside boxes and such. This is done by always favoring the right-hand side of figures and putting information in the *srch%* array when the routine suspects that it may have missed something. In addition, the routine tries to begin searches from what it suspects is the center of the open area. The problems come when the figure inside a figure is sharply off to the right hand side of the larger object. The routine will usually miss a part of the filling because of an inadequate scan. Maybe by next time there will be a hotter version, but meanwhile, it's easy just to move the cursor over and reissue the *fill* command to get what was missed.

After that apology, it should be pointed out that any simple figure, especially a convex one, will be filled reasonably well, as long as you start at the top. In fact, you might want to tweak this routine for simple figures by jettisoning the *srch%* array and changing the search to look up as well as down. Note also that no real optimization has been done. The usual tricks of multiple statements on a line and replacing constants—plus tighter coding—would probably speed up this routine considerably. Oh well, another project for you in your spare time.

One last parting shot. Often features are put in Basics that don't seem to be particularly useful. However every command has some real purpose; finding them can save lots of programming and will usually make your applications more efficient. In looking over the program above, you'll see a crying need for *instr* and *on gosub*. Notice that all of the *if key$=* statements could be replaced with the following:

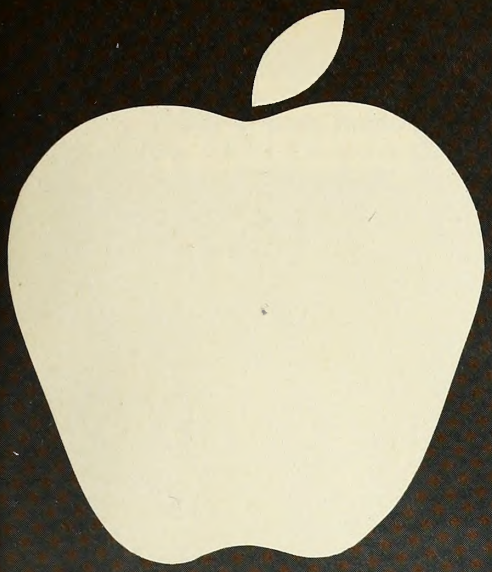
```
1057 commond$="DEZNHCBTRLXSPWF"
362  cmd=INSTR(commond$,key$)
364  ON cmd GOSUB 366,367,368,370,372,374,
     376,380,382,384,386,388,390,392,394
365  RETURN
```

Then each subroutine line could look like this example:

```
394  GOSUB 450:GOSUB 1300:GOSUB
     460:RETURN
```

This is not only neater, but much more efficient as well, since it doesn't require that you go through fifteen *if* statements just to find the one that's wanted. It also permits multiple line subroutines, which the other structure does not. Adapting the other structure to multiple line routines would require the *if* statements to be linked with *goto*, a situation to be avoided.

Well, so much for philosophy. This has been a meaty article. There are many features that need to be added to this month's package to make it truly useful, but by now, surely, you've dropped this magazine and are bent industriously over the keyboard. Time to tiptoe quietly away. . . . Genius at Work! ■



Put the missing piece back in your apple...

SMARTERM™

The Intelligent 80 Column Video Board for Apple II™ and Apple II Plus™.

Double the Apple text display to 80 columns. Smarterm is the solution—the **best** 80 column video board available.

Standard SMARTERM with 5x7 dot matrix includes all ASCII keys, 160x72 graphics mode, true shift-key operation, automatic video source selection and more.

The Enhanced Character set option provides all the above, plus 7x11 dot matrix, full descenders, an improved key layout for special characters, inverse video mode and 160x96 graphics mode. No additional hardware or firmware required.

Easiest to install. Just plug SMARTERM into your Apple and receive a crisp, easy to read display. Switching from 40 to 80 column display is automatic from the keyboard—no switches needed.

Fully compatible with Apple DOS, Pascal, and CP/M application programs. Most popular word processors are supported.

Put the missing piece back in your Apple today with the affordable, full-feature SMARTERM. See your Apple dealer today. For more information, call **800-538-8177**. In California call **408-730-0306**.

NEW

OPTIONAL ENHANCED CHARACTER SET WITH TRUE DESCENDERS.

als Advanced Logic Systems

1195 E. Arques Avenue, Sunnyvale, CA 94086

Your computer.

Compute.

Compute.

Compute.

Compute.

Compute.

Dump...

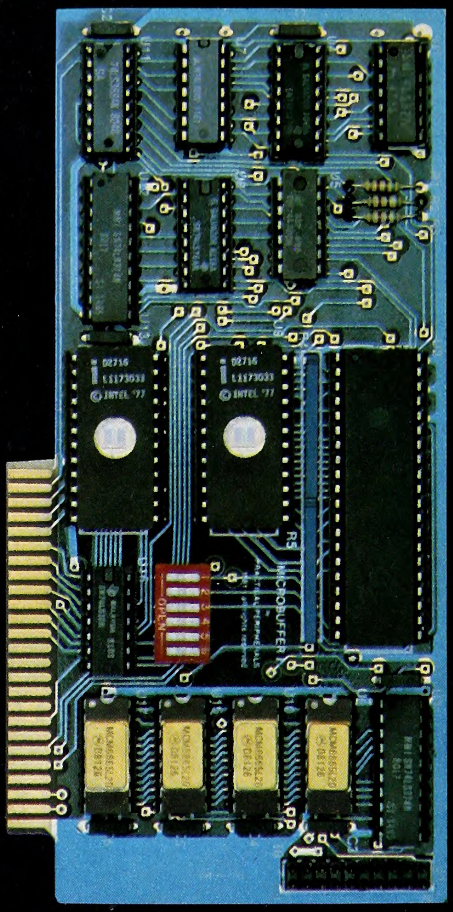
Compute.

Compute.

Compute.

Compute.

Compute.



Your printer.

...Print.

Print.

Print.

Print.

Print.

Print.

New Microbuffer II lets you use your printer without tying up your computer.

Time. As an important resource it shouldn't be wasted. One such waste is in printing, where your computer must wait for your printer. Now there's a way to eliminate this waste.

Introducing the Microbuffer II™, a buffered parallel printer interface for the Apple II® computer with 16K characters of memory (user expandable to 32K). It accepts data as fast as your computer can send it, allowing you to use your computer while the Microbuffer II is in control of your printing.

The Microbuffer II, compatible with Applesoft, CP/M® and Pascal, comes with complete print formatting features as well as advanced graphics dump routines for most popular graphics printers.

The Snapshot™ option permits you to dump the text screen or graphics picture to the printer while any program is running — without interuption.

The 16K Microbuffer II is available for $259. And the 32K version, for $299. The Snapshot option is $69.

So why waste time while your computer waits for your printer? Ask your computer dealer for the Microbuffer II or call us for the name of a dealer near you.

MICROBUFFER II

MII

PRACTICAL PERIPHERALS, Inc.
31245 La Baya Drive
Westlake Village, California 91362
(213) 991-8200

Keep Tabs on Your $$$

FIVE PROGRAMS THAT CAN HELP

BY CRAIG STINSON

A year ago, in April and May of 1981, *Softalk* published a two-part article reviewing home finance programs—single-entry accounting systems intended primarily for use in the home or in small businesses. Herewith, as an update to that account, we offer a review of five similar programs.

This year's roundup includes: *Personal Finance Manager,* from Apple Computer's Special Delivery Software, programmed by Jeffrey Gold; *Chequemate,* by Masterworks Software, programmed by Steve Collins and Tom Moch; *Financial Management System III,* published by Computerized Management Systems and programmed by Dennis Jarvis; Continental Software's *The Home Accountant,* by Robert Schoenburg, Stephen Pollack, and Larry Grodin; and *Personal Finance Master,* from Spectrum Software, programmed by Andrew Thompson.

Financial Management System III (we'll call it *FMS III*) is an update to a program reviewed last year. The earlier entry, *FMS II,* appeared first under the banner of D. R. Jarvis Computing; Jarvis has subsequently changed the name of his company.

Spectrum's *PFM* (to be distinguished, one hopes, from Apple's *PFM*), is a consolidation with enhancements of *Home Finance Pak,* which was also included in last year's reviews. *Home Finance Pak* consisted of three separate programs—one for your checkbook, one for your credit cards, and another for cash; *PFM* wraps all those functions up into a single, though not yet entirely integrated, program.

