SoftSide™

#3

THREE DOLLARS

Personal Finance

A SPEECH SYNTHESIZER ON A DISK!

That's why Apples and Ataris are saying: "Talk Is Cheap"

IT'S CALLED THE SOFTWARE AUTOMATIC MOUTH, S.A.M. FOR SHORT

It's a high quality speech synthesizer created entirely in software. You use it as a software utility, load it into RAM, and then use your machine as usual, except now you can make your programs talk. It generates the speech sounds on demand, so there is no limit to what it can say.

When you hear S.A.M., you'll probably agree that it sounds better than all the hardware speech synthesizers for Apple or Atari computers. And, it has a truly remarkable price.

YOU CONTROL INFLECTION, PITCH AND SPEED

With its user-variable inflection, S.A.M. can accent words on the right syllable and emphasize the important words in a sentence.

You can also make S.A.M.'s speech higher or lower, and faster or slower, over a wide range of settings.

USE EASY PHONETIC INPUT OR PLAIN ENGLISH TEXT

S.A.M. understands a simple phonetic spelling system, not a mysterious alpha-numeric code. S.A.M. helps you learn phonetic spelling by showing you your mistakes, and the owner's manual gets you started with an English-to-phonetics dictionary of 1500 words. So it's easy to make S.A.M. produce exactly the sounds you want.

But suppose you want to type ordinary English, or you want your machine to read a word processor file aloud. The S.A.M. disk comes with RECITER, an English text-to-speech conversion program that lets S.A.M. speak from plain English text.

ADD SPEECH TO YOUR PROGRAMS WITH EASE

In a BASIC program, you add speech with just a couple of commands. In a machine language program, it's just as easy. S.A.M. comes with four demonstration programs to show off its distinctive features and help to write your own talking programs. Write adventure games with talking characters, educational programs that explain aloud, or utilities with spoken prompts – put your imagination to work.

S.A.M.

S.A.M. programmed by Mark Barton.

You can order S.A.M. directly from DON'T ASK. Add $2.00 for shipping and handling to your check or money order (or order C.O.D.)

S.A.M. for the Apple II/II+
Includes an 8-bit digital-to-analog converter and audio amplifier on a board. Only **$124.95**
Requires 48K, disk. (S.A.M. takes up to 9K; RECITER 6K.) You will also need a speaker.

S.A.M. for the Atari 400/800
S.A.M. talks through your television speaker. No additional hardware is required. Only **$59.95**
Requires 32K, disk. (S.A.M. takes up 9K; RECITER 6K.) Note: to produce the highest quality speech, S.A.M. automatically blanks the screen during vocal output; the display is preserved. S.A.M. can talk with the screen on, but the speech quality is reduced.

DON'T ASK INC.
COMPUTER SOFTWARE

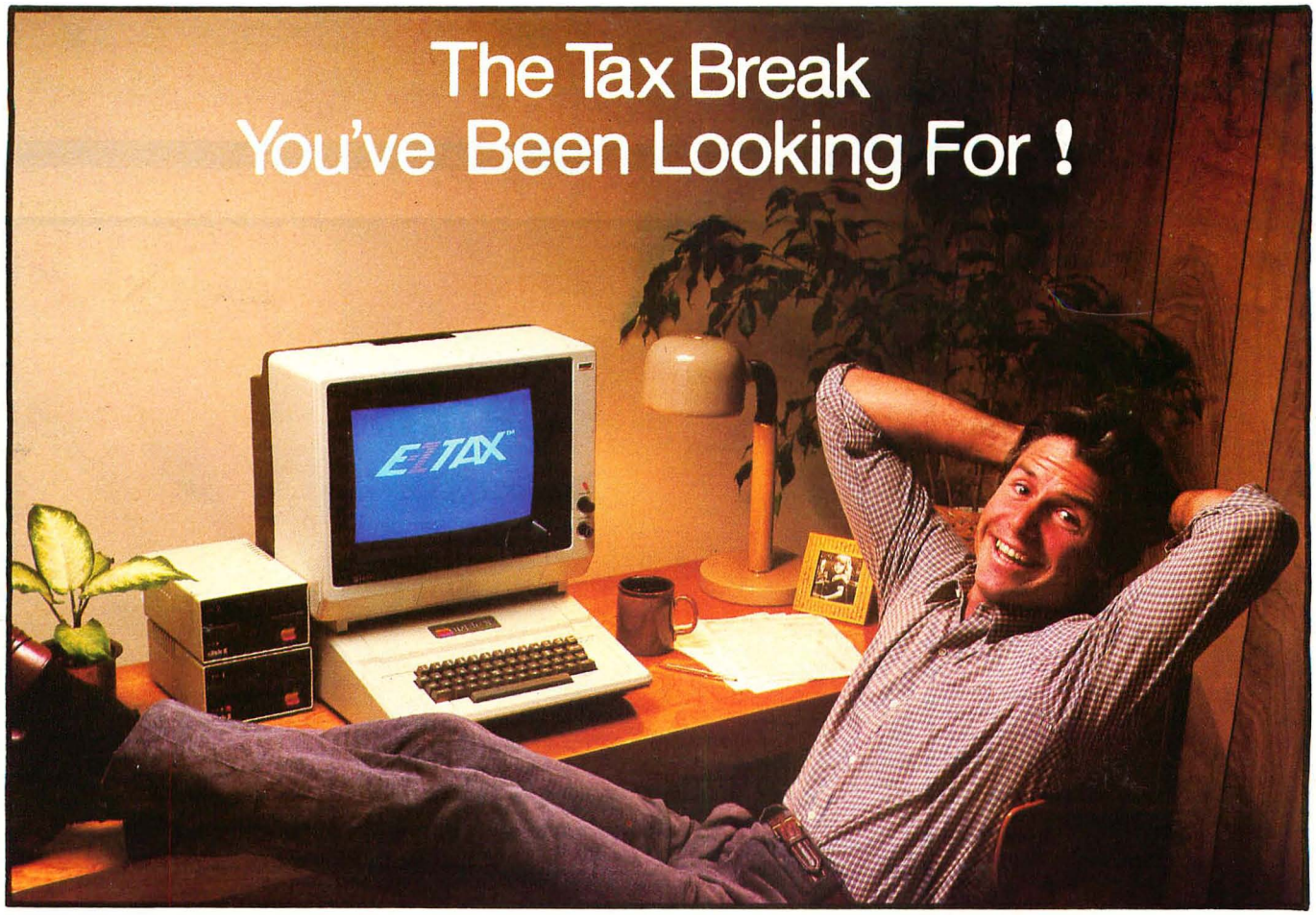
2265 Westwood Boulevard, Suite B-150
Los Angeles, California 90064
Telephone: (213) 397-8811

Hear S.A.M. at your favorite computer store today!
Dealer inquiries welcome.

Production: AUTOSTYLE GRAPHICS (213) 393-5728

Apple is a trademark of Apple Computer, Inc. ● Atari is a trademark of Atari, Inc.

The Tax Break You've Been Looking For !



You Just Found It!

E-Z Tax. The simplest tax preparation software ever developed was designed for your Apple II personal computer.

Now you can prepare your own tax return without **any** knowledge of taxes or computer programming. From the moment you insert the E-Z Tax floppy disk, you'll be in full control. Every question is self-prompting and nothing is overlooked.

If you make a mistake, the program lets you know about it immediately. If you need tax help, just press a button and you'll get the answer. *Its simply the most amazing tax preparation software ever.*

COUPON

Please send me the following # of kits requested:

_____ APPLE II _____ IBM PC

_____ ATARI 400 & 800 _____ CP/M

_____ **TOTAL REQUESTED**

x $69.95 each

_____ Total

_____ Plus Postage & Handling ($4/kit)

_____ Plus C.O.D. Charges ($3/kit)

_____ **TOTAL ORDER**
(Enclose payment for this amount.)

ACT NOW!

Send: ☐ Check ☐ Money Order ☐ C.O.D.
Charge my credit card: ☐ Visa ☐ Mastercard

Card #_____ Exp. Date _____

Signature_____

Name_____

Address_____

City_____ State _____ Zip _____

Mail this coupon to: **TAX HELP, INC.**

Prints on Federal Forms

When you're finished, E-Z Tax will print out your tax return on official federal forms. If you don't have a printer, just fill in the forms from the data on the screen.

If you need help, you can call E-Z Tax's toll free customer service phone number.

E-Z Tax prepares the following IRS forms and schedules:

1040A	2106
1040 EZ	2119
1040 page 1 & 2	2210
Schedule A	2440
Schedule B	2441
Schedule C	3468
Schedule D	3903
Schedule E	4137
Schedule F	4684
Schedule G	4972
Schedule R/RP	5695
Schedule W	6251
1040 ES	6252
1040 SE	

ACT NOW!

You just found the tax preparation program you've been looking for. Now here's how you can get your hands on it . . .

- *Fill* in the coupon, or
- Call toll-free to order over the phone. Just give the operator your credit card number or request a C.O.D. shipment.

Only **$69.95**
TAX DEDUCTIBLE

Your E-Z Tax Kit Includes. . .

- E-Z Tax Software Program (2 Disks)
- E-Z Tax Guide Book
- Over 35 Official Federal Tax Forms for 1982 Tax Returns
- Tax Organizer Envelopes
- Instruction Guide
- Warranty Card

E-Z TAX

Distributed By

TAX HELP™ INCORPORATED

BOX 7676
SAN JOSE, CA 95150
(408) 998-1040
WATS LINE: (800) 331-1040 - USA
(800) 344-1040 - CA

SoftSide™

EDITOR-IN-CHIEF
Randal L. Kottwitz

SOFTWARE MANAGER
Bill Kubeck

MANAGING EDITOR
Carolyn Nolan

ASSOCIATE EDITOR
Joyce Smith

PROGRAMMING STAFF
Rich Bouchard
Alan J. Zett
Fred J. Condo
Kerry Shetline

CONTRIBUTING EDITORS
Cary Bradley
Fred D'Ignazio
Peter J. Favaro
Ame Choate Flynn
Sheldon Leemon
Lance Micklus
Allen L. Wold

ART DIRECTOR/
PRODUCTION MANAGER
Lynn Wood

PRODUCTION STAFF
Lynda Fedas
Denise Chartrand

ADVERTISING
Sue Rowland
Christopher Smith

DEALER SALES
Nancy Broderick

CIRCULATION
Cindy Schalk
Cindy Zawacki
Donna Jean

STAFF
ACCOUNTING, **Doris Miller**
ACCOUNTING, **Karen Lawrence**
DUPLICATION, **Jeffrey Garrod**
EDITORIAL, **Suzanne W. Wellington**

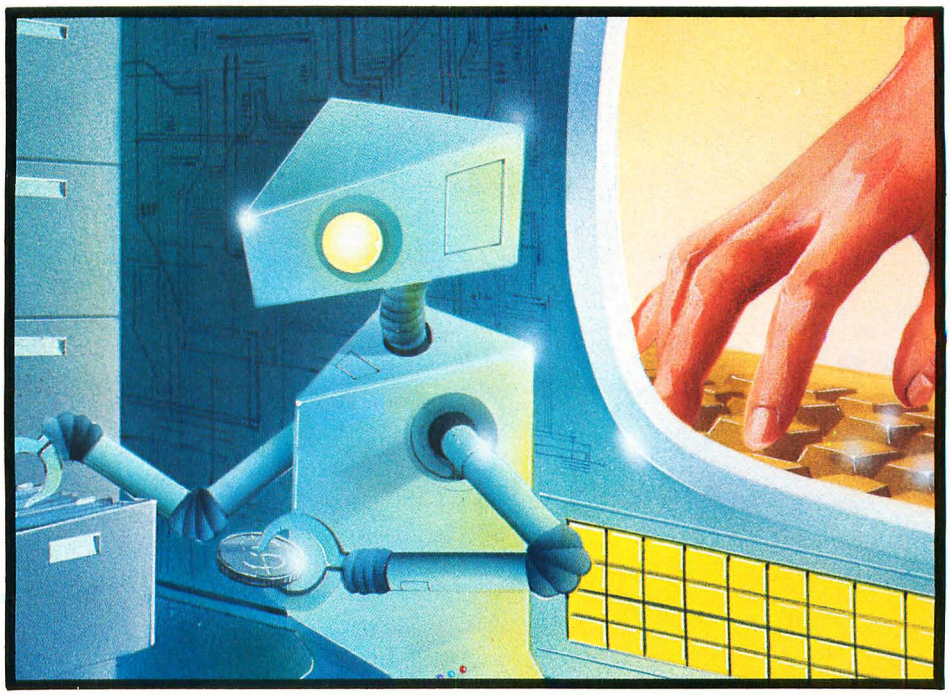
ASSOCIATE PUBLISHER/
CIRCULATION MANAGER
Nancy Lapointe

PUBLISHER
Roger W. Robitaille Sr.

SoftSide Vol. 6, No. 4

SUBSCRIPTION INQUIRIES should be sent to ***SoftSide*** Publications, 100 Pine Street, Holmes, PA 19043. EDITORIAL AND ADVERTISING CORRESPONDENCE should be sent to ***SoftSide*** Publications, 6 South Street, Milford, NH 03055. Telephone (603) 673-0585.

SoftSide (ISSN 0274-8630) is published monthly by ***SoftSide*** Publications, Inc., 6 South Street, Milford, NH 03055. Printed at Wellesley Press, Framingham, MA. Second class postage paid at Milford, NH, and additional mailing offices. Subscription rates: US and Canada, $30/12 issues. First Class US, APO, FPO, Mexico, $40/12 issues. Other foreign countries, $62/12 issues. Media subscription rates: US Magazine and Cassette, $75/12 issues. US Magazine and Disk $125/12 issues. APO, FPO, Canada and Mexico, add $20/12 issues. Other foreign add $50/12 issues. All remittances must be in U.S. funds. Entire contents Copyright © ***SoftSide*** Publications, Inc., January, 1983. All rights reserved. POSTMASTER: Please send form 3579 to ***SoftSide*** Publications, 100 Pine Street, Holmes, PA 19043.



FRONT RUNNER

FEATURES

DEPARTMENTS

CONTENTS

PC/SIDE

TRS-80®/SIDE

ATARI®/SIDE

APPLE™/SIDE

Apple™, ATARI®, IBM®, and TRS-80® are registered trademarks of The Apple Computer Company, Warner Communications, International Business Machines Corporation and the Tandy Corporation, respectively. Envyrn, Envyrnment, Envyrnese and diversions thru Envyrn are registered trademarks of Roger W. Robitaille, Sr.

*Available on DV — Enhanced Disk Version (See bind-in card to order).
**Available on DV — Enhanced Disk Version and CV — Cassette Version (See bind-in card to order).

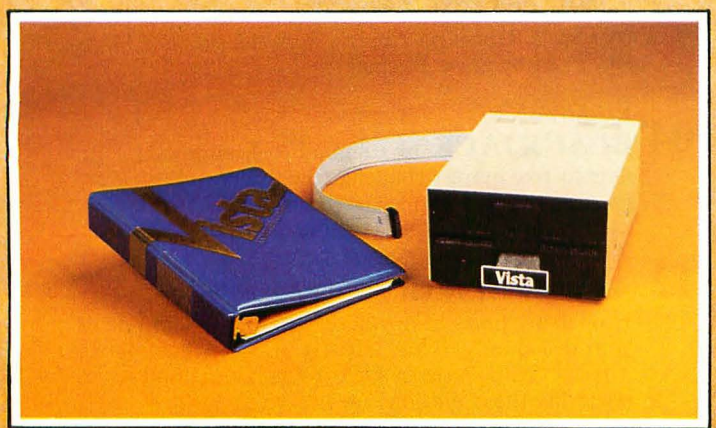
Vista DaVinci
GREAT INNOVATORS

In Italy in the 15th century Leonardo Da Vinci was a multi-talented genuis. Leonardo's ideas and designs were endless. He was always creating new devices to solve the world's problems. Pictured here are many studies for military arms and armor and battlements.

Today Vista is developing many high technology computer peripherals for use with IBM™ and Apple Computers.

Introducing Solo, a new cost effective solution to Apple Disk storage offering all the capabilities of the standard Apple Disk II at a fraction of the cost.

If your needs do not demand the high performance of our Duet, Quartet, or V-Series Maxidrives, then Solo will provide you with what you need at the lowest possible cost.



Vista Solo™

The Vista Solo incorporates the proven reliability of the Shugart™ 5¼ Drive. To date Shugart has over one million units in operation throughout the world.

Solo is also available as the Solo Plus, which comes with the Solo Controller. The pair give you total compatibility with Apple hardware and software.

- Shugart Drive
- Totally Compatible
- Cost Effective
- 35 Track Compatible Drive
- 143K Storage
- Full Vista 120 Day Warranty

$299.00

Manufacturer's suggested list price

Solo (Drive only) #3101
Solo Plus (w/Controller) #3111

Contact Your Local Vista Dealer or Call our Vista Hotlines.

Vista COMPUTER COMPANY, INC.

1317 East Edinger / Santa Ana, CA 92705
(714) 953-0523

DISTRIBUTORS

Western: Group III Electronics
(213) 973-7844
(408) 732-1307

South Central: M P Systems
(214) 385-8885

South: Digitek of America Inc.
(504) 466-0894

Central: Wyatt & Associates
(317) 773-4791

Northeast: Computer & Peripherals, Inc.
(315) 476-6664

Northwest: National Micro Wholesale
(503) 779-6839

™Shugart is a registered trademark of Shugart Associates.
™IBM is a registered trademark of International Business Machines.
™Apple II is a registered trademark of Apple Computer Co.

Hardware Wars — The Movie

One rather bleary-eyed Saturday night when I was in college, I watched a short film clip on *Saturday Night Live* called "Hardware Wars." It was a take-off on *Star Wars*, with flying steam irons taking pot shots at electric can openers and blenders. As I've read what's happened to the computer marketplace in 1982 and what's expected to happen in 1983, I can't help but think of a lengthy version of the same film, with various micros battling it out for supremacy in the galaxy.

Many issues are at stake in this battle — software compatibility, 8 bit vs. 16 bit, bundled software, 3½ vs. 5¼ inch disks, and even a feud between microprocessor families. It would give a screen play author plenty of conflict on which to base a script.

Here are some observations and industry rumors pertaining to the computer manufacturers we think will interest the *SoftSide* reader:

Apple has been a long time coming with a new machine. After the embarrassment of their introduction of the Apple III to a rather lukewarm reception, they must have decided to make doubly sure that their next machine was ready before its unveiling. By the time you read this, the Apple IV (code-named Lisa) may have debuted. Early reports say it's a phenomenally user-friendly, business oriented machine priced between $10,000 and $20,000. (That's a wide margin, but it depends on whose reports you read.) They hope to have a more consumer oriented version on the market by the end of the year called the Mackintosh, priced in the $2,000 range. In the meantime, they are introducing the Super II+ (or IIe) for the consumer. Essentially, it appears to be a II+ update that solves some of the problems from which the II+ has suffered.

Atari filled 1982 with rumors of the 600, 1200 and "Sweet 16," and very little other information. In mid-December (some say as a stop-gap measure due to Warner Communications' sharp drop in stock prices), they introduced the 1200XL. It, like Apple's IIe, appears to be an update to the 800. It has 64K, one cartridge slot, two joyports, four programmable function keys which can be toggled to three configurations for a total of twelve functions, and a seven-minute run of self-diagnostics when you turn on the machine. Said to be totally software compatible with the 400 and 800, its price has yet to be announced, but will be somewhere between $750 and $1,000. Concurrently, they announced a new 80-column printer (a variation on one of the Microlines), a new tape drive, and a four-color, 40-column printer/plotter with sixteen pens for $299. All of these are due for shipment in March. They plan to introduce a high density disk drive sometime in 1983 and rumors abound of an Atari 600 to be similar to the 400 with a full-stroke keyboard.

Tandy has announced very little in 1982. They introduced the Model 16, a 16-bit machine with applications primarily in the business market, and an upgrade package for the Model II to make it a Model 16. More significantly, they announced a marketing move to repackage the Color Computer under the name TDP-System 100 and sell it through RCA dealers. However, we have yet to see these machines in the stores or any advertising support.

In keeping with the big blue's tradition, practically no rumors are coming out of IBM. All we've heard is that they are preparing one or several low-cost entries, priced with the consumer in mind. If they follow with as big a blast as the PC, the impact on the marketplace could be substantial.

In other random rumors, 1983 is to be the year Ma Bell enters the micro arena. Commodore is said to be considering a substantial price cut for the 64, making it stiff competition for Texas Instruments, Atari and its own VIC 20. As has been the case for two years, the Japanese are said to be looming just over the horizon, like the Death Star waiting to make its attack on a rebel planet.

I don't know about you, but I'm waiting for the sequel, "The Software Strikes Back."

Randal L. Kottwitz
Editor-in-Chief

computer case company

Attache-style cases for carrying and protecting your complete computer set-up. Accommodates equipment in a fully operational configuration. Never a need to remove equipment from case. Simply remove lid, connect power, and operate.

AP101	Apple II with Single Drive	$109
AP102	Apple II with Two Disk Drives	119
AP103	Apple II, 9 Inch Monitor & Two Drives	129
AP104	Apple III, Two Drives & Silentype Printer	139
AP105	13" Monitor with Accessories	99
AP106	AMDEK Color Monitor	119
RS201	TRS-80 Model I, Expansion Unit & Drives	109
RS204	TRS-80 Model III	129
AT301	ATARI Computers with Peripherals	109
P402	Centronics 730/737 & Radio Shack Printer	89
P403	Epson MX70/80 or Microline 82A	89
P404	Epson MX100 Printer	99
P405	IDS 560 or Prism 132 Printer	109
P406	Starwriter/Printmaster F-10 Printer	119
P407	Okidata Microline 83A or 84 Printer	99
P408	Prowriter 2 Printer	99
P409	Prowriter (Apple Dot Matrix) Printer	89
IB501	IBM Personal Computer	129
IB502	IBM Monitor	99
HP601	HP41 with Accessories	99
CM703	Commodore Model 64 with Drives	119
CM704	Commodore Model 64 with Dataset	109
NS010	North Star Advantage	139
CC80	Matching Attache Case (5")	85
CC90	Matching Attache Case (3")	75
CC91	Matching Accessory Case	95
CC92	5.25" Diskette Case	49

computer case company
5650 Indian Mound Court
Columbus, Ohio 43213
(614) 868-9464

CALL TOLL FREE
800-848-7548

INPUT/OUTPUT

From our readers

INPUT

A Book With No Pages

Dear *SoftSide,*

I just finished reading Mr. Wold's "A Book with No Pages" in *Entertainment Tomorrow, SoftSide* 34. He ends with "...I want my computerbook today." Mr. Wold, who else is in a better position than you to produce, if not a computerbook, at least a computer magazine article?

Why don't you, the editors of ***SoftSide,*** further the cause of paperless publication by distributing some articles only on magnetic media? To paraphrase Mr. Wold, you will need a program to present the text, but it will not need to be very complex. In fact, of the list of functions he presents for his hypothetical keyboard, only scroll/stop and back up (reverse) would be needed. You might want to add one he seems to have left out, a table of contents. How much of the expense of producing ***SoftSide*** is from the preparation of the print as opposed to the material that is printed? If authors send you their material already prepared in machine readable form, you could distribute "literature" which has never appeared on paper! (Any spoilsports who want to see your paperless magazine printed could print your magazine text files with their own word processors.) You could even explore the possibility of trying another "first" — DV Market/Side. Don't Ask Computer Software could have their speech synthesizer, S.A.M., speak their ad, telling us that "Talk is cheap" and letting us decide if we want to hear more. The possibilities are endless.

Joan Bixby Dunham
Silver Spring, MD

Editor's Reply: I always find it interesting that our readers can anticipate our plans for the future before we announce them. It happens more often than you might imagine. Indeed, we plan to start offering special text files, only on DV, sometime in the next year. We are currently developing a system which will be even more interactive than that described by Mr. Wold in his article. We hope to offer not only an interactive table of contents, but several levels of more detailed information accessible from within the text itself. Imagine reading an article on whales which makes a reference to "baleen." If you want more information about the term, you can simply move the cursor over the word and ask for it. The most substantial stumbling block we see in the implementation of such a system is that it will require authors to think of their text in a much more parallel than serial type of format.

Computers In the Classroom

Dear *SoftSide,*

My compliments on producing a fine magazine. Perhaps you or your readers can help me.

This fall an Apple II+® was installed in my fifth grade classroom. The kids and I are learning all sorts of wonderful things. I suffer from one real problem that can best be solved with some conversations with others who have struggled, or are struggling, with the same problem: How do I integrate the computer into the daily flow of activities? Computer management in the classroom is giving me fits! I would like to talk to some folks who have some ideas to share. And, one more question: Can I convert Integer BASIC programs to Applesoft? How?

Warner Lord
Madison, CT

Editor's Reply: We're happy to offer this column as a forum for discussion on the topic of computer integration in the classroom. I hope your letter generates some helpful replies. As for converting programs from Integer BASIC to Applesoft — it can be done, but is as complicated as translating a program from Atari BASIC to Applesoft. Good luck!

Atari® Microsoft

Dear *SoftSide,*

A letter in *Hints & Enhancements* of issue 34 gives an extremely erroneous impression of Atari Microsoft BASIC. I own seven microcomputers, four of them with Microsoft BASIC, and **none** of those allow the MID$() function to appear on the left-hand side of an equation. When writing Adventures, for example, on Radio Shack, Ohio Scientific and Commodore computers, I have **always** had to resort to the method illustrated in the column when I wish to change the contents of a string.

Even if this feature is available on an enhanced version of Microsoft BASIC, such as TRS-80® Disk BASIC, or the IBM® PC Disk BASIC, I don't consider that to be sufficient reason to call it a "standard" feature or a "normal" Microsoft BASIC command, and point to its absence from Atari Microsoft BASIC as a deficiency.

My point is that the letter gives the impression that there is something wrong with the Atari version of Microsoft BASIC, when it is actually exactly the same as that in probably 90% of your readers' systems.

Incidentally, the much maligned "standard" Atari BASIC **does** allow such string manipulation with its A$(x,y)= function, making such projects as Adventures **easier** with Atari BASIC than with my "standard" Microsoft Systems in this respect!

Robert J. Retelle
Ypsilanti, MI

Rotberg Synthesizer

Dear *SoftSide,*

Since I first subscribed to *SoftSide* (Atari® Disk Version) last June, I have anticipated each month's issue eagerly. That is no longer the case. Your issue 34, featuring *The Rotberg Synthesizer*, was a waste of money, in my opinion, and after what it did to some of my most valuable software today, I am angry beyond belief.

In the first place, the *Synthesizer* is nothing more than a demonstration playback gimmick; I can't even use it to write my own music. I bought the enhanced disk version to get good utility and game programs. The *Rotberg Synthesizer* is cute, but certainly not worth the additional cost. If I just wanted to listen to someone else's music, I would have purchased a record for my stereo.

I probably would have bottled-up my hostility over that program though, if it weren't for what happened today. I tried to boot up the *Synthesizer* with the BASIC language cartridge installed. When it gave me a READY prompt, I wasn't sure what was going wrong. Since I wasn't in love with the program anyway, I decided to

continued on page 8

A feast of computing ideas.

If you work with a 6502/6809-based system, you're probably hungry for the facts and ideas that will help you understand the inner workings of your computer. You want to go beyond canned software—use your computer for more than games—learn the advanced programming techniques that enable you to get the most out of your 6502/6809 system.

MICRO, The 6502/6809 Journal, gives you page after page, month after month, of solid information to sink your teeth into. **MICRO** is the premier how-to magazine for serious users of the Apple, PET/CBM, OSI, Atari, AIM, SYM, KIM, and all 6809 based systems including the TRS-80 Color Computer. It's a resource journal internationally respected by professionals in business, industry, and education. Every issue of **MICRO** keeps you informed with up-to-the-minute data on new products and publications:

- **hardware catalog** with organized, concise description
- **software catalog** in an easy-to-use format
- **new publications** listed and annotated
- **reviews and evaluations** of significant products

And there's much more:

- **In-depth hardware tutorials** bring expert advice into your home or office.
- **Detailed discussions of programming languages** deepen and broaden your programming ability.
- **Complete program listings** enable you to increase your machine's capabilities.
- **Bibliography of 6502/6809 information** helps you to find pertinent articles in a timely manner.
- **Special monthly features** with in-depth treatment of one subject or system increase your knowledge of the field.
- **Balanced mix of machine-specific and general articles** for your everyday use as well as long-range reference needs.
- **Informative advertising** focused specifically on 6502/6809 machines keeps you abreast of latest developments.
- **Reader feedback** puts you in touch with other micro-computerists.

MICRO is the magazine you need to get the most from your own 6502/6809 system!

You'll love every byte.

To order, send your check or international money order (payable to MICRO) and the order form at left, to:

Subscription Fulfillment
MICRO, Dept. MI
34 Chelmsford Street
P.O. Box 6502
Chelmsford, MA 01824

Or, for your convenience, call our toll-free number:

1-800-227-1617, Ext. 564

(In California, 800-772-3545, Ext. 564)

and charge your subscription to your MasterCard or VISA. (All orders must be prepaid in U.S. dollars or charged to your MasterCard or VISA.)

YES! I want to get more from my microcomputer. Please send me

___ year(s) of MICRO at $_____/year. (Outside U.S. and Canada, please indicate via ☐ surface or ☐ air mail.)

Name

Company

Street

City State Zip Code

☐ Check enclosed for $ _____
☐ Charge my credit card account
☐ VISA ☐ MasterCard

Signature

Card number

Expiration date

SUBSCRIPTION RATES (U.S. dollars)
Yearly subscription (ISSN 027-9002) saves 20% off the single-issue price.

U.S. $24*

Canada $27

Europe $27 ($42 by air mail)

Mexico, Central America, Mideast, North and Central Africa $27 ($48 air)

South America, Far East, South Africa, Australasia $27 ($72 air)

* SPECIAL OFFER—U.S. ONLY:
Save even more—30% off single-issue price: 2 years, $42

Dept. S S

ATTENTION AUTHORS

SoftSide Publications is actively seeking program, article and review submissions for the TRS-80®, IBM®-PC, Apple™ and ATARI® home computers.

- **Programs — SoftSide** has always been the leader in the field of BASIC software. BASIC remains our specialty. However, with the advent of Disk Version (DV), we can now also offer an outlet for Machine Language and multiple language programs which do not lend themselves to printed versions. Games, utilities and educational software, as well as any other applications for the home computer user are preferred, although we will consider virtually any type of program. Hybrid mixes of articles and programs are also welcomed.

Please be sure to include **full** documentation of subroutines and a list of variables, also a brief article describing the program.

- **Reviews** — Well written, informed reviews of all software for the systems we cover are a regular feature of **SoftSide.** Reviewers should take into consideration all aspects of a particular software package, from speed of execution to programming creativity to the estimated length of time that the product will hold the customer's interest.
- **Articles** — We welcome article submissions of all types, but prefer those specifically geared to the home computer market. We give our readers information as a first priority, but vary our content to include some humor and commentary.

All text, including documentation and descriptive articles for programs, should be typewritten and double-spaced. Extra monetary consideration will be given to articles and reviews submitted on disks (Scripsit, Super-Text II, etc.). Programs should be submitted on a good disk. TRS-80® BASIC programs should function under both Level II and Disk BASIC.

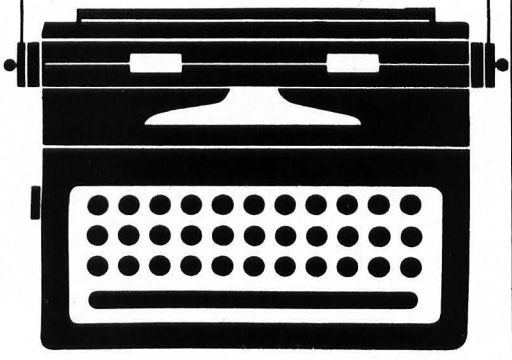
Please be sure to pack your disks carefully and to include your return address and phone number.

Send to:

SoftSide Publications
SUBMISSIONS DEPARTMENT
6 South Street
Milford, NH 03055

We regret that due to the volume we receive, we are unable to return submissions which do not include return postage.

Be sure to send for our **FREE AUTHOR'S GUIDE.** It further outlines the specifics of our submission procedure.



forget it. I then inserted a disk with three very important programs that I developed myself, and attempted to call up DOS. I got a "Please Standby" message and then one that read "Thank You. Your pirated copy has just been cleaned." Subsequent attempts to boot that disk caused the disk drive to run continuously. It had to be turned off with the "busy" light on. Worst of all, my programs which I spent nearly a week writing are lost.

I don't personally believe in copy protecting computer disks, but booby-trapping the disks you send me is another matter entirely. The documentation stated the disk was copy protected and that the user should not attempt to access its files. However, it didn't even begin to address what happened in my case. I certainly did not expect your programs to erase my software library.