The Home Accountant could be regarded as a revision of Continental's *Home Money Minder,* but the enhancements provided in *The Home Accountant* are significant enough to point it toward a slightly different end user; hence, the new name and the continued availability and popularity of *Home Money Minder.*

The last two items on this year's survey, Masterworks's *Chequemate* and Apple's *PFM* (not to be confounded with Spectrum's), are a relatively new entry to the market and a relatively old one, respectively. Apple's program is included in this discussion because it was not available for review a year ago.

Here's what the five pieces have in common:

All will let you design a budget for yourself, your family, or your business—a list of projected expenditures per month, classified by expense categories of your own definition. Most—all but the Apple *PFM*—will also let you establish a list of expected monthly income amounts by category.

All five programs give you the means of recording your income and expenses—at least those transactions that involve your checking account—and compare monthly totals to your budget. The programs differ considerably in their power to track transactions outside of your checkbook.

All offer a variety of printed and onscreen reports to help you monitor your financial life. The feedback comes in three

general forms: summary reports, usually on hard copy; pinpoint searches, to display individual transaction records; and graphics. All but one of these programs (*FMS III*) offer some form of graphic display, although none at present offers any convenient way to dump the graphics onto hardcopy.

Additionally, each of these programs has some kind of routine to help you reconcile your checkbook with the bank's version of your account. If success at this monthly balancing act has proved elusive and your troubles have been chiefly arithmetical, then any of these programs should be a great help. They will not, of course, prevent errors of omission—failure to record transactions—but they may help you discover such errors.

So much for common ground.

Probably the most crucial difference among these five programs has to do with scope—the amount of accounting they will do. *The Home Accountant* leads the field on this score: it allows you to maintain five separate checking accounts and monitor them all, in the aggregate, against a common budget. This multiaccount capability makes the program appealing to a small businessman, work-at-home freelancer, club treasurer, or anyone else who may need to keep several checking accounts but doesn't want to get into double-entry bookkeeping.

Furthermore, *The Home Accountant* will also keep track of your credit cards—virtually as many as you wish—and cash. One cash account is made available for each checking account you keep on the system.

What makes all this nifty is that your various accounts will speak to one another. Write a check for a Visa payment and the balances in both accounts will be decremented appropriately; putting cash in the bank will lower your cash balance while hoisting your checkbook, and so on.

If cash is going into a savings account, that fact can also be noted. *The Home Accountant* lets you maintain miscellaneous accounts—classified as assets or liabilities—such as savings accounts, stock accounts, debts owed by you, or any other source of value, positive or negative, that affects your net worth. The value recorded by the system for these miscellaneous accounts can be changed either by way of a transaction in a checking, cash, or credit card account, or through an editing module; the editing module provides for changes that do not involve your more liquid accounts, as, for example, the rise and fall of your fortunes on the stock exchange.

When you set up your budget with *The Home Accountant,* you organize your financial affairs into five broad types of categories: assets, liabilities, credit cards, income, and expenses. Only the last two of these are properly considered budget categories; the other three are really accounts. But all five types have a certain logical equivalence to the system—they are places to file your transactions.

When you subsequently record a transaction, you designate, among other things, the appropriate category; if that category is something that falls under the broad type of income or expenses, then your transaction will find its way into a budget report. If the category is something like MasterCard that belongs to one of the other three types, then the transaction will be recorded as a transfer of funds from one account to another.

This design feature—this apparent equivalence of accounts and budget categories—is unique to *The Home Accountant.* With the other programs you can track multiple accounts if you wish, but transfers from one account to another will have to be recorded in two separate steps.

IF YOU WANT THE BEST IN PERSONAL FINANCE SOFTWARE SPECTRUM HAS THE ANSWER!

PERSONAL FINANCE MASTER™

- Tracks your checks, deposits, charges, loan payments, credit cards, IRA accounts, broker accounts, even cash and fixed assets
- Integrates them all into a personal net worth statement
- Budgets your expenses and cash flow
- Reconciles your bank statements
- Prints your checks on standard computer forms
- Addresses your checks or other mail using standard computer address label forms
- Produces Hi-Res plots of income & expense
- Sorts your expenses into tax deductible categories
- Automatically records recurring transactions
- Handles split transactions
- Provides an easy to use year-end rollover
- Provides a generalized search capability for custom reports
- Uses "on the screen" forms for convenient data entry, review and editing
- Handles up to 7 different asset or liability accounts
- Provides up to 700 transactions per month capacity
- Provides up to 100 user-defined budget categories
- Provides a user defined menu of standard names & purpose codes for speeding data entry and producing compact data storage (up to 3000 transactions per diskette)
- Produces 8 CRT or printed reports/graphs
  - Account register
  - Net worth statement
  - Search (custom) report
  - Income/expense (Budget report)
  - Standard names & purpose codes
  - Standard addresses
  - Chart of accounts
  - Income/expense vs time graph
- Provides extensive error trapping, user prompts, and default values for easy data entry

Personal Finance Master is available at your local dealer or you may order directly from Spectrum Software. PFM is supplied on a standard DOS 3.3 diskette and includes an extensive 50 page bound instruction manual. PFM requires an Apple II with Applesoft in Rom or an Apple II Plus, 48K RAM, one disk drive (DOS 3.3) and an 80 column or larger printer (optional). If you would like additional information write for our free PFM Factsheet.

Ordering Information: Price $74.95
To order send your name & address and your check, money order, COD request, or credit card information to the address below, or you may order directly by phone. Add $2.00 postage & handling with each order ($5.00 foreign). California residents add 6½% sales tax.

SPECTRUM SOFTWARE
142 Carlow, P.O. Box 2084
Sunnyvale, CA 94087

VISA master charge

For Phone Orders: (408) 738-4387
Dealer inquiries invited.

Jarvis's documentation for *FMS III,* for example, recommends that if you wish to monitor cash expenditures as well as those from your checkbook, you specify a budget category called transfer—or some such. A check for cash can then be recorded as an expenditure to transfer and a subsequent deposit from transfer to the cash account. A slush fund category like that should, theoretically, show up as a zero on the budget summary report.

Of the four programs other than *Home Accountant, FMS III* is the largest in accounting power—potentially larger than *Home Accountant,* though less automatic in its functioning. There is very little, in fact, that's automatic in *FMS III,* and that may constitute its main appeal to certain users.

In *FMS III,* you do the file management yourself, much as you do with programs like word processors. It might be customary to organize your transactions into files by month and account name—say January checkbook or April Diners Club—and, in fact, the auditor module, a separate program that performs the budget analysis and reporting, requires that the names of files it scrutinizes begin with the first three letters of a month name; but, really, you can do anything you want. You can create files that represent six months of frugality or a weekend of high living, or you can keep alternate versions of your books, on the same disk or not—whatever. This amount of flexibility is bound to have advantages for certain kinds of applications, but for the novice it makes for more work and more opportunity for confusion and error. To Jarvis's credit, the program is loaded with fail-safe features that discourage the unwary from overwriting or accidentally deleting files.

Furthermore, while file management in *FMS III* is a roll-your-own affair, the data-entry module—like that of the other four programs under review here—is amply supplied with simplifying, keystroke-reducing features. We'll come to those presently.

The Spectrum *PFM,* like *The Home Accountant,* invites you to record accounts for checking, cash, credit cards, and longer-term assets and liabilities. Like its competitor from Continental, *PFM* uses all this information to generate a comprehensive statement of your net worth. However, like the separate checking, cash, and credit modules that made up Spectrum's older *Home Finance Pak,* these various accounts of yours remain aloof from one another and will not communicate except through the mediation of a transfer fund.

Also, unlike *The Home Accountant,* in which accounts share file quarters with budget categories and hence are nearly unlimited in number (actually that program allows a maximum of a hundred accounts and/or budget categories), the organization of *PFM* is such that only seven accounts can be accommodated. If you want to take advantage of the program's ability to print a comprehensive personal balance sheet, you need two accounts for miscellaneous assets and liabilities, and at least one for a checkbook, leaving you only four for anything else that's liquid—like cash and credit cards.

With *Chequemate,* you can establish as many accounts as you wish, but they need to be on separate disks, and tracking them all against a common budget looks like it would be a difficult proposition. When you go through *Chequemate*'s initialization procedure, you're given the option of putting account and budget information on the program disk or on a data disk. Separation of data from program not only gives you expanded data space; it also could be useful for a family that wants to track separate accounts against separate budgets.

Chequemate, like Apple's *PFM,* is somewhat less extensive in scope than the other three programs. It doesn't, for example, have adequate provision for recording cash receipts or expenditures.

It does, however, include a feature that's unique. It lets you record your bills—your accounts payable. When the gas bill arrives in the mail, you can call up *Chequemate*'s charge module and post the amount due, along with the appropriate budget category. Later on, when you get around to paying the bill, you'll go into the area of the program that records transactions, enter your check number, the payee, and the budget category; at that point, all outstanding bills for the specified category will appear in a list near the bottom of the screen, each with its own program-generated number. To pay a bill off, you just type P plus the appropriate number, and the blot is removed from your record.

This bill-recording feature gives you some primitive ability to track credit card accounts, although you may find its use in that connection a little awkward. For example, if you happen to owe $800 to the Bank of Grinnell, and you want to give them only $35, you'll be disappointed that *Chequemate*'s bill feature won't make the necessary adjustment. To use this feature, you must pay bills in full. Of course, you can record your credit purchases one by one as you make them. In the transaction module they'll appear as separate bills, and you can pay them all off at once.

In any case, *Chequemate*'s charge module feature is unique, and could be a real plus if you're one who likes to misplace, or just forget about, your bills.

Apple's *PFM* is designed for one checking account, one cash account, and up to a dozen credit cards. Like the Spectrum *PFM* and like *FMS III,* it's capable of passing information from one account to another by way of intermediary budget categories. In the case of Apple's program, rather than make a deposit to, say, your cash account, you would record a negative expenditure. The program permits deposits only to your checking account, but will accommodate negative expenditures.

The main virtue of Apple's *PFM* is its simplicity; designed for a relatively small-scale application, it is the easiest of the five programs to understand and the quickest in start-up time. The price of simplicity in this case is limitation of scope; the program permits a maximum of two hundred transactions per month sorted into twenty-four budget categories. The number of transactions is probably adequate for most users, but the budget category limitation may prove constraining.

By contrast, *The Home Accountant, FMS III,* and Spectrum's *PFM* all permit one hundred categories, and *Chequemate* has no specified limit. The number of budget categories actually available on *The Home Accountant* is somewhat under one hundred, depending on how many asset, liability, and credit card accounts you've established.

There are some important differences amongst these five programs with respect to the manner in which data is entered and displayed, and with respect to what kind of data may be entered. Not all, for example, provide you with the means to flag transactions for recall at tax time, and not all give you room to write a memorandum explaining the purpose of a transaction.

With the exception of *FMS III,* the programs fall into two categories—those that organize your data in columns across the screen and those that organize it in rows. The tradeoff is this: if the program puts your data—check number, payee, amount, and so on—in columns, then it can display a good many transactions on the screen at once, and the screen will look somewhat like a page of your manual check register. If it puts each data field on a separate row, you'll be able to see only one transaction at a time. On the other hand, the row-displayer offers the luxury of putting a lot more information into each field.

Both *PFM*s, Spectrum's and Apple's, fall into the column-displaying category, while *The Home Accountant* and *Chequemate* are row-oriented programs. Meanwhile, Jarvis has found a way to make *FMS III* straddle the fence by taking a basically column-oriented approach but wrapping each transaction entry into two rows on the screen.

Chequemate accepts input in seven fields: item number, date, payee (or source, as the case may be), amount, budget category, comment, and reconciliation status. There is no provision for a tax flag. The item number begins with an alpha code that selects one of four transaction types: check, deposit, automatic teller withdrawal, and bank charge. The item num-

ber in each category increments automatically as a default option. Defaults are also provided for date and reconciliation status—presumed to be outstanding.

Chequemate also allows you to designate four-letter macros representing frequently occurring transactions. In a separate section of the program, you can establish a table of these standard entries; for each standard entry you can put data in as many fields, with the exception of date and item number, as you wish. So the standard entry for rent or house payment might include an amount, whereas the entry for the local grocery store will probably not. In the latter case, you would plug in the appropriate amount when you call up the standard entry to record a transaction. Hitting a question mark in the payee field displays a list of your standard entry macros.

Hitting an asterisk in the comment field allows you to designate a second budget category for your check. So, for example, if you write a check for groceries plus extra cash, you can do the appropriate accounting as a single transaction. The check may be split into no more than two categories, however.

Above the transaction data, the program displays an action line, showing the commands available to you at any given moment. Below the transaction data appear your outstanding bills for the currently selected specified budget category. You can dump the contents of any screen to the printer with a control-Y.

Apple *PFM*'s transaction entry module lines up data in six fields across a single row per transaction. The fields are: number, date, item, amount, tax, and budget category. There's no space for a memorandum. Items you designate as reconciled via the program's separate reconciliation module appear in subsequent displays with asterisks next to their transaction numbers.

Like *Chequemate,* Apple *PFM* displays your current command options at all times. It also displays your budget category options. In the budget portion of the program you are asked to supply two-letter codes for each category. These two-letter codes appear at the bottom of your screen when it comes time to record a transaction. Your credit card options are displayed similarly.

Unlike *Chequemate,* the Apple *PFM* does not have any overtly indicated procedure for recording split transactions—those that require multiple accounting. The program, however, makes no objection to your entering several checks with the same check number. So a tiny subterfuge on your part—the pretense of having written several checks with the same number—will get the proper accounting done.

Nor is there any provision for transactions, like an automatic teller withdrawal, that involve your checking account but not your checks; but you can enter a check number of zero to cover such items.

Checks, deposits, cash expenditures, and credit card purchases are all recorded from the same program module. You simply enter a C under transaction number for cash, a D for deposit, or the appropriate two-letter code for a charge card.

A control-P will dump any screen to the printer.

The *Home Accountant* stores your data in seven fields, including both a tax flag and ample space for a memorandum. On the checking account screen there's an eighth field to indicate reconciliation status; this defaults to outstanding. The transaction number increments automatically as a default. On the cash and credit card screens, where a check number type of entry wouldn't be appropriate, the system provides a slot for a six-character alphanumeric code, in case you should need one. Any transaction-entry screen can be printed with a single keystroke.

Should you have forgotten the name of a budget category, the program offers a soundexlike routine to assist you. Just enter the beginning letter of your category and the system will bring up the first category in its budget file beginning with that letter. If that's not the right one, hit the spacebar and you'll get the next one starting with that letter, and so on. When the right category comes up, you just hit return to accept. If you happen to have a hardcopy list of the budget category numbers—generated by the system at the time you establish your budget—you can enter the appropriate number on the category line and the program will substitute the category's name, again letting you hit return to confirm.

The split transaction procedure for multiple disbursements works for any number of splits and works for cash and credit cards as well as for checks and deposits. Unfortunately, the program, like the Apple *PFM,* requires you to enter irregular transactions with the bank, like less cash from a deposit, through the artificial means of a dummy check.

For recurrent monthly bank transactions that don't involve a check—something like a transfer of funds from checking into savings—the program lets you set up as many as five automatic transactions. As soon as you initialize a new month on the system, these transactions are posted to the appropriate accounts.

The Spectrum *PFM* uses a seven-column horizontal format to record your transaction data. Of all the programs in this survey, Spectrum's is the one whose style most closely imitates that of your own checkbook. Up to eighteen transactions can be viewed onscreen at one time. At the bottom of the screen, the

program flashes the available command options; some users may find the flashing display tiresome.

Spectrum's seven columns include a tax flag and a reconciliation flag, but, alas, there's no room for a memo. There's also no room for more than four characters in the payee and budget category fields. The system, however, records internally the full entries in these fields and prints them in full on hardcopy.

Entries to the payee and category fields are made by way of prearranged numeric codes. Hitting M in the budget field allows a multiple disbursement to as many as ten budget categories.

Unfortunately, there is no longer a feature to display the prearranged category and payee codes from the transaction-entry screen. You have to remember them or keep a printout handy. The old *Check Register* program from Spectrum's *Home Finance Pak* was more convenient in this respect.

Not all payee or category entries have to be predefined; there's a separate number available that codes for items of both kinds that don't come up often enough to merit their own number. The system prompts for details in such cases and handles the items just as it does coded items.

There's also a way to set up standard monthly transactions, like house or car payments. As with *Chequemate*'s standard transaction feature, these can be supplied with full data or not, as you choose. If you wish, you can leave certain fields blank and enter them manually through an edit procedure.

An interesting novelty: this *PFM* does not have a separate reconciliation module. Rather it lets you scroll through your check register by way of the edit procedure and mark off items in at the bank. Then a single keystroke will summon a reconciled balance—the number that's supposed to agree with the bank's statement. If you don't require a formal reconciliation statement, this may be simpler than going through a separate reconcile program.

FMS III presents a compromise between the column and row-oriented display techniques, revealing at any time the latest four transactions in your file, organized in six columns wrapped over two lines.