Jack A. Zichterman
Loring AFB, ME

Editor's Reply: *The Rotberg Synthesizer* was included on the Atari DV, Issue 34, to show the sound potential of the Atari. In the same issue, we published *Pokey Player* as a music editing system for readers to write their own music. Our apologies for the damage done to your software. We included the strongly worded warning about accessing the files on the disk in order to prevent damage to readers' software. We did not anticipate the problem you encountered — our apologies. However, this situation points up how important it is to keep backup copies of your software — especially that which you develop yourself and which cannot be recovered from another source.

Roses and Thorns

Dear *SoftSide,*

I enjoy almost all of the programs in your magazine, and hope that you will continue to publish it for a long time. It is very hard to find a computer magazine of your calibre.

I have two or three minor complaints about your magazine. First, some of my friends have been receiving their copies of your magazine late. I realize that this is mostly the Post Office's fault, but you could send them out very early to get there on time. The second complaint is that I have problems typing in the data statements. I hope that in the future you might include source code and the manner which it should be typed in if I had the Atari Assembler Editor.

Keep up the good work.

Robert Lippmann
Woodmere, NY

Editor's Reply: We regret that the delivery date of second class mail is left to the discretion of the Post Office. We mail *SoftSide* at approximately the same time every month (with only occasional delays). As for including source code listings for our programs, it's only possible for very short routines. As you can see from every issue of *SoftSide*, the line listings eat up many pages of the magazine and we must publish our programs in the form usable to the greatest number of readers. There's simply not space to publish them in two different forms, especially when source listings are so space consuming.

OUTPUT

by Randal L. Kottwitz

Get out your checkbook, your bank statements, your credit card bills, and all the other financial litter gathering on your desk. Our special emphasis this issue is on personal finance. The time has come to put your financial affairs under control and we're going to do our best to help you do it.

As we were preparing this issue, several of the people we talked to made it clear they felt the computer had yet to offer a more convenient option for personal financial record keeping than pencil and paper. You'll even find that statement several places in this issue. However, one truth rings clear — using the computer to keep track of your financial affairs **forces** you to organize your records better in order to funnel them through a central point — the computer. In almost all cases, the software will demand some alteration in your current personal record keeping system. Could it be that the majority of people who complain about the inconvenience of the computer are the ones whose affairs are in the greatest disorder? Most financial management packages geared to the home are little more than intelligent database managers, and we all know that the most tedious element of using a database manager is setting up the initial database.

Our overall findings concerning these pieces of software (the commercial packages as well as the one we're publishing in this issue) have been good. The labor necessary to confine your organizational style to that of a good software package will be well worth the effort as you gain a greater understanding of your financial standing. We have some distance to go before the computer can accomplish the organizational miracles many salesmen claim for it. The most major changes must be made, not in the actual capabilities of the software, but in its user friendliness. But, with a little reorganization of our record keeping system, the personal financal management software available today can give us a much clearer understanding of the small business principles by which we must operate our personal financial matters. They won't make us money, but they can tell us where it's hiding.

Until next time, Happy Hacking!

Integrated, Menu-Driven File And Information System for Home or Office*...with PCHMS™, our Personal Computer Home/Office Management System.

Instantly put your IBM PC to work! PCHMS is a filing and information system that manages all those details that make your office, home and personal life smooth and efficient. Organize personal, household and business details—names and addresses, phone numbers, credit cards, home inventory, zip codes, medical and dental records, insurance policies, recipes, expense accounts, shopping and other lists, area codes, maintenance records and more.

*Note: Over half of our users have PCHMS™ in the office.

PCHMS is an integrated menu-driven system that allows you to add, delete, modify, or print out records in any file by full or partial key search. And you can create 10 full-feature user-defined files for anything you can think of. PCHMS provides all this plus a letter/memo-writer, built in printing calculator, 20 year calendar, constant display of date and time, alarm timer, worldwide time conversion, and metric/English converter. PCHMS files may be individually password protected. PCHMS runs in both monochrome and color.

And . . . that's not all . . .

You can add the PCHMS Auto-Dialer™. Automatically dial any stored phone number in your file with the press of a single key. Auto-Dialer gets you local, long distance, and international phone numbers, as well as networks such as Sprint and MCI.

Want more? Add the PCHMS Mailing List Module, the PCHMS Electronic Mail System, and the PCHMS Budgeting and Home Finance System.

PCHMS™ runs on an IBM Personal Computer with 64K of memory, at least one disk drive (single or double sided). Printer optional. PCHMS Auto-Dialer requires a Hayes Stack Smartmodem and RS232 card.

Ask for PCHMS™ and PCHMS Auto-Dialer™ at your IBM PC dealer or order directly from Arlington Software + Systems.
NOW AVAILABLE AT COMPUTERLAND STORES.

PCHMS and PCHMS Auto-Dialer are trademarks of Arlington Software + Systems. The Hayes Stack Smartmodem is a trademark of Hayes Microcomputer Products, Inc. IBM is a trademark of International Business Machines.

Dealer inquires invited.

PCHMS software is supplied on single sided diskettes together with comprehensive user documentation, backup utility and configuration program. Satisfaction Guaranteed!

☐ please send me _____ PCHMS @ $89.95 _____
☐ please send me _____ PCHMS Auto-Dial @ $69.95 _____
☐ please send me _____ PCHMS Mailing List @ $59.95 _____
Shipping and handling $ 3.50 _____
MasterCard and Visa welcome, please add 4% _____

Name: _____
Address: _____
City: _____ State: _____ Zip: _____
Phone: (_____)_____ Card Exp. Date: _____
Charge Card # _____
Signature: _____

97 Bartlett Ave. Arlington, MA 02174
(617) 641-0290

ARLINGTON Software + Systems

Bugs, Worms, and other Undesirables

Apple® *Hopper* Documentation Correction

The documentation for the Apple version of *Hopper (SoftSide* Issue 35) incorrectly listed the lines to be deleted to create the cassette version of the program. The correct lines to delete are 150 through *170,* and *580* through 600.

Apple *Fugue* Correction

Line 230 was omitted from the listing of Apple *Fugue (SoftSide* Issue 34). It should look like this.

```
230 COLOR=15: PLOT
ZA,VP(GG(N))
```

TRS-80® *Hopper* Correction

Five lines of data for the sound routine were omitted from the end of the listing of the TRS-80 version of *Hopper (SoftSide* Issue 35). They are reproduced here.

```
60120 DATA35,205,55,35,43,229,205,127,10
,42,33,65,58,-167,60
60130 DATA183,87,24,4,24,48,24,44,66,62,
9,211,255,16,252,66,62
60140 DATA10,211,255,16,252,58,64,56,230
,4,32,7,124,181,40,3,43
60150 DATA24,228,175,50,154,64,225,209,1
93,215,195,30,29,83,79
60160 DATA85,78,68,209,225,241
```

Alternatively, you may use line 60120-60160 from *Puzzle Jumble (SoftSide* Issue 34), which uses the same sound routine at the same line numbers.

HINTS & ENHANCEMENTS

From our readers

Apple® *CATS3* Enhancement

I am a teacher in a Middle School and I have two Computer Education classes each day. I was really excited to see the *C.A.T.S.* program appear in ***SoftSide.*** I am currently using it and I am recommending it to other educators.

After using the *C.A.T.S.3,* or SCORE, module, I found that the POKE CLR,0 (keyboard strobe clear) part of line 10010 turns off the printer after printing the first page of student results.

A simple fix for this is to change line 1290.

```
1290  PRINT : IF I / 18 =  INT (I
      / 18) THEN  PRINT : GOSUB 1
     0000: IF PF = 1 THEN  PR# 1:
      PRINT : PRINT
```

The value of the variable PF survives the POKE CLR,0 in line 10010.

I am like every other person who is a "user" first and a "programmer" second. Once I have something that works well, I always want just a bit more. I would like an easy way to correct typing errors or to reword a question after the test is stored in a text file. I would also like to be able to BLOAD a hi-res picture to go with some questions.

I know how much work it is to design, code, and then debug a program such as *C.A.T.S.,* and I do appreciate what you have accomplished. Please consider writing other useful educational programs in ***SoftSide.***

Robert Hofemann
San Jose, CA

Apple *Sabotage* Joystick Modification

Your magazine is terrific. I especially liked *Sabotage* in Issue 34. I changed it so you can use a joystick.

Here are the changes.

```
135 JX =  PDL (0):JY =  PDL (1)
140  IF JY < G8 THEN H = Z0:I = Z
     2:DX = H:DY = I: GOTO 190
150  IF JX > G9 THEN H = Z1:I = Z
     0:DX = H:DY = I: GOTO 190
160  IF JY > G9 THEN H = Z0:I = Z
     1:DX = H:DY = I: GOTO 190
170  IF JX < G8 THEN H = Z2:I = Z
     0:DX = H:DY = I: GOTO 190
220  IF (P0 > PN OR P1 > PN) AND
     M = Z0 THEN M = Z1:MX = X:MY
      = Y:XM = DX:YM = DY: POKE Y
     0,Z0
595 PN = 127:G8 = 50:G9 = 200
```

Paul C. Ossenbruggen
Durham, NH

Atari® *SWAT* Enhancement

Thanks for *SWAT.* It's made my favorite software magazine even better by removing all qualms about typing in large programs and being faced with the possibility of time-consuming debugging.

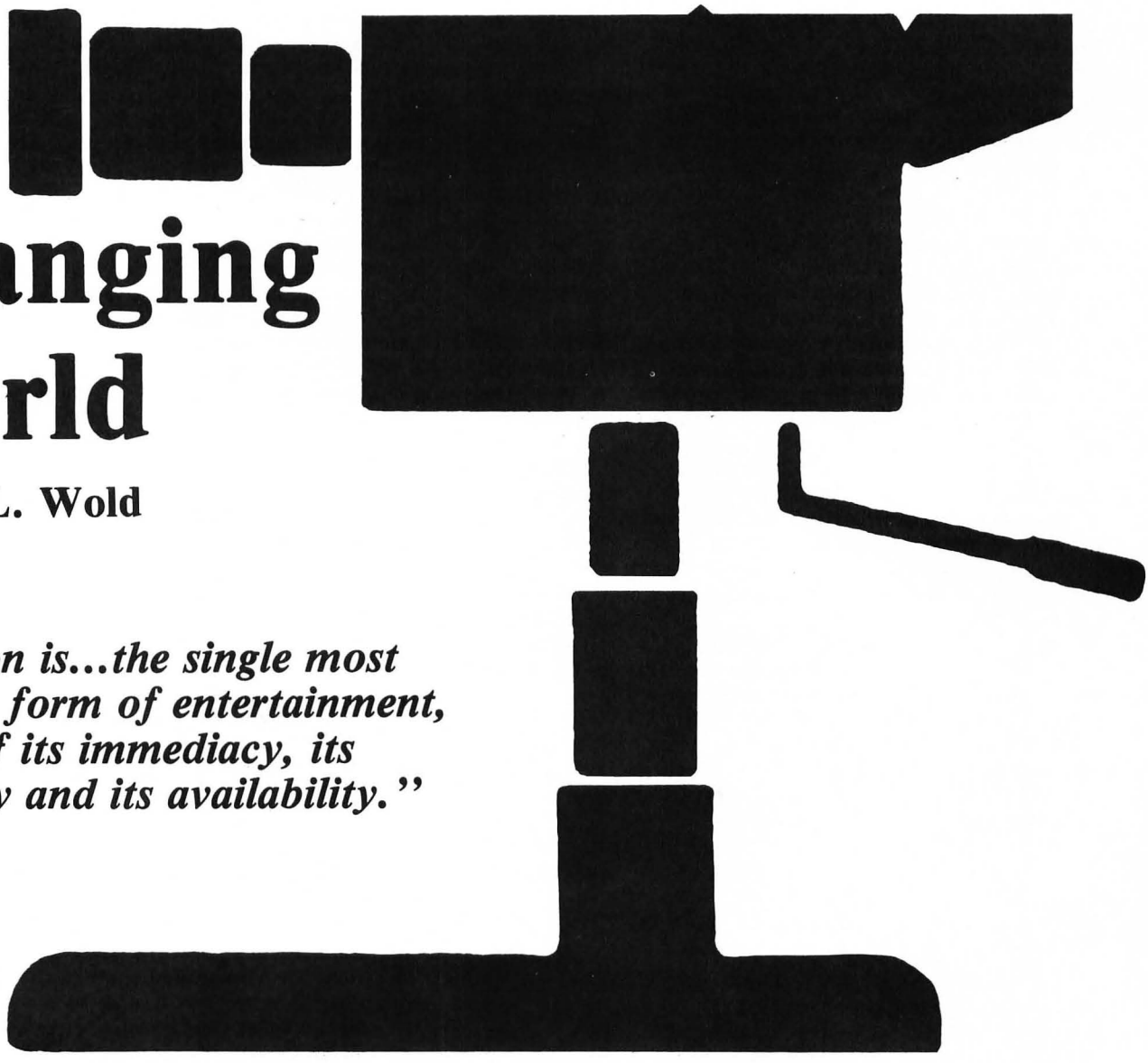
I'd like to offer an enhancement to the Atari version. Since running *SWAT* can be tedious once the beginning lines of the program have been corrected, I added new code to allow the user to designate a starting line, thus bypassing already edited material.

```
32015 POSITION 7 ,3:? "STARTING
LINE # :";:INPUT START
32055 IF L1 < START THEN
A=A+PEEK(A+2):GOTO 32050
```

After typing GOTO 32000, the program will ask for a starting line number. INPUT a number (0 for the beginning of the program), RETURN, then answer *SWAT* regarding output device (there will be a delay while a search is made for the starting line number). Incidentally, I've included no error trapping, so a mistake requires GOing TO Line 32000 again.

Paul Jung
Detroit, MI

Television — A Changing World



by Allen L. Wold

"Television is...the single most important form of entertainment, because of its immediacy, its complexity and its availability."

The form of entertainment with the largest audience today is television. Though sometimes turned on only for "company," or as background noise, most people watch television for several hours every day.

In contrast, a radio plays while we go about our business, and we really don't pay any more than marginal attention to it most of the time. Like pictures on the wall or perfume in the air, it is always there, but peripheral.

We still go to a lot of movies, but not night after night, four or five hours at a stretch. Movies are more engrossing than television, but are taken in smaller, less frequent doses. Theatre, opera, and other forms of public entertainment are much less significant on a day-to-day basis.

Television is, then, the single most important form of entertainment, because of its immediacy, its complexity (music, comedy, drama, news, etc.) and its availability (in many cases, 24 hours a day).

For the purposes of this column, I'd like to divide the television phenomenon into roughly four elements, though there is considerable overlap and interdependence between them.

Television's Four Elements

The first element is the hardware technology of the camera and receiver. Originally, cameras were as bulky as a couple of fruit crates, and were so heavy that they required special dollies to move them. Now, in some cases, cameras are as small as a cigar box. They can be hand-held, with

motion-control devices which produce an image as steady as if the camera were mounted on a rock.

The original TV receivers were as big as a juke box, but had miniscule screens, showing only black and white images at 30 lines per inch. Today, we have TVs with screens of 25" diagonal measurement or larger, with full color pictures and as many as 625 lines per inch. There are also projection screens six or more feet across. At the same time, screens are being reduced once again to under three inches, but in a cabinet you can carry in the palm of your hand.

The second element is the technology of the transmission of the signal. The very first transmission, in 1930, was by radio waves, and radio wave broadcasting of television signals has remained the most widely used form of transmission. Closed circuit TV, which sent the signals by wire, was also widely used, but usually only for short distances and for special purposes, such as machine monitoring, until the development of cable television.

Radio wave broadcasting is very power inefficient. Only a tiny fraction of the wave is actually received, but the wave must be strong enough to provide a good signal to all the sets within a given range. Narrow beam radio requires less power, but controlling the precise direction of the beam is difficult and impractical when transmitting to thousands or millions of receivers. Cable is very power efficient, but requires a physical cable, which is expensive. As fiber optics develop, cable will become ever more practical, as the optical fibers, while requiring boosters and using more energy, carry much more information per area section of cable than do metal conductor cables.

Then, too, there is satellite broadcasting, where a beamed signal to or from the satellite transmits to places otherwise out of range — not only across the globe, but across the solar system as well.

The third element of television is programming, which can be roughly classified in three categories: Informative (*The Six O'Clock News*); Educational (*The French Chef*); Entertainment (*All In The Family*).

There is, of course, no show purely in one category or another. They all overlap, and a few (*Nova*, for example) belong solidly in all three. However, entertainment, without a doubt, constitutes the largest portion of television programming.