The six available fields are number, date, item, explanation, amount, and budget category. No tax flag, but the provision for memoranda is generous.

Data entry is simplified by a system of three-letter macros, established during your initialization phase on the program. A question mark in the appropriate data-entry field will summon a display of your macro codes and their meanings.

In *FMS III*, as in the Spectrum *PFM*, you establish macros for both payee and budget category. Unlike in Spectrum's *PFM*, however, your codes in *FMS III* are linked. This means that if your check to Store Q nearly always buys a can of paint, you can enter the appropriate macro for Store Q and just hit return in the category field. On the other hand, if you have occasion to purchase toothpaste from Q or get your paint from R, you can override the default linkage and use separate macros in each field.

FMS III recognizes five kinds of transactions, symbolized by #, D, M, C, and *. The first is for checks (or cash or credit card expenditures, depending on what kind of account you're monitoring). The D is for deposit and the C for charge—miscellaneous transactions with the bank that don't involve your writing a check.

The M is for memo, which is something of a novelty in these home finance programs. Jarvis's documentation indicates that the memo option is provided strictly for informational purposes—for reminders of financial obligations, perhaps, or some such. The transaction so entered has no effect on your account balance and is ignored by that part of the system that audits your account for summary reports or budget analysis.

The last transaction type, indicated by the asterisk, is for multiple disbursement, and here *FMS III* really seems a little clumsy. If you want to split a check between groceries and cash, you have to set up special budget categories to receive the split transaction. In other words, if you're in the habit of getting some of your pin money straight from the bank and some from the local market, you need at least two budget categories for cash. This might be all right if the audit routine then lumped the two categories back together in its report—but it does not.

On the matter of reports, both printed and screened, here's how the five programs compare.

Apple's *PFM* has no print routine as such, only an option to dump the current screen. That may well be its most serious limitation, since a whole pile of screen dumps—complete with all those prompts that are so handy in their proper context—does not make for ideal documentation.

On the other hand, the program is loaded with search features. There are five types of search criteria that can be designated, either singly or in combination: entry type (all entries, a range of check numbers, all deposits, all cash, or a specific credit card), entry date (current month or a specified range of months), specific budgets, payee, and additional criteria. The last of these brings up other options—reconciled only, unreconciled only, or tax deductible only. The search routine will also sort the items it finds, if you desire.

So if you don't require comprehensive printouts, this search power combined with the screen-dump capability may prove adequate after all.

The Apple *PFM* also will give you lo-res bar graphs of actual versus projected expenditures in specific budget categories for a whole year or in all categories for a given month.

With respect to searching and printing, *Chequemate* stands in pretty direct opposition to the Apple *PFM*. It has no search routine as such, but it's amply provided with the capacity to generate printed reports. In place of a search routine, *Chequemate* offers a review option that will let you scroll through your transactions, or a range thereof designated by transaction number, one by one. Because this sequential review process can become time-consuming as your files grow, and since not everybody needs their entire financial life socked away on disk for posterity, the program provides the option of deleting reconciled items.

Chequemate's hardcopy report options include a check list—between designated dates—sorted, if you wish, by transaction number, category, or payee. The check list includes a summation of checks, deposits, automatic teller transactions, and bank charges, each category subdivided into cleared and outstanding.

Other reports include a budget list, an actual versus projected budget comparison—for a specified time interval—a list of your standard entry macros, and a list of outstanding bills.

Chequemate also generates hi-res bar graphs for onscreen budget analysis.

FMS III generates the following printouts: a complete transaction list for a specified group of files, a list of reconciled items, a list of unreconciled items, a summary of a trial reconciliation, the results of an onscreen search request, an audit report, and a budget report. The audit report is a summation of expenditures or receipts in specified categories, while the budget report performs the same function and compares results to projected amounts.

FMS III's search routines act on only one search criterion at a time, rather than performing more selective searches the way some of the other programs do. The search criterion can be any of the program's six data fields and can cover as many as sixteen different files. There's an expanded search mode, in which the user needs to designate only the first few letters of a search criterion. The user can display search results either in a compacted one-line format or in the familiar two-line wrap-around style of the data-entry module.

The Home Accountant performs searches based on multiple criteria—virtually any grouping of data fields you choose—and it prints a hardcopy activity report for a specified time period, either in toto or restricted to transactions meeting specified criteria. The activity report can cover checking, cash, or

credit accounts, either singly or in combination.

Other printed report options include an income and expense summary for a month or range of months, various kinds of budget reports, and a personal balance sheet; this last comes in three flavors: for the current month only, compared to budget, and compared to the most recent month. A hi-res bar graph budget analysis is also available.

The Home Accountant also prints checks on standard forms and posts the transactions appropriately. And last, but not least, if you have an MX-80 printer, *Home Accountant* automatically sends the control characters necessary to generate 132-column reports in condensed format.

Spectrum's *PFM,* like Apple's *PFM* and like *The Home Accountant,* is capable of searching on multiple criteria. Uniquely, it allows you to specify a range in any field, so that you could, for example, ferret out all transactions between the amounts of one hundred and two hundred dollars filed in budget categories 9 through 21 inclusive.

Like *The Home Accountant,* Spectrum's program also prints checks and produces a personal balance sheet. Its printed budget reports offer you a choice between seeing your variance from budget indicated in absolute amounts or in percentages.

Its graphic budget analysis goes a little farther than *The Home Accountant*'s; in addition to a line graph showing amount spent in a given category for a range of months, it also offers you a linear regression plot.

So where does all this information leave you? Here are a few final words, complete with subjective impressions, about each of these five programs:

Apple's *Personal Finance Manager* is a pleasure to use. While somewhat limited in scope—it's really an expanded checkbook keeper as opposed to a full-fledged single-entry accounting system—it's an exceedingly well-made program. You can hardly get lost, confused, or annoyed in it. Its $50 price is reasonable, but if printed reports are vital to you, you'll want to look elsewhere.

At $39.95, Masterworks Software's *Chequemate* is the bargain of the bunch. If you don't require printed checks or personal balance sheets, and you don't need full accounting of cash and credit transactions, this may be ideal for you. *Chequemate* is the only one of these programs that provides for the posting of bills as well as payments.

Computerized Management Systems's *Financial Management System III* should appeal to the person who wants maximum freedom to manipulate and organize financial data. At $120 it may put a strain on the very budget it's meant to manage.

Spectrum Software's *Personal Finance Master* is an ambitious program that suffers only by comparison to *The Home Accountant.* Within its scope it does a great deal of accounting, but it does it less easily, less gracefully, and less quickly than *The Home Accountant.* What's most unfortunate is that the person who really needs the kinds of accounting services that *PFM* can provide will probably need to track more than the seven accounts available in *PFM. PFM*'s price is a reasonable $74.95.

Of the five programs reviewed here, *The Home Accountant* is the most thorough and powerful. Considering how much it can do, it's remarkably easy to use. Its price is also $74.95. ∎

Apple Computer Special Delivery Software, 10260 Bandley Drive, Cupertino, CA 95014; (800) 538-8400; in California, (800) 672-1424. Computerized Management Systems, 1038 Cadiz Drive, Simi, CA 93065; (805) 526-0151. Continental Software, 16724 Hawthorne Boulevard, Lawndale, CA 90260; (213) 371-5612. Masterworks Software, 1823 W. Lomita Boulevard, Lomita, CA 90717; (213) 539-7486. Spectrum Software, Box 2084, Sunnyvale, CA 94087; (408) 738-4387.

Once you accept the challenge, science fiction becomes reality.

SENTiENT SOFTWARE

Until now, you've never heard of a Cyborg. Now you've become one. After the operation, you weren't yourself. A voice kept resounding in your skull, offering opinions without being asked. Since there was nothing left of your life, you decided to accept the mission NASA II offered you. You decided to make the best of a bad situation . . . so you thought.

Michael Berlyn, the author of our first science-fiction adventure, Oo-topos, now gives you a chance to test your skill in the realm of the unknown. Cyborg takes the adventure game beyond any other game of its kind. Complex and involving. Cyborg is an exciting way to interact with other creatures. With an original plot, including hidden real time graphic simulation, Cyborg is a totally logical program for adventure.

See your local software dealer and discover Cyborg. Requires DOS 3.3 48K Applesoft™* in ROM.

Exclusively distributed by SOFTSEL, 8295 S. La Cienaga Blvd., Inglewood, CA 90301, (800) 421-5770, in California (800) 262-4242

*Applesoft is a registered trademark of Apple Computers, Inc.