I am one of those who feel that programming has, in general, deteriorated over the years. This is due, in part, to a lack of good faith on the part of the producers.

Anything produced in good faith, be it a television show or a sandwich, is produced with the idea that, if it's the best that can be made or done, the public will buy it, and one's financial investment will pay off. Without good faith, the producer's only interest is profit. Good faith assumes, on the part of the producer, an interest in customer satisfaction. Lack of good faith assumes an interest only in the producer's gain.

Programs continue or fail because of ratings. This is supposedly a measure of how popular a show is. In fact, it is a rating of what channel happens to be tuned in at a particular time. If a show is on when the sample is taken, it is assumed that the show is being enjoyed. It may be on because the alternatives are worse. Even if it is the choice of the moment, it may not necessarily be enjoyed by the viewer. Is a show which captures 50 percent of the audience when the alternatives are really poor, any better than a show which captures only 25 percent of the audience when the alternatives are very good?

The statistical methods used to evaluate viewership are less than precise. The margin of error in TV ratings means, for example, that a show rated as number ten could, in fact, be the most popular, or could really rank as number twenty.

Ratings are what advertisers use to judge the effectiveness of their messages. It is assumed that popular shows sell more of the product than unpopular shows. If the message gets to 25 million people, and only one percent buy the product, is that better than if it gets to only one million people, and ninety percent buy the product? The question here is one of feedback.

In TV, there is no direct feedback. The advertiser cannot really know how popular a program is, nor how well it sells his product. Shows are funded according to real or expected ratings. A good show with a low audience might actually be more profitable than a poor show with a high audience which doesn't buy the product.

Up until recently, the three major networks had a virtual monopoly on television programming. They produced shows based on ratings and profits rather than true feedback and a desire to provide the best entertainment possible. They justified their current programming trends with the argument that these shows are what the public wants. If that is true, then the networks are doing their job, and those of us who are dissatisfied will have to shut up and suffer.

The Impact of Cable TV

With the advent of cable television, the public is, in fact, being made aware of alternatives. We cannot teach the public to prefer *Wall Street Week* to *Charlie's Angels*, but we can bring the existence of an alternative to the public's attention.

Part of the reason this will have an effect on programming is because cable makes it possible to get direct feedback. Viewers will vote for a show, not by turning the set on to keep burglars confused, or to provide background noise for their poker game, but because they want to watch that particular show. This is because cable now does, or can, charge specifically for shows watched. The viewer isn't willing to pay cash for a bad show, although he or she might watch it if it were "for free."

Cable networks can record directly who is watching what, and bill the customer accordingly. Hence, producers will be paid not on the whim of an advertiser, but according to public interest and demand. Thus, a TV show need not have good ratings to entice advertisers, but need only be popular enough to pay for itself. The more popular, the larger the budget. Any faltering on the part of the producers will be felt directly and at once, rather than indirectly from faulty statistical methods some time after the event.

Direct instead of indirect competition for dollars will also bring about accelerated improvements in technology. As the cable companies have demonstrated that they can take audiences away from free TV, cable service is expanding. With competing cable services, people will demand more in the way of programming, picture quality, (TV pictures are lower resolution than many computer terminals, for example.) sound, (which is already being improved) color, size, and so on.

But a TV set capable of reproducing 1200 or more lines per inch and high fidelity stereo will be of no use unless the cameras and studios are also improved, and the transmission medium made capable of carrying the more complex signal. Under public pressure, and the inducement of dollars, however, these improvements will surely take place.

Thus, we see that the viewer, the last element in our list, by having a direct influence on programming, the third element, will also have a powerful influence on the other two elements, hardware and transmission.

Video Recording Systems

There are a variety of video recording systems now available. Here is where the first element, technology, and the fourth element, the viewer, link. There is, at present, the question of the legality of recording programs off your television set. I do not wish to discuss that here. I will assume that, after much expenditure of money and hot air, some kind of solution will be reached. What will we have then?

Taking the optimistic approach, I will assume that the advent of nearly universal cable or cable-like TV transmission will have a positive effect not only on the programming, but also on the technology as a whole. Home recording, with proper payment of royalties and so on, will become legal and common. This, in itself, will have an evolutionary effect on the phenomenon of television, but there are several other things to add to the scenario.

The first is that TV cable and phone lines can be shared. Thus, everyone would have videophones, not just audiophones, especially with the development of low-cost, miniature cameras and inexpensive imaging devices.

Secondly, as is already being done in France, a computer terminal can be hooked up to the same lines. In this case, I wish there were a better word for "computer." I'd rather call it a data-device, since not everybody will actually write programs on it, or use it for "computation" in the strict sense of the word.

It becomes apparent now that we're getting into a rather complicated and powerful piece of equipment. We have personal communications, public and subscription entertainment, and access to software packages of all kinds. Our device is not really a television, a telephone, or even a computer, but rather a computerized communications console — what I have called a "comcon" in my novels. For example, when calling a friend on your comcon, you may, simultaneously, listen to music in the background, play chess, and see and hear each other talk. That is, you do not have three or four separate functions to choose from, but a simultaneous combination.

Suddenly, our much maligned TV becomes a thing of wonder. With a proper set-up, there is no need for any white collar worker to go the office. This has been mentioned by other people before, (and perhaps reaches its epitome in *The Naked Sun* by Isaac Asimov) but it will take more than a computer to accomplish it. The burden of installation of cables will be paid for by the profits of the entertainment industry, with the services shared by them and the communications industry. Add the facilities of the computer industry, and it becomes not a dream, but a reality.

MARLIN SAYS:
Is your new Christmas computer awaiting some more refined instructions? Are friends asking "What can it do?". Are the kids itching to get their fingers on the keys? Want to give your toddler an education? To help you out, we've selected some of the top programs in both the entertainment and utility areas to appeal to every member of the family. Choose one program from each category and reduce the already meager tab by an additional 5%. So gather the family around the computer and enjoy!

ENTERTAINMENT/EDUCATION

HODGE PODGE — This is ***THE*** program for young children 2 to 6 (D—$16)
TEACHER'S PET — Educational wonder for your favorite 3 to 7 year old (D—$15, C—$12)
EMPIRE OF THE OVERMIND — Embark upon an heroic quest to a different plane of reality? Game of the year award! (D—$28, C—$23)
LORD OF KARMA — Decipher secrets while exploring a mythical city and countryside (D—$20, C—$16)
SHOOTOUT AT OK GALAXY — Arcade excitement with a touch of strategy! (D—$20[no disk for TRS—80], C—$16)
PREPPIE — The only thing to top this is a duck! (D/C—$23)

UTILITIES (disk only)

TEXT WIZARD — I highly recommend this word processor for your Atari (D—$77)
LETTER PERFECT — Write your grandma! This will help your apple take the byte out of being close. (D—$113)
VISICALC II — ***THE*** electronic spread sheet for the Apple. (D—$188)
TEXT EDITOR — All essential features for serious text editing (D—$32)

Mastercard/Visa (all #'s & ex. date)
CA res add 6%. $2.00 p/h.
Orders over $50 shipped free.

Brown Knows T—Shirts $10 or free with any order over $100!
Send for catalog

BROWN KNOWS COMPUTING

P. O. Box 2875
San Bernardino, CA 92406
Call us at (714) 886-7360

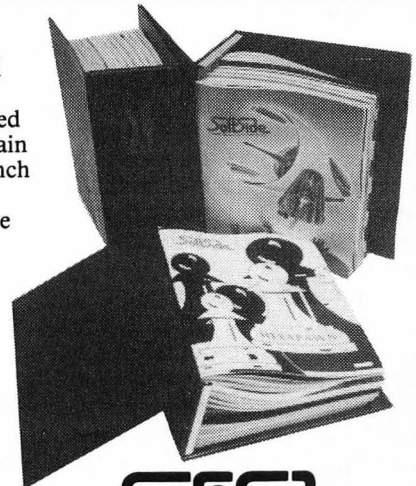
APPLE, ATARI, TRS—80, are registered trademarks of Apple Computer, Atari, and Tandy Corp. respectively

Protect Your Investment

Protect your **SoftSide** back issues (combined editions) with these sturdy binders. Covered with durable wood-grain vinyl, each 8½ x 11 inch binder has an inside pocket and clear sleeve on the spine which you can label for easy identification. Each binder holds 12 issues.

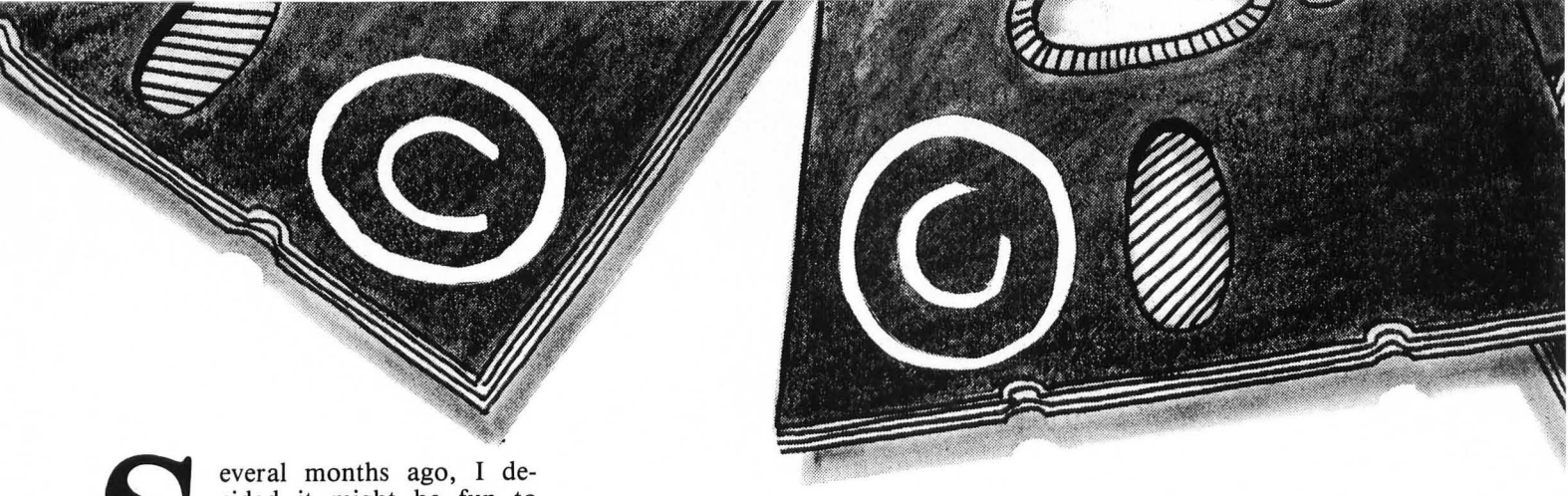
8½ x 11 . . . . . $7.95

Please include $2.50 per order for shipping and handling.



See page 69 for ordering information & back issues bind-in card

SoftSide™
6 South Street, Milford, NH 03055



My Side of the Page

Music Copyright – A Parallel for Software?

by Lance Micklus

Several months ago, I decided it might be fun to learn how to play a musical instrument. The one that caught my eye was the Radio Shack Realistic Moog Synthesizer. In the process of learning how to play it and then finding music to play, I once again came across the rules and regulations regarding copyright.

Because of my computer background, I couldn't help but draw comparisons between situations involving music and those involving computers with regard to copyright. I found the comparisons rather striking.

What Is A Copyright?

People's understanding of the copyright law seems to be based on one or both of two concepts. The first is that a copyright is the right to make a copy of something. You can do whatever you want to with the copyrighted work as long as you don't make a copy of it. The second concept is that of "fair use". If you use the work of another person, then you are obligated to compensate him.

For the sake of discussion, let's talk about just one song which happened to be written by my next-door neighbor, Pat Wells. Pat is the musical director for the folk group at the Catholic Church down the street. Like many churches, her church uses the Lord's Prayer as part of the service. One day, Pat decided to set the words to music. Although the song is actually called *Our Father*, I will refer to it here as *The Lord's Prayer* since that's what it actually is.

The words used in Pat's song were first spoken by Christ 2000 years ago and are in public domain. This means that, of the complete work, half of it is in public domain (the words) and the other half (the music) isn't. Can Pat copyright the song? Before you jump up and say, "Yes", let's think about this for a moment.

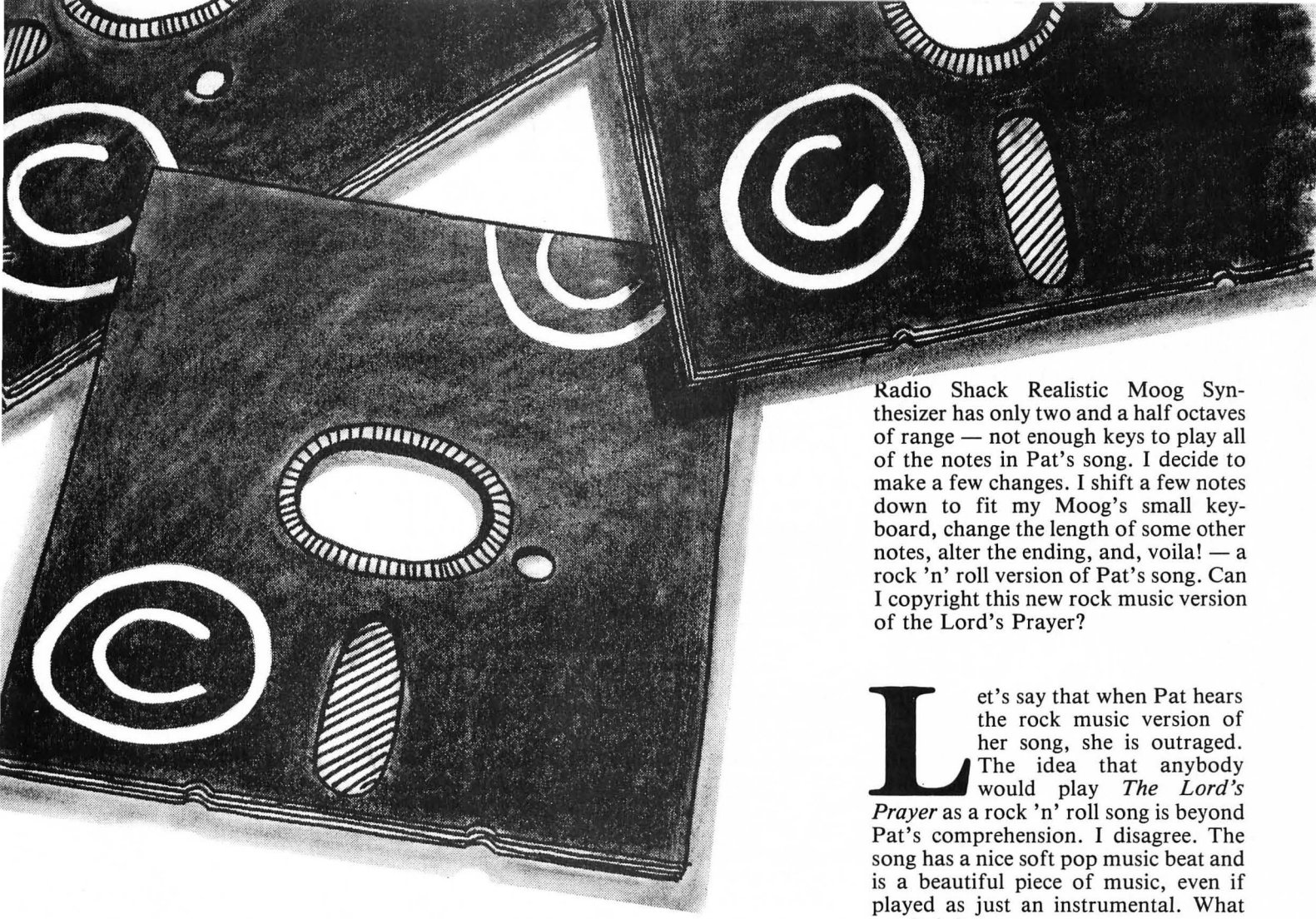
In order to copyright any work, the work must be significant, creative, and original. Let's say that Pat's song was just spoken words with a musical "Amen" added on the end. We would probably say that "Amen" cannot be copyrighted unless it is done in an unusually creative manner. The difference between a spoken Lord's Prayer with a musical "Amen", and just a spoken Lord's Prayer with no music, is insignificant.

The song that Pat actually wrote contains much more than a musical "Amen". The words in the prayer have been set to a melody which contains over one hundred different notes. I will vouch for Pat's honesty and state that her work is original. We'll give her the benefit of the doubt and say that her work is creative. Having met the above requirements, it appears that Pat can copyright her song.