Softalk Presents The Bestsellers

The midwinter doldrums settled with vengeance into the Apple marketplace in the month of February. Typical comments on business from the retailers ranged from "not bad for a February" to "cruddy," this last from a suburban Chicago dealer.

In general, the downturn was no greater than normal after the holiday buying splurge of November through January. Interestingly enough, sales of the Apple III and Apple III software were not affected by the slump. The Apple III continues to perform better than it ever has, although the value of that comparison is mitigated by the machine's previous poor showing.

Starting next month, *Softalk* will include in its Bestseller lists a special category for Apple III software. That list may have added interest just from the fact that *VisiCalc* won't be the perennial leader. The *Softalk* list measures aftermarket sales and *VisiCalc* is included in the sale of the basic machine, thus foreclosing it from consideration.

Preliminary sampling indicates that *Personal Filing System III* from Software Publishing Corporation presently leads the pack, with *Apple Writer III, Mailing List Manager,* and *Word Juggler* positioned next. The former two programs are from Apple and the latter is from Quark Engineering.

In the Apple II marketplace, *VisiCalc* retained its customary position at the top of the heap; but it was pressed for that honor for the first time in months. *Wizardry*, a program that's becoming a success story all its own (as documented in the March/April issue of *Softline*), vaulted into second place just before the release of a new scenario.

Wizardry's showing was even more remarkable in that only three entertainment programs made the top ten places, and *Wizardry* is more expensive than the others.

Snack Attack dropped from fourth to fifth, but remains strong; and *David's Midnight Magic* clung to tenth, a drop of three places.

Dropping from second to third is that other surprise program, On-Line's *Superscribe II*, now to be known as *Screen Writer*. The program was not shipped for most of the month because of legal entanglements about the name, but there were already so many in the pipeline that it clung to a high position on the charts.

Personal Filing System dipped from third to fourth but remained the leader in the red-hot database competition. There were some changes there, however, as *DB Master* vaulted past *VisiFile* to be the second-ranked database manager. New competition from an old source also surfaced, as the revised *Data Factory* reappeared on the Top Thirty list for the first time in several months. Micro Lab's entry has a long way to go, but it's clearly back in the running.

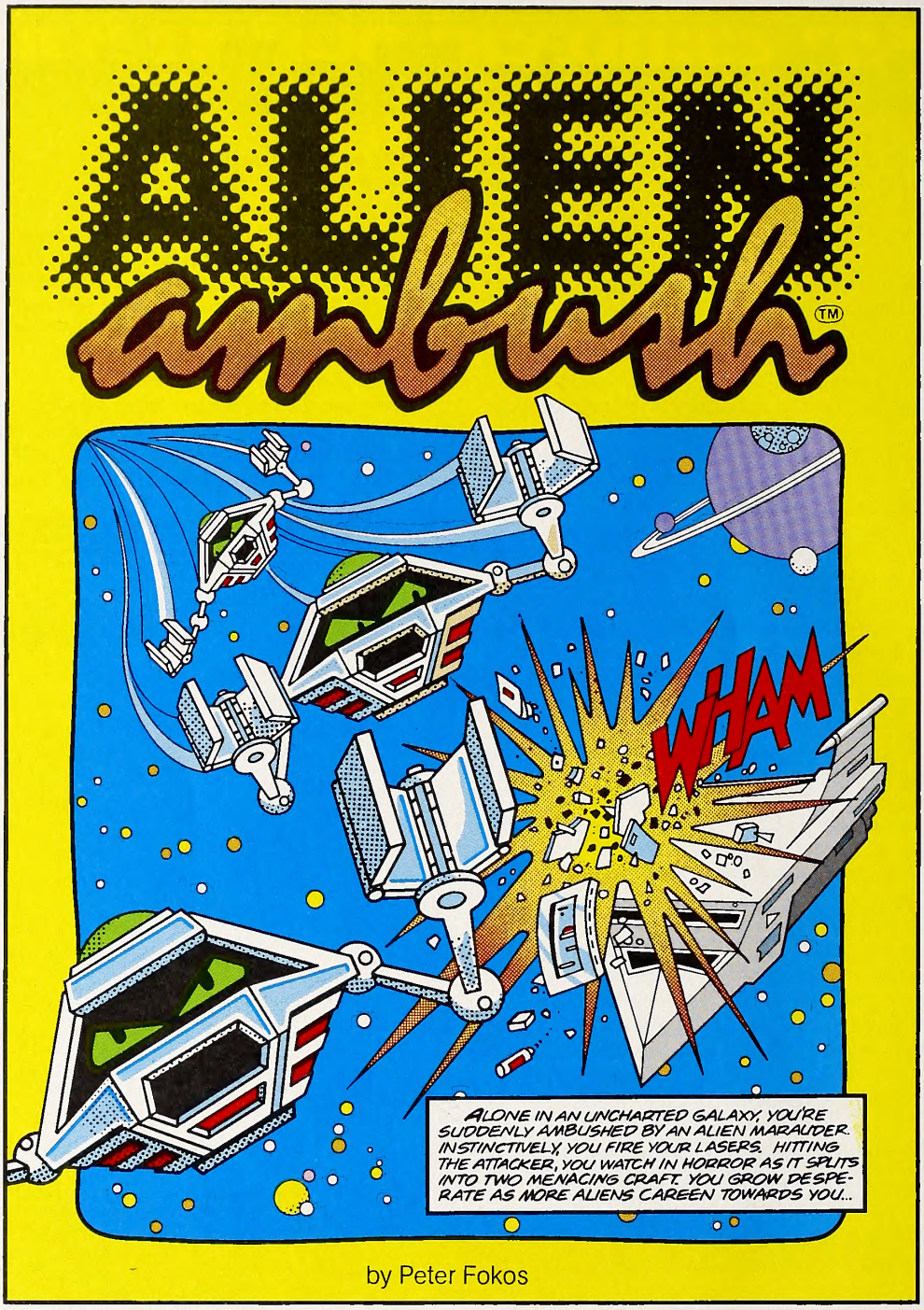
Home Accountant rose from eighth to sixth and continued its domination of the personal finance market. Apple's *Personal Finance Manager*, Continental's *Home Money Minder*, and Computerized Management Systems's *Financial Man-*

Business 10

This Month	Last Month	
1.	1.	**VisiCalc,** Software Arts/Dan Bricklin and Robert Frankston, VisiCorp
2.	2.	**Personal Filing System,** John Page, Software Publishing Company
3.	4.	**DB Master,** Alpine Software/Stanley Crane and Jerry Macon; and Barney Stone, Stoneware
4.	6.	**BPI General Ledger,** John Moss and Ken Debower, Apple Computer
5.	5.	**VisiTrend/VisiPlot,** Micro Finance Systems/Mitch Kapor, VisiCorp
6.	3.	**VisiFile,** Creative Applications/Colin Jameson and Ben Herman, VisiCorp
7.	—	**Data Factory,** Bill Passauer, Micro Lab
8.	7.	**PFS: Report,** John Page, Software Publishing Company
9.	10.	**Accounting Plus II General Ledger,** Software Dimensions, Systems Plus
10.	—	**VisiPlot,** Micro Finance Systems/Mitch Kapor, VisiCorp

Word Processors 5

This Month	Last Month	
1.	1.	**Superscribe II,** David Kidwell, On-Line Systems
2.	—	**Magic Window,** Gary Shannon and Bill Depew, Artsci
3.	2.	**Apple Writer,** Apple Computer
	3.	**WordStar,** MicroPro
5.	—	**Supertext II,** Ed Zaron, Muse



You haven't lived until you've died in space.

And here's your chance.

Software author Peter Fokos has created Alien Ambush, a space age nightmare. This hi-res, full-color arcade game is written completely in assembly language to give those nasty aliens every advantage.

So if you have access to a 48K Apple* with DOS 3.3, and you're hot for some new thrills, Alien Ambush was written for you. But be warned: It just got a lot tougher to survive in space.

Available at finer computer stores everywhere. *Apple is a trademark of Apple Computer, Inc.

Softalk Presents The Bestsellers

agement System III all continue to show strength.

Although the arcade genre is currently taking a back seat to business software, it's providing some of the more interesting developments. Broderbund released *Star Blazer*, the newest from *Alien Rain* author Tony Suzuki; and although it had limited distribution for a limited time in February, it still was the highest ranking new game at fourteenth. *Alien Rain*, under its original name of *Apple Galaxian*, was the first program ever to dethrone *VisiCalc* from first place.

Broderbund had another new entry in the Top Thirty in *Track Attack*, giving it a total of four.

Twerps, the newest from Sirius, leaped into fifteenth and gave the company three in the Top Thirty, *Gorgon* and *Sneakers* being holdovers.

Another new program to the Top Thirty represented an old author in a new venue. Nasir, whose Sirius offerings mostly resided in the top ten, took two tries in his own company before hitting the charts, but *Horizon V* did it, tying for eigh-

Strategy 5

This Month	Last Month	
1.	1.	**Castle Wolfenstein**, Silas Warner, Muse
2.	—	**Sargon II**, Dan and Kathe Spracklen, Hayden
3.	—	**Robot War**, Silas Warner, Muse
4.	2.	**Flight Simulator**, Bruce Artwick, SubLogic
5.	—	**Southern Command**, Roger Ketieg, Strategic Simulations

Adventure 5

This Month	Last Month	
1.	1.	**Hi-Res Adventure #1: Ulysses and the Golden Fleece**, Bob Davis and Ken Williams, On-Line Systems
2.	2.	**Zork II**, Infocom
3.	—	**Mummy's Curse**, Highlands Computing
4.	4.	**Hi-Res Adventure #3: Cranston Manor**, Harold DeWitz and Ken Williams, On-Line Systems
5.	—	**Time Zone**, Roberta and Ken Williams, On-Line Systems

Fantasy 5

This Month	Last Month	
1.	1.	**Wizardry**, Andrew Greenberg and Robert Woodhead, Sir-tech
2.	—	**Apventure to Atlantis**, Bob Clardy, Synergistic Software
3.	2.	**Ultima**, Lord British, California Pacific
4.	5.	**Crush, Crumble, and Chomp**, Automated Simulations
5.	3.	**Empire I: World Builders**, David Mullich, Edu-Ware Services

penguin software

The Leader in Apple Graphics

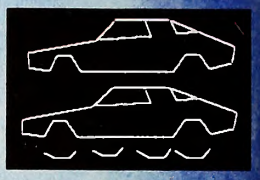
by Chris Jochumson and Mark Pelczarski

Now anyone can put professional graphics into their own programs. *The Graphics Magician* contains machine language animation routines that use the same techniques as most of the popular Apple arcade games. Three animation editors let you design your figures, their paths, and assemble animation with up to 32 independent objects. Also included is a hi-res picture/object builder that lets you store hundreds of 100-color pictures on a single disk and recall them quickly from your own programs. These exact routines are being used in the new graphic adventure games from Scott Adams' *Adventure International*. Plus, a new shape editor greatly extends the capabilities of Apple shape tables with multicolors and angles that are preserved on scaling. All design of graphics is done through menu-driven editors; to use in your programs, just attach our machine language routines. Extensive documentation makes this package easy to use for the beginning programmer, yet flexible enough for the most advanced.

THE COMPLETE GRAPHICS SYSTEM II

by Mark Pelczarski

This improved version of the already popular *Complete Graphics System* is a complete and easy to use 2-dimensional and 3-dimensional graphics design package for the non-programmer. Create color screen images with lines, circles, ellipses, automatic 108-color filling, and "paintbrushes". Use high-resolution text anywhere on the screen in dozens of colors. Create and edit shape tables, and shrink pictures for multiple displays. Amazing 3-D routines let you draw the parts of 3-D objects on the screen, then assemble, rotate, and edit them visually in true 3-D perspective. All the software is easy to use, even for the novice. Programmers can use the graphics, and even our machine language routines, in their own software. This one package gives you more than most other graphics software packages combined. Find out why *The Complete Graphics System* is top rated in Apple graphics.

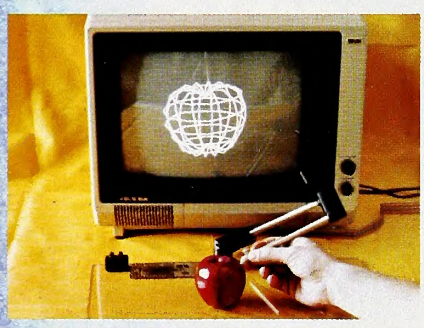


Development of a 3-D image using The Complete Graphics System.

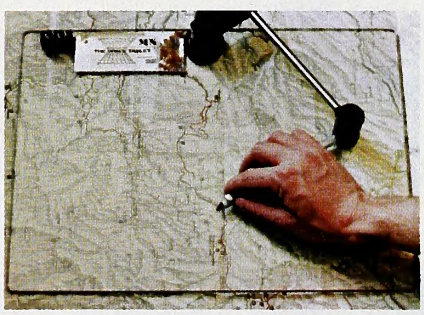
Add a New Dimension to your Apple! with THE SPACE TABLET

MCS micro control systems, inc.

a 3-D Graphics Tablet



This incredible new device produced by Micro Control Systems, Inc. functions as a standard graphics tablet in two dimensions, until you decide to expand to real-world 3-dimensional input. Now you can trace 3-D objects, either real or imaginary, then view and modify them in 3-D perspective with your Apple computer. The 16"x13" tablet comes with a customized version of *The Complete Graphics System*, specifically designed to accept input from three dimensions. In addition, the *Space Tablet* includes machine language software that allows you to design your own programs using 3-D input. This unique hardware/software package also comes at a very palatable price -- half the cost of most standard graphics tablets. See your Penguin Software dealer today!



Special Effects

by Mark Pelczarski

Like nothing else on the market, this unique software package allows you to escape the "coloring book" approach to computer graphics, giving you a palette of 108 colors and 96 different brushes for creating or enhancing color computer images. Also included is a magnifying mode that lets you magnify images 2 or 4 times and edit them point-by-point, a "picture packer" that lets you store images in a fraction of the space normally taken, and a set of tricks that allow you to reverse colors, perform mirror images, and move parts of screen images around and to other pictures. *Special Effects* is great as a stand-alone package -- or the perfect complement to *The Complete Graphics System*.

The Complete Graphics System II
paddle/joystick version $69.95
Apple Tablet or Hi-Pad version $119.95

Additional Fonts and Character Sets
for The Complete Graphics System
(50 fonts on two disks) $19.95

The Graphics Magician
paddle/joystick version $59.95

Special Effects
paddle/joystick version $39.95
Apple Tablet or Hi-Pad version $69.95

The Space Tablet, with custom version of
The Complete Graphics System $395

All Products Now on Non-Protected Disks for your Convenience

All products require an Apple II with Applesoft, 48K, and a disk drive.

See your dealer, or contact us.

penguin software
830 4th Avenue
Geneva, IL 60134
(312) 232-1984

VISA and MASTERCARD accepted.

Apple II is a trademark of Apple Computer, Inc.

Softalk Presents The Bestsellers

teenth with the fading *Apple Panic*.

The Business 10 underwent no radical changes with the exception of the reappearance of *Data Factory*, as mentioned before.

That was not the case with Word Processors 5, where *Magic Window* and *Supertext II* regained the list. *Magic Window*, after a one-month hiatus, jumped into second.

The story on *Supertext* is apparently that many users who were waiting for the new 40/80 column version to appear finally decided to make the plunge with the old version.

The only major change in the Home 10 was the appearance of *Apple Speller*, Sensible Software's dictionary program for Apple word processors. It came from nowhere to capture fifth and nearly made the Top Thirty list. Howard Software's *Tax Preparer* continues to dominate the tax packages and *ASCII*

Home 10

This Month	Last Month	
1.	1.	**Home Accountant,** Bob Schoenburg, Larry Grodin, and Steve Pollack, Continental Software
2.	3.	**Tax Preparer,** James Howard, Howard Software
3.	2.	**Personal Finance Manager,** Jeffrey Gold, Special Delivery Software, Apple Computer
4.	4.	**Typing Tutor,** Image Producers, Microsoft
5.	—	**Apple Speller,** Sensible Software
6.	7.	**Mastertype,** Bruce Zweig, Lightning Software
7.	5.	**Home Money Minder,** Bob Schoenburg and Steve Pollack, Continental Software
8.	6.	**Tax Manager,** TASO, Micro Lab
—	—	**"The World's Greatest Blackjack Program,"** Special Delivery Software, Apple Computer
10.	10.	**ASCII Express,** Bill Blue, Southwestern Data Systems
—	—	**Data Capture 4.0,** David Hughes and George McClelland, Southeastern Software

Hobby 10

This Month	Last Month	
1.	2.	**DOS Tool Kit,** Apple Computer
2.	1.	**Utility City,** Bert Kersey, Beagle Brothers
3.	6.	**Zoom Grafix,** Dav Holle, Phoenix Software
4.	4.	**DOS Boss,** Bert Kersey and Jack Cassidy, Beagle Brothers
5.	5.	**Locksmith 4.0,** Omega Microware
6.	3.	**DOS 3.3,** Apple Computer
7.	—	**Super Disk Copy III,** Chuck Hartley, Sensible Software
8.	8.	**Alpha Plot,** Bert Kersey and Jack Cassidy, Beagle Brothers
9.	—	**Multi-Disk Catalog,** Chuck Hartley, Sensible Software
—	—	**The Complete Graphics System,** Mark Pelczarski, Penguin Software

Softalk Presents The Bestsellers

Express and *Data Capture 4.0* continue neck-and-neck in the communications environment.