Of course, Pat would like some people to play her song, so she comes over to my house and shows it to me. Let's say that I pay a couple of dollars for one copy. I clamp on my headphones and play the song on my newly acquired Radio Shack Realistic Moog Synthesizer. Does this violate Pat's copyright? I hope not! I am the only one using the copyrighted work and the only one who can hear the music.

The synthesizer has a lot of different controls on it which affect the final sound that comes out of the instrument. To play the song, I need to write down the settings of each control. Rather than trying to cram this information on my one and only copy of the song, I photocopy the music, leaving the back side of each copy blank so I have room for the settings on my Moog. Does this violate Pat's copyright?

If Pat bought one of my computer programs, she could make a work copy so her original could be kept safe and in new condition. The Supreme Court said she could. Is that any different than making a work copy of her song and playing from the work copy instead of the original? I've already indicated that doing so is both a necessity (so there's room for my Moog settings) and a convenience (so the music and the Moog settings are kept together). Let's say that the answer to the



question is *Yes.* I can make work copies of the song for my own use.

Now that I've become proficient at playing this song, my family wants to hear what it sounds like. If I play the song for them, does that violate Pat's copyright? Before you say, "No," let's go one step further. My wife plays the guitar and, after hearing the song on my Moog, decides she'd like to play it on her instrument. Does this violate Pat's copyright?

Or, suppose we decide to play it together. Now there are two people using the same, single, copyrighted musical software. Would it make any difference if we both played from work copies while the original stayed safely away in a drawer?

There are two ways to answer these questions. If your concept of the copyright law is that it defines the right to make copies, we have not violated Pat's rights, providing that when Dianne and I play together, we use only the original sheet music and Dianne looks over my shoulder while I play the Moog. Of course, my son, Tony, could look over the other shoulder to play his saxophone.

The second way to answer this question is from the standpoint of fair use. From this point of view, one must ask who bought the music from Pat — was it Lance Micklus or the Micklus household? If the music was sold to Lance Micklus, then Dianne and Tony Micklus are going to have to buy their own copies. If we say that it was sold to the Micklus household, it is another matter. Since the Micklus household paid for the fair use of the song, we each can play, even if it is necessary to use work copies so I don't have my son's saxophone blasting in my ear.

It comes down to the difference between the two concepts of copyright: The concept of "the right to make a copy" of the original versus the concept of "the right to use" a copy of the original. There can be a world of difference between the two, as the above example shows.

Software Modifications

But I'm not done with poor Pat yet. Another problem has just arisen. Radio Shack Realistic Moog Synthesizer has only two and a half octaves of range — not enough keys to play all of the notes in Pat's song. I decide to make a few changes. I shift a few notes down to fit my Moog's small keyboard, change the length of some other notes, alter the ending, and, voila! — a rock 'n' roll version of Pat's song. Can I copyright this new rock music version of the Lord's Prayer?

Let's say that when Pat hears the rock music version of her song, she is outraged. The idea that anybody would play *The Lord's Prayer* as a rock 'n' roll song is beyond Pat's comprehension. I disagree. The song has a nice soft pop music beat and is a beautiful piece of music, even if played as just an instrumental. What can I do?

First, let's take a look at what we have. We have a song which is now 50 percent in public domain (the words), 25 percent written by Pat Wells, and the remaining 25 percent by Lance Micklus. How much of Pat's song can I use and still copyright the song? The choices are:

A) Lance can't copyright the song if it contains any of Pat's original music.

B) Lance can copyright the song if some amount of it is his own original work. (If you pick this for an answer, you can also figure out how much of it has to be Lance's original work.)

C) Lance can only copyright the changes.

I don't know the answer to this dilemma, but here's what I think: Unless the part of Pat's song I used was insignificant or obvious — such as her "Amen" — I cannot use any of it. Here's my reasoning: if I compile a program using the Microsoft BASIC compiler, I have to pay Microsoft a royalty if I sell the program — even though most of the code is my own and only a small part of the program (the library routines) are Microsoft's. I would assume the same thing would

apply to music. If Pat could prove that I used part of her music in the pop music version of *The Lord's Prayer,* then she would probably have grounds for a law suit. In fact, based on court ruling, she'd have a much better chance of winning her law suit than Microsoft would in a similar situation.

The only thing I can do is copyright the changes. Let's say I didn't like the ending to Pat's version so I wrote my own. I could copyright the ending of the song and I wouldn't need Pat's permission. Not only that, if Pat heard my ending, she couldn't use it without violating my copyright.

One night, Pat and I were discussing some of the things mentioned in this article and Pat brought up a rather interesting point. There's a song she enjoys very much, but she feels it would sound a lot better if it had a flute accompaniment. She asked her guitar instructor about this and he suggested that she contact the composer. The composer may have just the accompaniment she needs or, if she has to write one herself, he may want to make arrangements to use it himself. On the other hand, the composer may not care or may get upset with Pat for "tampering" with his song. Assuming the latter, Pat would then have to weigh the need to have a flute accompaniment — which is her legal right — against the risk of upsetting the original composer.

I thought the advice given to Pat by her music instructor was a very nice courtesy. There have been people who have come up with patches for my programs — which is something like writing a flute harmony for a song. When the patches didn't work, I got tons of calls and letters from people who wanted me to fix the patches and I didn't even know what was going on.

Let's forget about the rock 'n' roll version of the song and try to create some more misery for my best neighbor. It turns out that every Saturday night, Lance and Dianne have a wild party. In the midst of the festivities, Lance and Dianne are making music on the Radio Shack Realistic Moog Synthesizer and the Fender electric fuzz guitar. Our neighbor Pat, being a good Christian, does not participate, and tries to get to bed early so she can lead the folk group at church the next morning.

One day, I go over to Pat's house and inquire about her song. She would love to hear us play her song, but the thought of it being played at our Saturday night party is more than poor Pat can tolerate. Is there any way she can let us have the song, but prevent us from playing it at our Saturday night party?

Copyright vs. Licensing

There is a solution to Pat's dilemma. Instead of selling us a copy of her song, she can license the song to us. Under the terms of the license, Pat can stipulate that the song never be played at places where alchohol is consumed.

If you use a copyrighted work under a license arrangement, the owner of the work maintains control over his copyright. The copyright holder can obtain rights not normally given him under the law, so long as the license does not contain terms which violate the law. If the copyrighted work is sold, the copyright holder loses control over his work. The only control he has is that afforded him under the law.

Until a few years ago, computer programs were never sold. Instead, you purchased a license to use the program, and the author of the program maintained control over its use. Since the copyright law, until recently, gave either little or no protection to computer programs, this was the only way to prevent users from taking unfair advantage of the author.

Many people still don't understand the difference between a software license agreement and a software purchase. If you buy a license to use a computer program, you are bound by the terms of the license. What the copyright law says you can or cannot do is beside the point. It's what the license says that counts.

In signing the license agreement, I am giving up my right to play at least one religious song during my party in exchange for the right to use the music. I cannot get out of that obligation simply by saying I have the right, under the law, to play religious music at my party. I waived that right when I signed on the line.

Let us say, however, that while Lance and Dianne were talking to Pat about the license to use her song, Pat began preaching the Gospel to us and we saw the error of our sinful ways. From now on, it's early to bed on Saturday night so we can rise, well-rested, on Sunday morning to go to our church. Since Pat is no longer concerned about the places we play our music, she sells us a copy of her song.

Public Performance/Use

One day, someone at our church finds out that I am a musician and hears me play Pat's song. I am asked to play the song for the congregation, and do so on the following Sunday. Does this violate Pat's rights? Your choices are:

A) It violates Pat's copyright because it is a public performance.

B) It violates Pat's copyright not because the song was played in public but because it was played for profit.

C) There is only one musician — Lance — using the work so it does not violate Pat's rights.

Some people have told me the answer to this question is "A" because the copyright law does not consider a public performance the same as personal use. For example: A radio station may buy a phonograph record and play it over the air. The phonograph record is one — and only one — copy of a copyrighted work. Yet, under the copyright law, it is illegal to broadcast the record unless rights are obtained specifically for this purpose. In other words, the copyright law is designed to compensate the author of a work not only according to the number of people who actually use the work (the musicians) but also according to the number of people who benefit from the work (the listeners).

Let's say that you set up a public bulletin board system — like FORUM-80 — and have a special section for playing games. One of the games is *The Mean Craps Machine* copyrighted by Lance Micklus. Only users can play the game. They cannot download a copy of it. Now you'd think this was all legal. Yet, isn't that the same as a radio station broadcasting a copyrighted song — an action known to be illegal?

You might argue that Lance Micklus ought to be happy that people get a chance to play his game without being

able to obtain a copy of it. That was the same argument used in the late 1930's and early 1940's when disc jockeys began playing records over the air. Since the tape recorder hadn't been invented, listeners couldn't pirate copies of the songs played on the radio.

You'd have thought the music industry would have been delighted at all of the free promotion they were getting. Instead they were outraged.

The solution was to set up licensing agencies who collected a fee from the broadcasters and divided it up among the composers and publishers. This is why, on the back of all of your record albums, you'll see either the letters BMI (Business Music Inc.) or ASCAP (American Society of Composers, Authors, and Publishers) next to each song title. The letters indicate with which licensing agency the song is registered.

Let's apply that to a discrete computer program. Suppose the Jones Wicket Company buys a *VisiCalc®* program to do sales projections. The results are printed out on the lineprinter, and then taken to a print shop where two hundred copies are made for distribution to the various sales managers. Isn't that the same thing as one person playing a musical instrument and sharing the results — the music that instrument creates — with a lot of other people. If so, I think Visicorp should take the Jones Wicket Company to court on a copyright violation.

Maybe the difference is in the concept of profit, so "B" might be the right answer. Unless the Jones Wicket Company sells the projections, they are not making a profit from the use of the VisiCalc program. Does a church make a profit when it conducts a religious service?

That depends on how you look at it. When people worship, they usually collect *money* in the offering tray. The case could be made that the hymns add to the enjoyment of the church service and therefore help the church to raise money. The case could be made that if there is an offering taken during the church service, then the church service is — among other things — a fundraising activity. Churches are bound by the same rules regarding the use of copyrighted music as everybody else, even though they're non-profit organizations.

I suspect that most of the people reading this article probably picked "C"as the correct answer to the question regarding Lance's solo performance in church. I am the only one playing the music and therefore there is one user per copy.

What if the congregation needs to see the music to be able to sing it? Instead of buying a few hundred copies of the music for the entire congregation, let's use an overhead projector to project the single copy of the song on a screen large enough for everybody to read. That's something like what timesharing computers do. You have one computer and one copy of a program which many people can share.

If you picked "C" as the answer, then you are saying that an author is only entitled to receive compensation for the number of copies in use. If some technical device allows many people to share a single copy without actually duplicating it, it's tough luck for the author.

Is there any final solution to all of this? There must be, but I'm not sure what it is. My purpose here was to explore the copyright issue as it applies to musical and computer software, not to decide it. Hopefully, the up-coming Supreme Court ruling on the copyright law as it affects home video recording may be able to shed some new light on all of this.

Special Thanks

My special thanks to my neighbor, Pat Wells, for allowing me to use her name and her song as examples for this article.

Convert Your TRS-80* into a World Class Computer

THAT REDUCES EYE FATIGUE AND DOESN'T FLICKER

— with LSI's new *Soft-View* ™ Replacement CRT —

The black & white "TV Screen" CRT (picture tube) which came with your TRS-80* is an inexpensive rapid "P4" Phosphor CRT intended for TV use. The display is actually strobing 60 times a second. No amount of "green plastic" will stop this strobing or eliminate the eye fatigue it causes. But a new *Soft-View* CRT display tube with a slower decaying, colored Phosphor will.

- Available in slow-decay green (similar to new IBM* and APPLE III* monitors) or medium decay "European Orange" (easy on the eyes, elegantly beautiful, and the standard for CRT displays in Europe).
- Leaded glass stops X-ray emission.
- Optional Anti-Glare Frosted Glass available to reduce eye strain from glare.
- Easy installation — tubes come with pre-mounted hardware.
- 30-Day Money-Back Guarantee, 1 Year Warranty.
- Ideal for Word-Processing & Programming, fast enough for Games & Graphics.
- Finest quality double-dark glass and phosphor fields make the letters seem to be coming out of black space.

LSI SYSTEMS *Soft-View*™ CRT's:

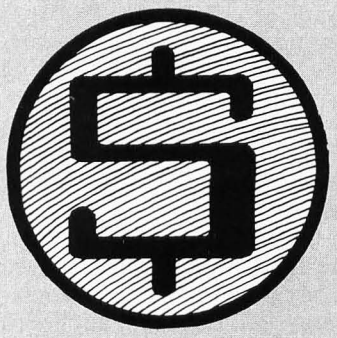
☐ #GN42 Green Phosphor $79.95
☐ #GN42G Green Phosphor with anti-glare $89.95
☐ #OR34 Orange Phosphor $89.95
☐ #OR34G Orange Phosphor with anti-glare $99.95

ADD $7 FOR PACKAGING AND UPS SHIPPING.

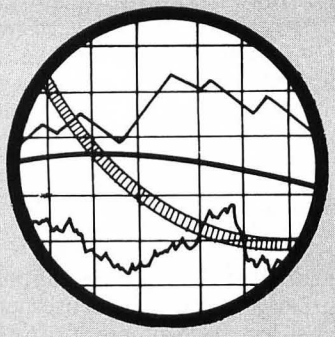
Langley-St.Clair Instrumentation Systems, Inc. To Order Call: **1-800-221-7070** Or ask your Local Dealer.

132 West 24th Street, New York, N.Y. 10011 212-989-6876

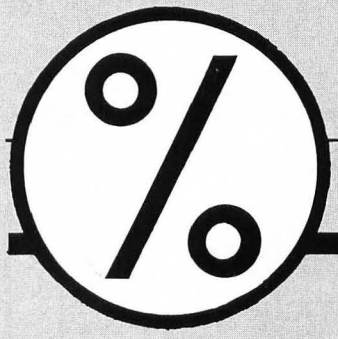
IBM, APPLE* and TRS-80* are trademarks of IBM, APPLE Computer & TANDY Corp.



CALC SIDE



by David Peters



The *VisiCalc*® Spreadsheet Comes Home

"We should all run our households like a company, treating the flow of money in and out as a trackable, analyzable fact of life."

Thanks for the positive reaction to our first *Calc/Side* in Issue 35. Thanks especially to those who bought VisiCalc when they found out it could help in the home environment. This month, we will continue building the simple checkbook management model into the next step — a home general ledger.

The general ledger used to be a big book maintained by a company — often with a quill pen! It recorded the movement of money in and out of the company, and detailed transactions as well. With the general ledger, the company could determine sources of income and profit. They could also identify whom they paid, thus tracking expenses and determining if they were spending their money wisely.

We should all run our households like a company, treating the flow of money in and out as a trackable, analyzable fact of life. Then, perhaps, we could control spending, and end up with a surplus instead of breaking even, or dropping in the hole!

Look at Figure 1. Those who punched along with us last time, and put this model into their Apples®, Ataris®, IBM® PCs, or TRS-80s®, will recognize the outline of the top lefthand part of the model. However, note the changes. A new column has appeared, headed TYPE CODE. The full description of the transactions has gone, and we now note only if it is a DEPosit, a check number, or a machine withdrawal. Otherwise it is the same.

The TYPE CODE is a specific identifying Value designed to cover the way you want to keep track of your money. The single digit values in our model represent income — this householder has two sources of income, (identified as 1 and 2) wants to keep track of an investment account, (numbered 3) and keeps a record for MISC (other casual sources of money).

The expense account is numbered with two digits. We have picked a few representative accounts, but you can have as many as you need, up to the limits of memory. The food account is ll, the car expenses go under 12, pocket money is lumped under 13. (How many of us can keep track of our cash with precision?) You could break this item up into the actual areas in which you spend your pocket cash — lunches, bus fares, taxis, and so on.

Now look at Figure 2 — the actual ledger columns. As you can see, each of the categories of money we are going to track has a column. The formula is the same for each column, personalized with the TYPE CODE number:

IF(C9 = 1,E9,0)

This tells VisiCalc to look at column C and see if the TYPE CODE equals 1, i.e. a paycheck from the first job. If it does, VisiCalc will bring over the

amount from column E, the deposit. If not, it will print zero. The same formula, with only the TYPE CODE changed, is in the other income columns. For those of you who have no logic function (@IF), patience — we have a method for you to achieve this distribution to the accounts, too.

Here's a tip for those whose VisiCalc version has the EDIT feature — replicate the first formula right across the columns, using the (N)o Change indication. It is easier to /Edit the formulae than to type them in. Hit the command slash and E. Then advance with the right arrow to one space past the TYPE CODE figure, delete the resident TYPE CODE and substitute the correct one, hit right arrow to enter it and do the same for the next one.

There is a slight change when you come to the expense columns. Here, you want to bring over the value in the CHECKS column if a match is found. When you get to the first expense column, food in this case, modify the formula to:

IF(C9 = 11,D9,0)

Now the CHECK figure will be brought over if the TYPE CODE is 11.

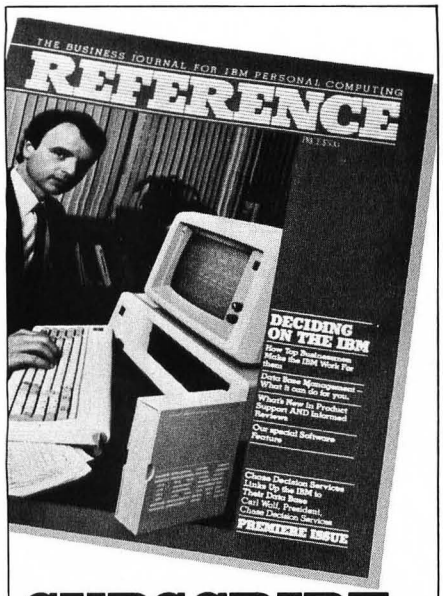
That's all the hard stuff! The two columns shown in our last article (***SoftSide*** 35) that kept track of the transactions that had not appeared yet on the bank statement are off to the right of everything — we just did not illustrate them again. When you are up-dating

Fig 1. The Transaction Entry Area.

	A	B	C	D	E	F
1	CHECK BOOK MAINTENANCE & HOUSEHOLD LEDGER					
2						
3						
4	'1'=OK	TRANS-	TYPE	CHECKS	DEPS	BALANCE
5		ACTION	CODE			
6						
7		OPENING				
8		BALANCE:	4		300.00	
9	1	90	1		650.53	950.53
10	1	92	11	39.23		911.30
11	1	93	12	69.86		841.44
12	1	94	12	70.00		771.44
13	1	DEP	1		625.00	1396.44
14	1	95	13	50.00		1346.44
15	1	DEP	2		267.16	1613.60
16	1	DEP	4		44.10	1657.70
17	1	MACHINE	13	40.00		1617.70
18	1	96	16	50.00		1567.70
19		97	12	26.95		1540.75
20		98	14	250.88		1289.87
21	1	DEP	2		450.00	1739.87
22		DEP	3		26.95	1766.82
23		99	15	29.95		1736.87
24		100	17	69.80		1667.07
25						1667.07
26						1667.07
27						1667.07
28						1667.07
29						
30			TOTALS	696.67	2363.74	1667.07
31						
32		ENTER STATEMENT BALANCE>			2017.70	
33						
34		ENTER CHECKBOOK BALANCE>			1667.07	
35						
36		DEPOSITS NOT RECORDED—>			26.95	
37						
38		CHECKS OUTSTANDING———>			377.58	
39						
40		TRUE BALANCE			1667.07	
41						
42		RECONCILIATION			0.00	
43						
44		YOUR ACCOUNT OKAY IF ZERO				

Fig 2. The Ledger Accounts.

	G	H	I	J	K	L	M	N	O	P	Q
1											
2	———	——INCOME TYPES———			————————E X P E N S E A C C O U N T S ————————						
3	PAY	PAY	INVEST	MISC	FOOD	AUTO	CASH	ENT/	HOUSE	TELE-	UTIL-
4			MENTS					VACATION	REPS	PHONE	ITIES
5	1	2	3	4	11	12	13	14	15	16	17
6											
7											
8	0.00	0.00	0.00	300.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
9	650.53	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	0.00	0.00	0.00	0.00	39.23	0.00	0.00	0.00	0.00	0.00	0.00
11	0.00	0.00	0.00	0.00	0.00	69.86	0.00	0.00	0.00	0.00	0.00
12	0.00	0.00	0.00	0.00	0.00	70.00	0.00	0.00	0.00	0.00	0.00
13	625.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
14	0.00	0.00	0.00	0.00	0.00	0.00	50.00	0.00	0.00	0.00	0.00
15	0.00	267.16	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
16	0.00	0.00	0.00	44.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00
17	0.00	0.00	0.00	0.00	0.00	0.00	40.00	0.00	0.00	0.00	0.00
18	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	50.00	0.00
19	0.00	0.00	0.00	0.00	0.00	26.95	0.00	0.00	0.00	0.00	0.00
20	0.00	0.00	0.00	0.00	0.00	0.00	0.00	250.88	0.00	0.00	0.00
21	0.00	450.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
22	0.00	0.00	26.95	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
23	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	29.95	0.00	0.00
24	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	69.80
25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
26	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
27	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
28	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
29											
30	1275.53	717.16	26.95	344.10	39.23	166.81	90.00	250.88	29.95	50.00	69.80
31											



SUBSCRIBE TO SUCCESS

REFERENCE, the only business journal for the IBM Personal Computer.

Are you in business? Read **REFERENCE** to find out how you can utilize the IBM Personal Computer in your business.

Yes, I want to subscribe to **REFERENCE** Magazine for 1 year (6 issues) for only $18.00!
PLEASE PRINT

NAME

COMPANY

ADDRESS

CITY STATE ZIP

☐ Personal Check Enclosed
☐ Money Order Enclosed
☐ MasterCard
☐ Visa

Card #

Exp. Interbank #

Signature

REFERENCE™

P.O. Box 1200, Dept. M
Amherst, NH 03031
(603) 673-9544

the model, use the /MOVE function to stash them temporarily — perhaps in the double letter column area. When you have created all the expense columns for your particular household needs, you can /MOVE them back again, and VisiCalc will keep the coordinate references straight as it moves the columns around. At the bottom of all the ledger columns, the @SUMS give you a running record of your income, its sources, and your expenses.

"...the @SUMS give you a running record of your income, its sources, and your expenses."

Manual @IF

If you have no logic function in your version of VisiCalc — the @IF we use to carry out the transactions, you have to carry it out manually. Here is a way to do this that reduces the effort, the chance of error, and checks your accuracy. This method is one of hundreds of useful ideas sent in by members of InterCalc, our spreadsheet users group.

First, enter the correct amount in the Check or Deposit column. Then instead of retyping it, risking an error, /Replicate it to the correct column. With the cursor on the amount, hit Command slash, R, Return or Enter, then move the cursor to the right column and hit Enter again. Now you have reduced the chance of a mistake.

However, to check your accuracy, we have an error checking column — see Figure 3. The formula in this column is

(Col D + Col E)-(@SUM(Col G ... Col Q)

Since you never make entries in both columns D and E, checks and deposits, you are subtracting the sum of the ledger columns from one or the other of the columns. The answer should be zero if a correct distribution was made. If it is not, then the line on which you made the mistake will be identified by a value appearing on it. (We have inserted one in the illustration.) Here's a tip for you — put the bottom of this column in a single column window on the right of your screen during data entry. In this way you can check periodically to see that everything is okay.

That's it for this time. Feel free to write with questions or comments. (That's how I am going to know that I am writing things you want to read.) The address is InterCalc, PO Box 254, Scarsdale, New York 10583. Enclose a stamped, self-addressed envelope if you want a direct reply.

Fig 3. The Error Check Column (Used in manual distribution)

```
          R
TRANS--
  ACTION
   CHECK
---------

        0.00
        0.00
        0.00
        0.00
        0.00
        0.00
        0.00
        0.00
        0.00
        0.00
        0.00
        0.00
        6.00
        0.00
        0.00
        0.00
        0.00
        0.00
        0.00
        0.00
        0.00
---------
        6.00
IF NOT 0
ERROR IN
DISTRIB--
   UTION
```

If You Sell Small Business or Personal Computer Systems, Word Processing, Software, Media & Supplies, or Computer Services . . .

MEET YOUR MARKET, HEAD-ON.

COMPUTER SHOWCASE EXPO is the best place to meet the vast and growing computer market. Face-to-face.

This proven series of regional, end-user expositions delivers thousands of prospects, with needs in mind and buying dollars in hand.

THE INTERFACE GROUP, the world's largest producer of computer shows, offers you instant availability to a wide variety of end-users such as:

- ***Small Businessmen • Doctors • Lawyers***
- ***Accountants • Retailers • Corporate Managers***
- ***Many Other Professionals***

These face-to-face contacts are in business such as:

- ***Banking • Health Care • Insurance • Education***
- ***Government • Real Estate • Manufacturing***
- ***Distribution***

If you sell computers, or computer systems, as a solution for a variety of Small Business problems, COMPUTER SHOWCASE EXPO offers you more prospects in a few days than you could possibly see in months!

For more information on how you can reach the market you need, call THE INTERFACE GROUP, toll free at 800-225-4620.
(In Massachusetts, call 617-879-4502)

SPRING 1983

PHOENIX
Feb. 3-6, 1983

ATLANTA
March 24-27, 1983

CHICAGO
March 24-27, 1983

ST. LOUIS
April 21-24, 1983

SOUTH FLORIDA
April 28-May 1, 1983

BOSTON
April 28-May 1, 1983

ANAHEIM
May 5-8, 1983

WASHINGTON, DC
May 5-8, 1983

SAN DIEGO
May 12-15, 1983

HOUSTON
May 19-22, 1983

SEATTLE
June 2-5, 1983

FALL 1983

DETROIT
Sept 22-25, 1983

NEW YORK
Sept. 22-25, 1983

SAN FRANCISCO
Sept. 29-Oct. 2, 1983

ATLANTA
Sept. 29-Oct. 2, 1983

PHILADEPHIA
Oct. 6-9, 1983

PITTSBURGH
Oct. 20-23, 1983

SOUTH FLORIDA
Oct. 27-30, 1983

DENVER
Nov. 3-6, 1983

LOS ANGELES
Nov. 10-13, 1983

WASHINGTON, DC
Nov. 10-13, 1983

CHICAGO
Nov. 17-20, 1983


THE INTERFACE GROUP

The World's Largest Producer of Computer Conferences and Expositions

Producers of: THE COMPUTER SHOWCASE EXPOs (Nationwide), INTERFACE, FEDERAL DP EXPO, COMDEX/SPRING, COMDEX/FALL, COMDEX/EUROPE

160 Speen St., P.O. Box 927, Framingham, MA 01701 • (617) 879-4502, Toll Free (800) 225-4620, TELEX - 951176, TWX - 710-380-7645
Regional Office: Suite 121, 4700 North State Road 7, Fort Lauderdale, FL 33319 • (305) 484-6800, Toll Free (800) 327-0100

Deluxe Personal Finance is a personal accounting program for an Apple® with 32K RAM, an Atari® with 24K RAM (DOS 2), 32K RAM (DOS 1), a TRS-80 with 32K RAM, or an IBM® PC with 32K RAM. All four versions require a disk drive. The Checking Account module is presented here. The Budget module will appear in Issue 38.

Del
Per
Fina

TRS-80® version by Lance Micklus. Translations

The following is the checking account module of the *Deluxe Personal Finance System*. The checking account program presumes the budget program when it comes to using certain menu options. We will point these out as they occur in our testing and learning examples. Initially, you will be able only to enter checking deposit data. You will be unable to exercise any of the options requiring the budget module of the DPF System.

Because of the size and complexity of the checking account module, you will probably need about a month to enter the program and get up to speed using it. By that time you will have the budget module, which will appear in the next issue. Then you can proceed with the system.

Note: You will see references to the "ENTER" key. This key will be labelled "RETURN" on some systems.

The Initializer Program

Both the checking and budget programs expect certain data files. Before you run the checking program, run the initializer to create these files. When you run the initializer, it will create all the empty data files that the system will *need*.

uxe
rsonal
ance

and modifications by the *SoftSide* programming staff.

This includes the budget data file. The initializer program is called PFINIT/BAS or PFINIT.BAS.

Disk File Allocation

The checking and budget programs access the following data files:

1. CHECKING/DAT — Outstanding checks data file
2. BUDGET/DAT — Budget data
3. CANCELCK/DAT — Cancelled checks data file used to transfer checks from 1 to 5.
4. CKFILE/DAT — Data file containing this year's cancelled checks.
5. FINANCE/SC0 — Time and date and backup file for current data entry.

Note: The slash (/) is replaced by a period (.) in the IBM and Atari versions.

The program assumes a single disk drive and a single diskette.

File Verification

The TRS-80 version verifies the results of disk operations by saving the data file twice using different names, then reading the files back and comparing them. If the files match, the program proceeds. If not, you are warned of the problem. This routine was not implemented on the Apple or Atari versions, so the file FINANCE/SC0 is not used in these versions.

We assume that your checkbook is reasonably up to date and in balance. After all, no computer account system can organize a disorganized paper system. The best way to learn this system is to take your most recent bank statement, enter the data into the checking module and compare the results with those of your paper system. This will help you debug the program and learn how it operates. The program's methods may well differ from yours. Use the manual system as a comparison to see how the computer gets the same results. Experiment with familiar data to test and learn the system.

Budget Items in Sample Data

Budget names are contained in DATA statements beginning at line 10000. The default names are listed in Table 1.

Table 1.

0 Dentist/Doctor	11 Educat'l Exp	22 Gas/Oil
1 Medical Aids	12 Union Dues, Etc.	23 Vac/Entertn'mt
2 Pharmacy	13 Child Care	24 Clothes
3 Medical Ins	14 Rent/Mtg Princ	25 Furnishings
4 Med Mileage	15 Utilities	26 Household/Items
5 Interest/Mtg	16 Insurance	27 Misc Exp.
6 Interest/Chges	17 Groceries	28 Open Item
7 Fixed Taxes	18 Loan Principal	29 Salary
8 Other Taxes	19 Home Repairs	30 Misc Deposits
9 Donations	20 Savings	31 Interest
10 Loss	21 Auto Repairs	32 Checking/Cash

Note that we have designated the codes much as the IRS distinguishes tax deductible (0 - 13) from non-deductible (14 - 27) items:

- Interest separated from principal payment
- Major loans separated from charges
- Charges are subdivided according to type of purchase:

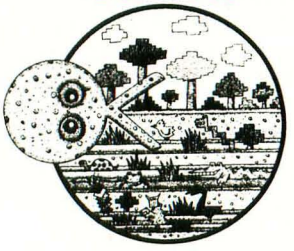
a. gas/oil
b. vacation/entertainment
c. clothes
d. furnishings
e. household
f. miscellaneous

- Home and Auto repairs are kept separate
- Utilities are combined
- Insurance is combined except for Medical Insurance which is deductible.

NEW ADVENTURE
ARCADE
CHALLENGERS

Here are the latest, most exciting arcade and adventure games PDI has ever offered ATARI® computer owners!

SWAMP CHOMP

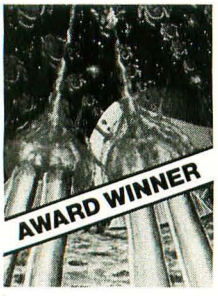


NEW

Life in the Muckedoo Swamp is tough. Alligators, snapping turtles, vampire bats and even ghosts—all try to eat you, a hungry defenseless Gorx. If only you can make it to the feeder station and metamorphose, you'll show them what a swamp chomper can do! One or two players. 24K Disk & Joystick/16K Cassette & Joystick.

MOONBASE IO

Most Innovative Game of 1982
(Electronic Games Magazine)



Moonbase Io is a winner every way. It's a voice-activated arcade game with three very different adventure settings. 1) Navigate the alien mine field. 2) Defend Moonbase Io. 3) Attack & destroy mother ship. If you win, you get a personal Presidential commendation from Earth! Seven levels of difficulty. Sensational graphics. 24K Disk, Cassette & Joystick/16K Cassette & Joystick.

Clipper

(Around The Horn in 1850)

NEW

You're the captain of a clipper ship bound from New York to San Francisco, with lots of decisions to make. You pick vessel, cargo, crew and course. Then use your skills to overcome storms, icebergs, illness, delays, doldrums, mutiny and more! Voice-narrated, this high adventure challenges your brain and navigation skills. 32K Disk, Cassette & Joystick/24K Cassette & Joystick.

Available at leading stores or direct from PDI.
ATARI® is a trademark of Atari, Inc.

PDI

Program Design, Inc. 11 Idar Court, Greenwich, CT 06830

This type of budget item assignment may help when income tax time rolls around. Item 28 has no assignment for this run. Items 29 to 31 are designated as income.