Biggest change in the Hobby 10 list was the continued climb of *Zoom Grafix*. The program edged its way into the Top Thirty as well as reaching third on the Hobby 10. Also notable was the decline of *DOS 3.3* to sixth, perhaps indicating that the changeover to the new operating system is finally nearing completion.

As one might suppose, *Wizardry* dominated the Fantasy 5 game list. But a new contender hove into view in Bob Clardy's *Apventure to Atlantis*, which wrested second place from *Ultima*.

Biggest news in the Adventure 5 category was the release of *Time Zone*, the magnum opus of On-Line Systems. The program was generally favorably received despite its high price and the well-publicized existence of some bugs in the first release version.

Castle Wolfenstein continued to lead the Strategy 5 list, but two old favorites—*Sargon II* and *Robot War*—regained the list in second and third place. Also new to the list was *Southern Command*, Strategic Simulation's most successful effort at incorporating hi-res graphics into a war simulation program.

Apple-franchised retail stores representing approximately 6.5 percent of all sales of Apples and Apple-related products volunteered to participate in the poll.

Respondents were contacted early in March to ascertain their sales leaders for the month of February.

The only criterion for inclusion on the list was number of sales made—such other criteria as quality of product, profitability to the computer retailer, and personal preference of the individual respondents were not considered.

Respondents in March represented every geographical area of the continental United States.

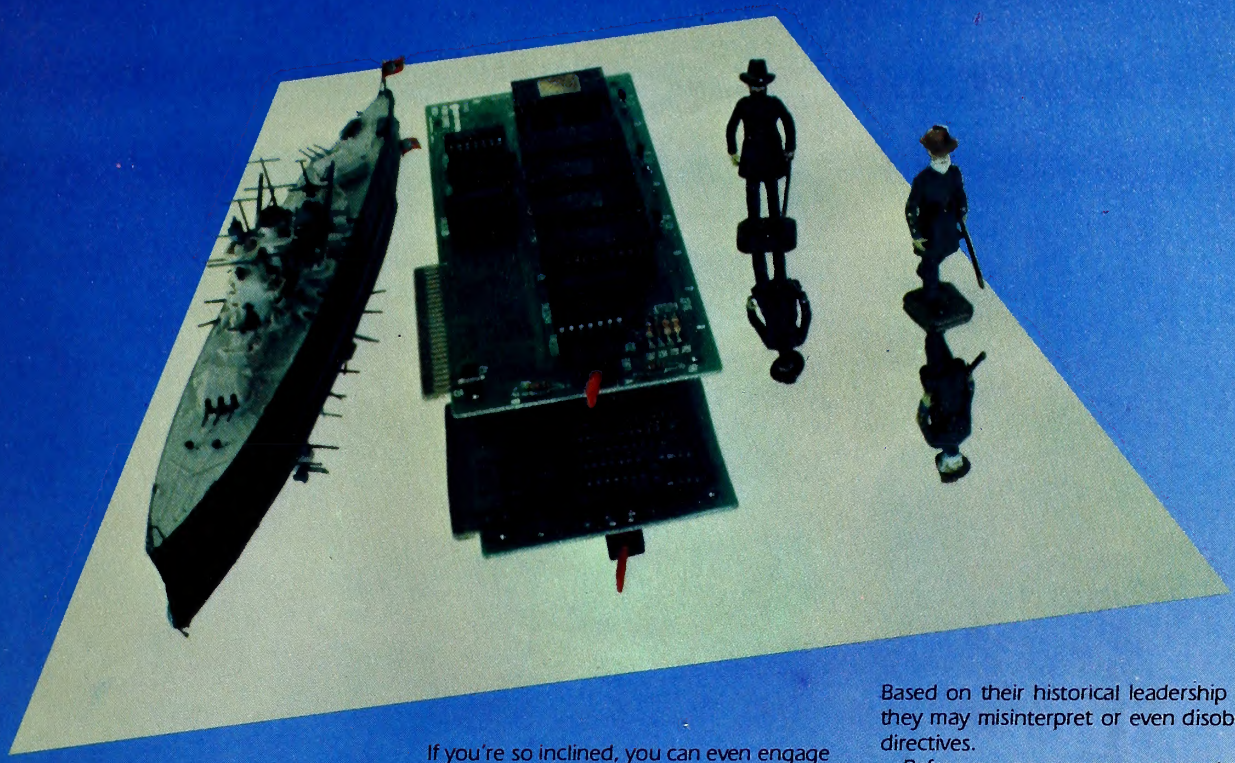
Results of the responses were tabulated using a formula that resulted in the index number to the left of the program name in the Top Thirty listing. The index number is an arbitrary measure of relative strength of the programs listed. Index numbers are correlative only for the month in which they are 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 5.4 percent, which translates roughly into the theoretical possibility of a change of 3.89 points, plus or minus, in any index number.

The Top Thirty

This Month	Last Month	Index	
1.	1.	112.42	**VisiCalc,** Software Arts/Dan Bricklin and Robert Frankston, VisiCorp
2.	5.	95.34	**Wizardry,** Andrew Greenberg and Robert Woodhead, Sir-tech
3.	2.	69.25	**Superscribe II,** David Kidwell, On-Line Systems
4.	3.	67.45	**Personal Filing System,** John Page, Software Publishing Corporation
5.	4.	62.06	**Snack Attack,** Dan Illowsky, DataMost
6.	8.	58.91	**Home Accountant,** Bob Schoenburg, Larry Grodin, and Steve Pollack, Continental Software
7.	9.	52.61	**DB Master,** Alpine Software/Stanley Crane and Jerry Macon; and Barney Stone, Stoneware
8.	16.	49.47	**BPI General Ledger,** John Moss and Ken Debower, Apple Computer
9.	10.	47.67	**VisiTrend/VisiPlot,** Micro Finance Systems/Mitch Kapor, VisiCorp
10.	7.	45.87	**David's Midnight Magic,** David Snider, Broderbund Software
11.	6.	44.52	**VisiFile,** Creative Computer Applications/Colin Jameson and Ben Herman, VisiCorp
12.	13.	44.07	**Castle Wolfenstein,** Silas Warner, Muse
13.	18.	38.67	**Gorgon,** Nasir, Sirius Software
14.	—	33.73	**Star Blazer,** Tony Suzuki, Broderbund Software
15.	—	32.83	**Twerps,** Dan Thompson, Sirius Software
16.	15.	31.93	**Personal Finance Manager,** Jeffrey Gold, Special Delivery Software, Apple Computer
17.	—	30.58	**Track Attack,** Chris Jochumson, Broderbund Software
18.	11.	27.88	**Apple Panic,** Ben Serki, Broderbund Software
	—	27.88	**Horizon V,** Nasir, Gebelli Software
20.	—	27.43	**Magic Window,** Gary Shannon and Bill Depew, Artsci
21.	27.	26.98	**DOS Tool Kit,** Apple Computer
22.	25.	26.08	**Utility City,** Bert Kersey, Beagle Brothers
23.	23.	25.63	**Olympic Decathlon,** Tim Smith, Microsoft
24.	12.	25.18	**Sneakers,** Mark Turmell, Sirius Software
25.	—	24.73	**Bug Attack,** Jim Nitchals, Cavalier Software
26.	—	24.28	**Typing Tutor,** Image Producers, Microsoft
27.	—	22.93	**Zoom Grafix,** Dav Holle, Phoenix Software
28.	—	21.58	**Data Factory,** Bill Passauer, Micro Lab
29.	—	21.13	**DOS Boss,** Bert Kersey and Jack Cassidy, Beagle Brothers
30.	14.	20.24	**Apple Writer,** Apple Computer
	28.	20.24	**WordStar,** MicroPro

WE BRING THE FUTURE TO THE PAST.



The future belongs to microcomputers.

At SSI, we apply tomorrow's programming techniques to re-create some classic battles of the past. As proof, we offer our latest games for the Apple® computer*: PURSUIT OF THE GRAF SPEE and THE ROAD TO GETTYSBURG.

PURSUIT OF THE GRAF SPEE™ transports you back to the South Atlantic of 1939 so you can command the deadly German pocket battleship to sink enemy merchant vessels while eluding British and French warships. Or you can direct the Allied fleet to locate and destroy the scourge of your shipping lanes.

If you're familiar with **Computer Bismarck** — which has become something of a phenomenon in gaming circles — you owe it to yourself to try this game. Like its older brother, PURSUIT OF THE GRAF SPEE has all the thrill of the chase, heightened by the complexity and richness of detail such as complete ship ratings for speed and strength, fuel restrictions, refueling capabilities, and realistic rules governing sighting and visibility.

However, we've also added a remarkable tactical combat system that accounts for gunfire accuracy, torpedoes and smoke screens. When it comes to calculating ship damage and battle results, you simply leave all the work to the computer, which does everything instantly and unerringly.

If you're so inclined, you can even engage in combat by using miniatures. Enter the results of your off-line battle and the computer will incorporate them into the strategic game.

This simulation contains two scenarios, all played on a 19-by-19 square-grid map of the South Atlantic displayed in colorful Hi-Res graphics. The first recounts the entire 25-day historic chase, where shadowing, pursuit and elusive maneuvers are your primary concerns. The second simulates the Battle of the River Plate, where the Graf Spee — already located — must deal with three British ships.

Finally, the solitaire scenario lets you match wits against the computer as it directs the Graf Spee. We wish you luck.

THE ROAD TO GETTYSBURG™ takes you even further back in time, to that fateful battle of 1863. This board-assisted computer simulation is designed to let you experience the actual feel of a Civil War command. Like a real Union or Confederate general, you must deal with the slow and frustrating dispatch system — your only means of communication with your troops. Knowledge of troop positions and estimates of enemy strength are only as good as your reconnaissance patrols.

The computer plays the individual corps commanders who implement your orders. Based on their historical leadership ratings, they may misinterpret or even disobey your directives.

Before you move a corps, you must consider the effects of weather, terrain, corps morale, fatigue, and even straggling on your troops.

THE ROAD TO GETTYSBURG simulates the entire week-long campaign. Depending on your strategies, the town of Gettysburg may not be involved at all.

For those who want to dispense with long-range planning, we've provided a scenario that just simulates the three-day Battle of Gettysburg. There is also a solitaire scenario so you can play anytime against a tough computer opponent.

Again, you can use miniatures to resolve all battles if you so desire.

This game is made for all you avid strategists out there. After you're done playing it, you'll really know the pleasure and pain of being a Robert E. Lee!

If you're interested in what the future has in store for the past, visit your local computer/game store and get these SSI games today. At $59.95 each, they must surely be the best bargain in time travel!

To order with your VISA or MC, call **toll free 800-227-1617, ext. 335** (800-772-3545, ext. 335 in California). To order by mail, send your check to Strategic Simulations Inc, 465 Fairchild Drive, Suite 108, Mountain View, CA 94043. All our games carry a 14-day money back guarantee.

*48K Apple® II with ROM and one floppy disc drive.

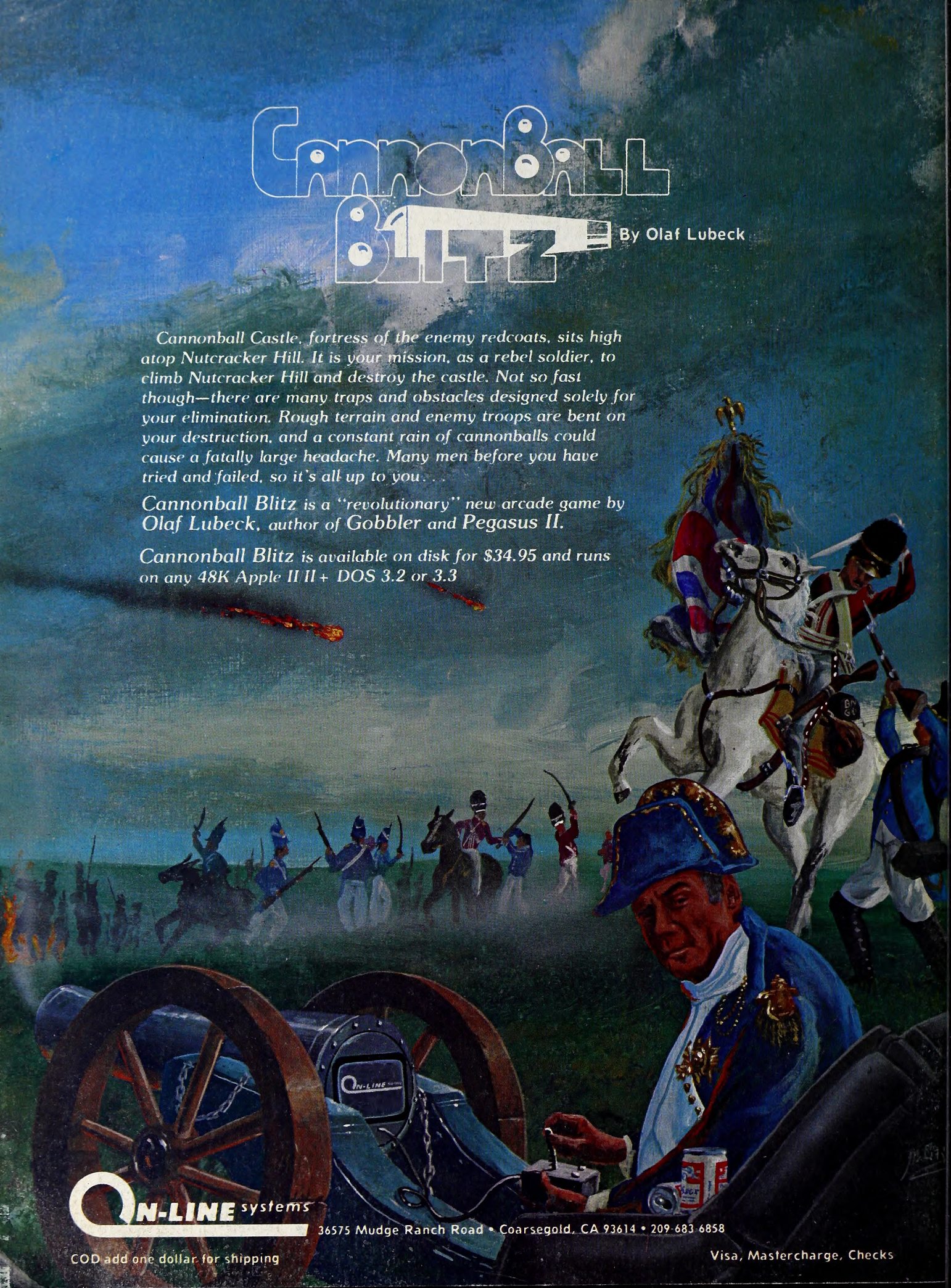
SSI

STRATEGIC SIMULATIONS INC

Apple is a registered trademark of Apple Computer Inc.

As part of our demanding standards of excellence, we use maxell floppy discs.

WE'RE LOOKING FOR A FEW GOOD DESIGNERS. We're always on the lookout for high-quality games from independent designers. If you have game programs that you'd like SSI to publish, give us a call at (415) 964-1353 or write to SSI, 465 Fairchild Drive, Suite 108, Mountain View, CA 94043.


CANNONBALL
BLITZ
By Olaf Lubeck
Cannonball Castle, fortress of the enemy redcoats, sits high atop Nutcracker Hill. It is your mission, as a rebel soldier, to climb Nutcracker Hill and destroy the castle. Not so fast though—there are many traps and obstacles designed solely for your elimination. Rough terrain and enemy troops are bent on your destruction, and a constant rain of cannonballs could cause a fatally large headache. Many men before you have tried and failed, so it's all up to you...
Cannonball Blitz is a "revolutionary" new arcade game by Olaf Lubeck, author of Gobbler and Pegasus II.
Cannonball Blitz is available on disk for $34.95 and runs on any 48K Apple II II+ DOS 3.2 or 3.3
ON-LINE systems
36575 Mudge Ranch Road • Coarsegold, CA 93614 • 209-683-6858
COD add one dollar for shipping
Visa, Mastercharge, Checks