The Magic Of Account #32

You must never change the name of Item #32 because this item allows you to write a check to cash, and later, if you choose, designate with the budget program how that cash was spent. A double entry does not result because the budget program ignores all transactions charged to budget #32. This unique item will be discussed in more depth later.

Be sure to bear these points in mind when you edit the DATA statements to change the budget names.

Running The Checking Account Program

Main options allow you to add, correct, or cancel individual checks, view or list all outstanding checks, justify your monthly statement, or estimate your total bills. On this first run, the sub-menu options 1, 3, 4, and 7 will not appear because you have no outstanding transactions. The options are:

0 TO END SESSION
1 TO LIST OUTSTANDING CHECK FILE
2 TO ADD NEW CHECK TO FILE
3 TO FIX NEW CHECKS WITH DATA ERRORS
4 TO CANCEL CHECKS RECEIVED FROM THE BANK
5 TO JUSTIFY THE ACCOUNT WITH BANK STATEMENT
6 TO ESTIMATE TOTAL BILLS DUE
7 TO PRINT OUTSTANDING CHECK FILE
8 FOR OUTSTANDING CHECK STATUS
9 TO RESET SCREEN (TRS-80 and IBM only)

Type 0 to exit the checking account program. Your data will be saved, all files will be closed and you will return to BASIC. Data is not saved to disk if you do not exit the checking program through 0.

We will view the other options by proceeding to enter data in normal sequence.

Type 9 to reset the screen (TRS-80 and IBM).
Now type 2 to add new check to file.

Information entered or changed in this section will be saved in the CHECKING/DAT file. It will remain there until you cancel checks.

a. When prompted, enter a TRANSACTION NUMBER, an AMOUNT, and specify CHECK OR DEPOSIT.

b. When you complete the deposit transaction above, the screen will display the list of budget items in Table 1 and prompt you for a budget number.

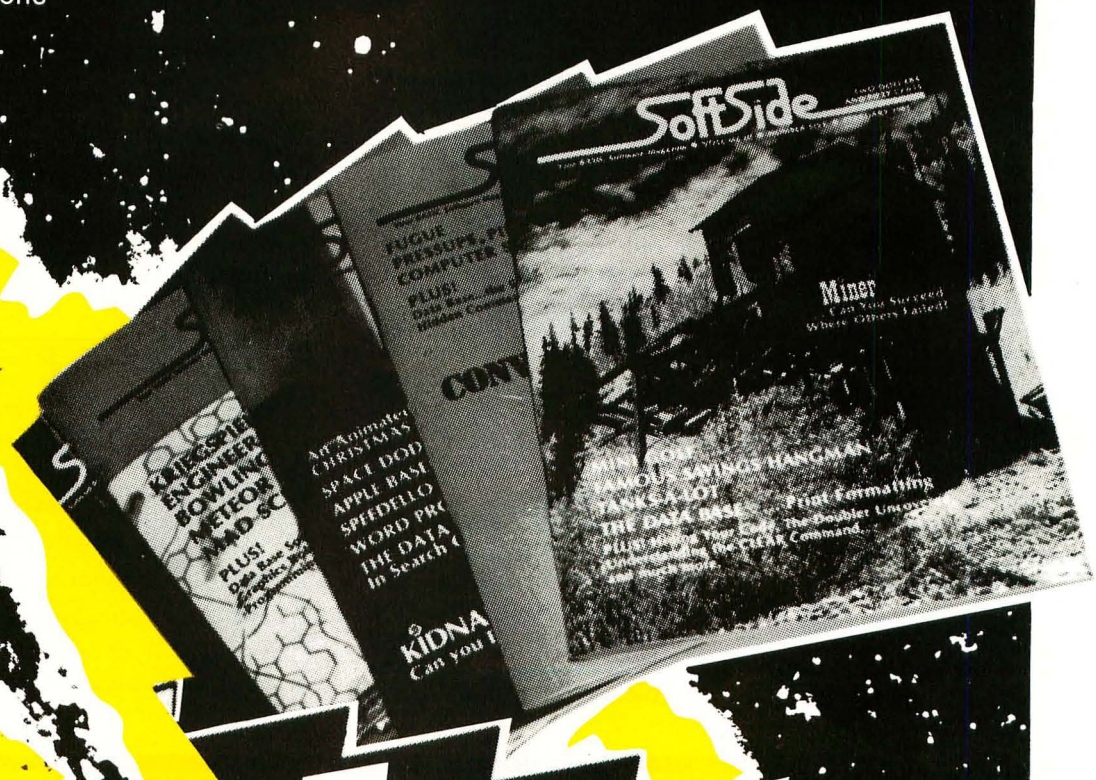
Select the income budget item #29 for salary and view your deposit slip. At this point you may either enter data from your personal checkbook or you may walk through the program using the samples we provide.

The screen display will show your data in the form of a deposit slip, including a recalculated balance.

continued on page 26

SoftSide DV, the magazine of the future, is here!

If your computer could pick a magazine, wouldn't it prefer one in its own language? Now there's one available. **SoftSide DV** is an enhancement of the **SoftSide** you have in your hands.



SoftSide DV contains not only the complete programs listed in every month's issue of **SoftSide,** but additional programs of every conceivable type, as well — multiple and Machine Language programs, modified languages, ongoing modular programs and software so extensive, it would take an entire issue of **SoftSide** just to print the code. Only the documentation for these programs will appear in **SoftSide** Magazine, **NOT** the code.

Feel as though you're missing something? You are! But, you needn't miss out on another issue. **SoftSide DV** is now available for Apple™, ATARI® and the TRS-80®. The cost to you — $125 for 12 magazines and 12 disks, packed with some of the best software available, all delivered to your home in the next year. For orders outside the USA, please add $36. For your convenience, we offer an installment payment plan for VISA and MasterCard holders: You pay only $32.50 per month for four months (a total of $130, which includes a $5 billing charge). Please use the special DV-CV bind-in card in this issue to order.

Computerists are offered the rare opportunity of marching into a new frontier. Advance to the front of the parade by subscribing to **SoftSide DV**, the magazine of the future, available today!

You are now back to Step a. You may enter another transaction or type 'Q' to exit. Transaction numbers that are outstanding (not cancelled) cannot be re-used. Try to enter a new transaction number 1.

The deposit slip appears on the screen with the message, ALREADY EXISTS.

Transaction numbers must be in the range of 1 to 99999. You might choose 1 - 99 for deposit transaction numbers and 100 - 9999 for checks. You may enter each transaction number with one trailing decimal place (i.e., 1.0 to 1.9). This feature allows you to have 10 sub-categories per transaction. Let us try a few uses for this option:

Enter an expense transaction for your mortgage. Since the interest portion is deductible but the principal is non-deductible, you might like to separate these for tax purposes.

Enter 100.0 FOR $19.67 AS A CHECK APPLIED TO ITEM #5

100.1 FOR $108.39 AS A CHECK APPLIED TO ITEM #14

You have written one check #100 to the bank for $128.06, but the system has applied it to two separate budget items. Any use of decimal places should be noted in your checkbook so that you may recall the transaction when necessary. Another use for this decimal feature might be to flag transactions. For example, all utilities have been lumped under one budget item but may include electricity, telephone, water, heat, etc. By assigning each a decimal value of its own, you would instantly be able to view any utility transaction and know to which utility you made the check payable.

Enter the following utility payments:

Table 2.

Transaction #	Paid To	Amount	Type	Applied To Item #
101.0	.0 = elect.	27.33	C	15
102.1	.1 = water	5.08	C	15
103.2	.2 = heat	80.00	C	15
104.3	.3 = tele.	10.49	C	15

Continue to write checks, noting decimal use for insurance:

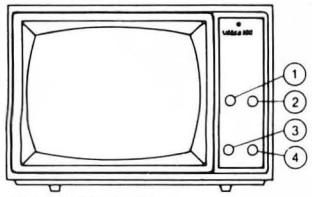
Table 3.

105.0	groceries	50.00	C	17
106.0	dentist	20.00	C	0
107.0	.0 life ins.	9.01	C	16
108.1	.1 car ins.	20.00	C	16
109.2	.2 home ins.	20.00	C	16
110.0	major loan	33.46	C	18
110.1	int./loan	42.78	C	6
111.0	to cash	25.00	C	32

continued on page 29

DO YOU KNOW WHERE YOUR COMPUTER IS TONITE ?

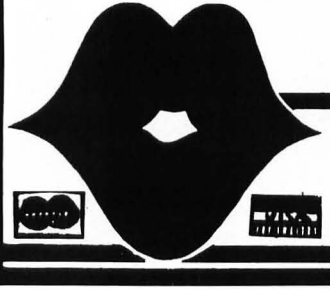
by AMDEK



VIDEO 100
1. CONTRAST
2. POWER-BRIGHT
3. V-HOLD
4. H-HOLD

12" BLACK¢WHITE MONITOR
FULL FACTORY WARRANTY
$79.95

for APPLE
16K RAM CARD
*LANGUAGE TRANSPARENT
$69.95
COEX FACTORY WARRANTY



floppy DISKETTES
5¼" - 100 PER BOX
ALL CERTIFIED — 100% GUARANTY
$149.00
ABOVE WITH HUB RINGS
$169.00

FLOPPY DISK DRIVE
DESIGNED FOR YOUR APPLE®
$287.95
TRACK ZERO MICRO SWITCH
DOS 3.2.1¢ DOS 3.3
PASCAL¢ CP/M
Fourth Dimension Systems
CONTROLLER CARD FOR ABOVE $99.00

EXTENDER CARDS
FOR APPLE. . . . $16.95
FOR I.B.M. $19.95

VISION-80© 80x24 Video Display Card $249.00
Vista Computer Company's new Vision-80 board is a sophisticated yet easy to use video display card for the Apple™ computer.

PROTOTYPING CARDS
Apple $19.95
IBM $49.95

NEW FROM COEX
EPSON TO APPLE PARALLEL INTERFACE!
$49.95
CABLE INCLUDED

Components Express, Inc
1380 E. Edinger, Santa Ana, CA 92705 (714) 558-3972

Why you should buy

ChequeMate™ PLUS

vs.

The Home Accountant™

Feature	ChequeMate™ PLUS	The Home Accountant™
Accounts per disk	20	5
Budget catergories	100	100
Accounts payable capability	**YES**	**NO**
On-Screen lookup of categories, etc.	YES	NO
Prints reports to screen or printer	**YES***	**NO**
Print checks	YES	YES
User defined report limits and sorts	**YES**	**NO**
Prints personal financial statements	YES	YES
All screens 80 column format	YES*	NO
Uses function keys	YES*	NO
Graphic budget analysis with trends	YES	YES

*IBM P.C. version

When you compare, on Apple or IBM, your choice becomes clear. ChequeMate™ PLUS standard features leave all the others behind. All reports allow you to limit and sort by any data field. This means ChequeMate™ PLUS offers you over 10,000 reports. Accounts Payable capabilities are also standard with ChequeMate™ PLUS. With the IBM P.C. version you can even view your reports on the screen.

ChequeMate™ PLUS features only tell part of the story. When you see it in operation you will really find out how user friendly a computer can be.

All in all once you make the comparison, your decision of which package to buy becomes easy. See ChequeMate™ PLUS at your local computer or software store. When you compare, there really is no competition!

Masterworks Software, Inc./25834 Narbonne Ave., Lomita, CA 90717/(213) 539-7486

Home Accountant is a registered TM of Continental Software. IBM P.C. is a registered trademark of International Business Machines, Inc.

KELLY'S COMPUTING

IBM and ATARI COMPUTER PRODUCTS

Post Christmas INVENTORY REDUCTION SALE!

ATARI 800 Computer 48K	$499.00
ATARI 810 Disk Drive	$440.00
WICO JOY STICK	$ 19.99

THRESHOLD
PROTECTOR II
NAUTILUS
SHAMUS
SLIME
DODGE RACER
CHICKEN
CLAIM JUMPER
FROGGER
—YOUR CHOICE—

29.95 EA.

IBM - E.A.S.Y. (Executive Accounting System)	$444.00
OKIDATA 82A Printer	$448.00
TRACTOR FEED FOR 82A Printer	$ 49.00
LETTER QUALITY BROTHER Daisy Wheel Printer (Parallel)	$799.00
LETTER QUALITY SERIAL Daisy Wheel Printer	$866.00

-Many other IBM & Atari Programs Available-

SEND FOR A FREE CATALOG TODAY

— OR CALL —

(800) 572-9215

Calif. Res. **(714) 369-8113**

SEND FOR FREE PRICE LIST

ALL ORDERS SHIPPED FROM STOCK WITHIN 48 HOURS. WE PAY SURFACE FREIGHT COSTS ON **SOFTWARR** ORDERS OVER $75 TO ANY **SINGLE** CONTINENTAL U.S. DESTINATION. CALIF. RESIDENTS ADD 6% SALES TAX. C.O.D. ORDERS ADD $5.00. FOREIGN & AIRMAIL CHARGES EXTRA. **MASTERCARD** AND **VISA** WELCOME. PRICES SUBJECT TO CHANGE WITHOUT NOTICE.

NAME _____

STREET _____

CITY _____ STATE _____ ZIP _____

CARD # _____ EXP. DATE _____

SIGNATURE _____

3515 BRYCE WAY, RIVERSIDE, CA 92506

When you have completed your transaction entry, enter transaction QUIT to return to checking menu.

Type 1 to list the outstanding check file.

The screen will display the following in sequence.

MM/DD/YY OUTSTANDING CHECK FILE

Table 4.

Transaction #	Amount	Itemized As	Type
1.0	$500.00	Salary	Deposit
100.0	$19.67	Interest/Mtg	Check
100.1	$108.39	Rent/Mtg Princ	Check
101.0	$27.33	Utilities	Check
102.1	$5.08	Utilities	Check
103.2	$80.00	Utilities	Check
104.3	$10.49	Utilities	Check
105.0	$50.00	Groceries	Check
106.0	$20.00	Dentist/Doctor	Check
107.0	$9.01	Insurance	Check
108.1	$20.00	Insurance	Check

Type 1 To Continue, ELSE 2

Type 1 to view the remaining outstanding checks.

Table 5.

Transaction #	Amount	Itemized As	Type
109.2	$20.00	Insurance	Check
110.0	$33.46	Loan Principal	Check
110.1	$42.78	Interest/Other	Check
111.0	$25.00	Checking/Cash	Check

END OF LIST (ENTER)#

Press ENTER.

These are the total outstanding transactions being carried in the system. You may check them against your checkbook. Suppose you find an error. Check 109.2 in your checkbook is for $25.00, not $20.00 as entered.

Type 3 to fix checks with data errors.

Enter the transaction # as 109.2. The data will be displayed on the screen in the form of a check, along with the following list of options:

0 - DONE
1 - FIX #
2 - FIX AMOUNT
3 - FIX ITEM NAME
4 - VOID

Select 2, enter the corrected amount of $25.00, and view the new check. Note corrected balance.

Enter 0 for no more changes and Q for no more corrections. You will be returned to the Checking Account Menu. Note: If you ever enter a deposit as a check in error, or vice versa, make the amount a negative value using option #2, and it will automatically switch its type of transaction.

Type 7 to print the Outstanding Check File.

The screen displays what is being printed and you will need to type 1 to print the second screen. If you have been typing in all the sample data, the outstanding check list will look like Table 6. If you have been using your own data, the report generated will reflect that. The format will be the same.

Table 6.

Transaction #	Amount	Itemized As	Type
1.0	$500.00	Salary	Deposit
100.0	$19.67	Interest/Mtg	Check
100.1	$108.39	Rent/Mtg Princ	Check
101.0	$27.33	Utilities	Check
102.1	$5.08	Utilities	Check
103.2	$80.00	Utilities	Check
104.3	$10.49	Utilities	Check
105.0	$50.00	Groceries	Check
106.0	$20.00	Dentist/Doctor	Check
107.0	$9.01	Insurance	Check
108.1	$20.00	Insurance	Check
109.2	$25.00	Insurance	Check
110.0	$33.46	Loan Principal	Check
110.1	$42.78	Interest/Other	Check
111.0	$25.00	Checking/Cash	Check

When the listing is complete, you see this prompt:

END OF LIST (ENTER)

Press ENTER and return to checking menu.

Type 8 for Outstanding Check Status.

This selection allows you to view the results of your data entry thus far, depending on whether you used your own or sample data:

CURRENT BALANCE IS $23.79

O. C. FILE	UPDATED
# ON FILE	15
OUTSTANDING	$23.79

HIT ENTER TO CONTINUE.

Hit ENTER to return to Checking Menu. Type 0 to End Session.

Transactions you have entered will be read and verified and you will exit to BASIC. For this exercise, let us suppose two weeks have passed and you have been paid again. Rerun the checking account program.

Type 2 and input a deposit transaction 2 for $500 and apply it to budget item #29, salary. Exit to checking account menu.

Type 6 to Estimate Total Bills Due.

This feature allows you to estimate payments before you actually write the checks. You can run this section as often as necessary to come up with the right combination.

When prompted, enter an amount you would like to pay. That amount will be deducted from your remaining balance. If you find yourself in the red, go back and revise your estimates until you are satisfied. Then enter Q and return to checking account program. Select Option #2 and enter your checks for real.

Until now you have been dealing with checks you have already written. Now you will deal with checks you are about to write. You have made a dry run to sort expenses before doing so.

Type 4 to Cancel Checks Received from Bank.

During the trial runs it is fine to cancel checks without running the budget module to carry the data forward for future reports; however, if you plan to run budget reports in the future, you should not cancel checks and reconcile a statement permanently until you have the budget module.

A point should be made about this program option. The DPF programs treat a check as being written for the month it clears the bank. In other words, if you wrote a check in March, but it wasn't cashed until June, then the money was actually spent in June, not March. Once a check is cancelled, this new data is stored in a special CANCELCK/DAT file and is deleted from the CHECKING/DAT file, which carried outstanding check information. A newly updated file CKFILE/DAT is created. It contains the merged data of all returned checks written this year.

You will now see the prompt:

TO WHICH MONTH SHOULD THESE CANCELLED CHECKS BE CHARGED:
(1 - 12)?

Enter 11 for this run. Enter the transactions (deposits and checks) returned from the bank with this month's checking account statement one at a time.

As you cancel each transaction, it will be displayed on the screen. At this point, you have three options.

1. Type W to uncancel the displayed transaction.
2. Type a new transaction number to move on.
3. Type Q to exit the cancel routine.

When an item is cancelled, you will see the message:

WRITING DISK

If you have been using the sample data provided in Table 6, cancel transactions 1.0, 100.0, 100.1, 101.0, 105.0, 107.0, 110.0, 110.1, and return to checking menu. If you have been entering your own data, cancel all but a few transactions to leave some easily recognizable outstanding checks. It will help you to learn about reconciliation in the next option if the outstanding items are easy to spot among the other figures.

Type 5 to Justify the Account with Bank Statement.

The reconciliation assumes a starting bank balance of $0 as though the account were just opened. For a previously

DISCOUNT COMPUTER SOFTWARE ACCESSORIES

APPLE

	Retail	Discount
Eliminator	$29.95	21.00
War	24.95	18.00
Adventureland	29.95	21.00
Pirates Adventure	29.95	21.00
Golden Voyage	29.95	21.00
Magic Window	99.95	72.00
Temple of Apshai	39.95	29.00
Upper Reaches of Apshai	19.95	15.00
Curse of Ra	19.95	15.00
Midway Campaign	16.00	12.00
Hi-Res Computer Golf	29.95	21.00
DOS Boss	24.00	18.00
The Arcade Machine	44.95	33.00
Star Blazer	31.95	23.00
Choplifter	34.95	25.00
Serpentine	34.95	25.00
Deadly Secrets	34.95	25.00
Raster Blaster	29.95	21.00
Bug Attack	29.95	21.00
The Home Accountant	74.95	54.00
Snack Attack	29.95	21.00
Pig Pen	29.95	21.00
Wordrace	24.95	18.00
Rendevous	39.95	29.00
Russki Duck	34.95	25.00
Horizon V	34.95	25.00
Sargon II	34.95	25.00
Zork I	39.95	29.00
Zork II	39.95	29.00
Deadline	49.95	36.00
Mastertype	39.95	29.00
Castle Wolfenstein	29.95	21.00
Supertext II	150.00	108.00
Softcard Premium System	775.00	600.00
Wizard and the Princess	32.95	24.00
Time Zone	99.95	72.00
Cranston Manor	34.95	25.00
Threshold	39.95	29.00
Softporn Adventure	29.95	21.00
Crossfire	29.95	21.00
Frogger	34.95	25.00
Laff Pak	34.95	25.00
Ultima II	59.95	44.00
Screenwriter II	129.95	94.00
Graphics Magician	59.95	44.00
Pie Man	29.95	21.00
Fastgammon	24.95	18.00
Congo	34.95	25.00
Goldrush	34.95	25.00
Gorgon	39.95	29.00
Beer Run	29.95	21.00
Snake Byte	29.95	21.00

Intec 32K Board $75.00
APPLE Compatible Disk Drive $265.00
VERBATIM/DATALIFE Disks $26.00

ATARI

	Retail	Discount
Threshold (d)	$39.95	29.00
Snake Byte (d)	29.95	21.00
Space Eggs (d)	29.95	21.00
Bandits (d)	34.95	29.00
Color Print (d)	39.95	29.00
Canyon Climber (d)	29.95	21.00
Shooting Arcade (d) (t)	29.95	21.00
Pacific Coast Highway (d) (t)	29.95	21.00
Clowns And Balloons (d) (t)	29.95	21.00
Wordrace (d)	24.95	18.00
Andromeda (d)	34.95	25.00
Deadline (d)	49.95	36.00
Zork I (d)	39.95	29.00
Zork II (d)	39.95	29.00
Alien Swarm (d)	34.95	25.00
Action Quest (d) (t)	29.95	21.00
Ghost Encounters (d) (t)	29.95	21.00
K-Razy Shootout (c)	49.95	36.00
K-Razy Kritters (c)	49.95	36.00
Ultima I (d)	39.95	29.00
Ali Baba and Forty Thieves (d)	32.95	24.00
Deluxe Invaders (c)	39.95	29.00
Gorf (c)	49.95	36.00
Wizard of Wor (c)	49.95	36.00
Preppie (d) (t)	29.95	21.00
Tigers in The Snow (d) (t)	39.95	29.00
Ghostly Manor (d)	24.95	18.00
Raster Blaster (d)	29.95	21.00

T=Cassette
D=Disk
C=Cartridge

ATARI®

	Retail	Discount
Rear Guard (d)	24.95	18.00
Rear Guard (t)	19.95	15.00
Caverns of Mars (d)	39.95	29.00
Atari Basic (c)	59.95	45.00
Star Raiders (c)	44.95	33.00
Centipede (c)	44.95	33.00
Pac Man (c)	44.95	33.00
Pilot (c)	79.95	60.00
Temple of Apshai (d) (t)	39.95	29.00
Upper Reaches of Apshai (t)	19.95	15.00
Curse of Ra (d)	19.95	15.00
Midway Campaign (t)	16.00	12.00
Apple Panic (d)	29.95	21.00
Track Attack (d)	29.95	21.00
Choplifter (d)	34.95	25.00
Star Blazer (d)	31.95	24.00
Wizard and the Princess (d)	32.95	24.00
Jawbreaker (d) (t)	29.95	21.00
Crossfire (d) (t)	29.95	21.00
Frogger (d) (t)	34.95	25.00
The Shattered Alliance (d)	39.95	29.00
Battle of Shiloh (d)	39.95	29.00
Submarine Commander (c)	49.95	39.00

SPECIAL OFFERS

MANY MORE PROGRAMS AVAILABLE — VISA AND MASTERCARD ACCEPTED

TERMS: Send check or money order for total purchase price, plus $2.00 for shipping. MI residents add 4% tax. C.O.D. accepted.

® MFGS. TRADEMARK

STRÖM SYSTEMS INC.
P.O. Box 197
Plymouth, Mi. 48170
(313) 455-8022

WRITE OR CALL FOR FREE CATALOG
PHONE ORDER HOURS
4 PM - 9 PM MON. - FRI.
INCLUDE CARD NUMBER AND EXPIRATION DATE WITH CREDIT CARD ORDERS.
INCLUDE TYPE OF COMPUTER.

existing account with an actual opening balance, add this amount when prompted for a new balance (described below). Note that the figure called BANK STATEMENT is the closing balance from the statement. Remember that we are showing sample data. Of course, if you are using your own data, the figures will be different.

To answer the first prompt, you would ordinarily enter the bank balance. Use $453.79 for this exercise.

ENTER CLOSING BALANCE FROM LAST BANK STATEMENT $

Entering that statement balance produces the following display.

Table 7.

*** ERROR ***	$0.00	(Not Displayed If No Error)	
Bank Statement	$453.79	Cancelled Checks	$546.21
Outstanding	-$414.60	Cancelled Deposits	$1000.00
Totals	$39.19		$453.79

Current Balance $39.19

ENTER NEW BALANCE TO BE USED BY PROGRAM $

The top line, "***ERROR*** $0.00," and the bottom line, which prompts you to ENTER A NEW BALANCE, will appear only if the balances do not reconcile.

The current balance will equal your checkbook balance if your check/deposit entry is up to date in the system. The New Balance to be used by the program allows you to reflect any bank service charges. Any interest you might receive on a checking account should be entered as a deposit for budget item #31, Interest. This type of entry will help at tax preparation time.

Upon entering NEW BALANCE, you will return to the checking menu.

Type 8 for Outstanding Check Status: (TRS-80 and IBM versions only).

Before closing, you may wish to type 8 for Program Status. This allows you to examine the changes that occurred as a result of the data you have entered and the operations you have performed.

This tells you that the cancelled check file has been opened, the outstanding check file has been updated and now carries 14 checks totalling $414.60. This figure can be used when estimating a budget for a pay period. It also gives you an indication of how much disk space you have left.

Hit ENTER and exit checking program by typing 0. This is the only safe way to exit the checking program because it closes all files and verifies your data. You will also see this message:

BE SURE TO RUN THE BUDGET PROGRAM IF YOU WANT THE RESULTS OF THIS SESSION INCLUDED IN FUTURE BUDGET REPORTS

You will return to BASIC when all the data is verified.

continued on page 32

NEW CLASSICS SOFTWARE

Pascal-80 **Phelps Gates**

This friendly, easy to use version of *Standard Pascal*, as reviewed in the December 1981 **Byte**, is now even better! New version works on TRS-80 Model I and Model III, under TRS-DOS, NewDOS, NewDOS 80, DOSPlus, LDOS, and DoubleDOS. An author package allows you to create your own */CMD* files without any royalty payments! Upper and lower case is fully supported. You can protect memory and call machine language programs. New extensions include SET, RESET, POINT, RND, and the UCSD Include procedure. Utilities are provided to convert to and from ASCII files. Pascal 80 now comes in a binder with an 80 page manual by George Blank.

With monitor, editor, and compiler in memory at the same time, no other Pascal is easier to learn! One college found that it could teach half again as many students on the same number of computers after switching from UCSD Pascal to Pascal 80.

Full 14 digit accuracy on all math functions, including log and trig functions, makes this a serious Pascal. Disk file handling is supported, with a mail list program included as a demonstration.

Upgrades are available for those who bought Ramware Pascal 80. Call or write for information.

Send $101 (includes shipping) to: **New Classic Software**
239 Fox Hill Road, Box S
Denville, NJ 07834

Credit card orders: (201) 625-8838

DISK DRIVE WOES?
PRINTER INTERACTION?
MEMORY LOSS?
ERRATIC OPERATION?

Don't Blame The Software!

Power Line Spikes, Surges & Hash could be the culprit! Floppies, printers, memory & processor often interact! Our patented ISOLATORS eliminate equipment interaction AND curb damaging Power Line Spikes, Surges and Hash. MONEY BACK GUARANTEE!

- **ISOLATOR (ISO-1) 3 filter isolated 3-prong sockets; integral Surge/Spike Suppression; 1875 W Maximumm load, 1 KW load any socket $76.95**
- **ISOLATOR (ISO-2) 2 filter isolated 3-rpong socket banks; (6 sockets total); integral Spike/Surge Suppression; 1875 W Max load, 1 KW either bank $76.95**
- **SUPER ISOLATOR (ISO-3) similar to ISO-1 except double isolation & Suppression $115.95**
- **SUPER ISOLATOR (ISO-11) similar to ISO-2 except double isolation & Suppression $115.95**
- **MAGNUM ISOLATOR (ISO-17) 4 Quad isolated sockets; For ULTRA-SENSITIVE Systems $200.95**
- **CIRCUIT BREAKER, any model (Add-CB) ..... Add $10.00**
- **REMOTE SWITCH, any model (Add-RS) ..... Add $18.00**

AT YOUR DEALERS

MasterCard, Visa, American Express
ORDER TOLL FREE 1-800-225-4876
(except AK, HI, PR & Canada)

ESP Electronic Specialists, Inc.
171 South Main Street, Box 389, Natick, Mass. 01760
(617) 655-1532

Deluxe Personal Finance *continued*

```
    SS SS SS SS SS SS SS SS SS SS SS
    SS                           SS
    SS        IBM PC BASIC       SS
    SS          'PFINIT'         SS
    SS   AUTHOR:  LANCE MICKLUS  SS
    SS   TRANSL:  RICH BOUCHARD  SS
    SS     COPYRIGHT (c) 1982    SS
    SS SOFTSIDE PUBLICATIONS, INC SS
    SS                           SS
    SS SS SS SS SS SS SS SS SS SS SS
```

If you don't wish to type this program, it is also included in this month's SoftSide DV.

```
10 CD$="CHECKING.DAT"
20 CA$="CANCELCK.DAT"
30 KD$="CKFILE.DAT"
40 SN$="FINANCE.SCO"
50 WIDTH 80
60 DEF FNC$(X$)=CHR$(ASC(X$+" ")+32*(ASC
(X$+" ")>96))
70 CLS
80 LOCATE 3,1
90 PRINT TAB(20);"INITIALIZATION PROCEDU
RE"
100 PRINT
110 PRINT"This program initializes a new
 data file disk.  If this is"
120 PRINT"the first time you've created
a data disk, load a formatted"
130 PRINT"but otherwise blank disk in on
e of your drives.  Otherwise,"
140 PRINT"load a BACKUP COPY of the most
 current data disk.  Do not"
150 PRINT"use the original copy since al
l data, except that for the"
160 PRINT"checking account program will
be erased.  If you forgot to"
170 PRINT"make a BACKUP COPY, hit (CTRL-
BREAK) and exit this program."
180 PRINT"Return to DOS and make a BACKU
P COPY of the current data disk"
190 PRINT"to use for a new initializatio
n."
200 PRINT
210 INPUT"Which drive is the disk in (A
or B)";DR$
220 DR$=FNC$(DR$)
230 IF DR$<"A" OR DR$>"B" THEN 210
240 IN$=DR$+":"+CD$
250 ON ERROR GOTO 280
260 OPEN"I",1,IN$:ON ERROR GOTO 0:CLOSE
1
270 GOTO 310
280 RESUME 290
290 ON ERROR GOTO 0:CLOSE 1
300 OPEN"O",1,IN$:PRINT #1,DATE$;"  ";TI
ME$:PRINT #1,0,0,"EOF":CLOSE
310 IN$=DR$+":"+KD$
320 OPEN"O",1,IN$:PRINT #1,DATE$;"  ";TI
ME$:PRINT #1,0,"EOF":CLOSE 1
330 IN$=DR$+":"+"BUDGET.DAT"
340 OPEN"O",1,IN$:PRINT #1,DATE$;"  ";TI
ME$:CLOSE 1
350 IN$=DR$+":"+SN$
360 OPEN"O",1,IN$:PRINT #1,DATE$;"  ";TI
ME$:CLOSE 1
370 IN$=DR$+":"+CA$
380 OPEN"O",1,IN$:PRINT #1,9999;9999;999
9;9999:CLOSE 1
```

HOTLINE

On February 2, 1983, the SoftSide Hotline will debut. Every Wednesday, from 7PM to 9PM Eastern Time, the SoftSide programmers will answer your questions about SoftSide programs. Call (603) 673-0585.

The Hotline can:

- Tell you the fix for any known program bugs, or mail you the long ones.
- Explain program instructions.
- Check out bugs you may have found.

The Hotline cannot:

- Answer subscription or editorial questions.
- Give programming advice.
- Accept collect calls.

When you call, follow these simple rules:

1. Write it down and, if possible, sit next to your computer when you call. This is a certain shortcut to your point when you are calling long distance.
2. Be specific. It betters our chance to zero in on your problem quickly. The first thing we'll want to know is what program you're inquiring about and what computer you're using.
3. Be brief. There are 50,000 of you and five programmers...
4. Before calling, **SWAT** the program, if we published a **SWAT** table with it. Most "bugs" are hidden typos. **SWAT** will find them and may save you a call.

Remember: The **Hotline** will give a fast response to the short questions. We will deal with long questions in writing.

(Programmers will be available to answer questions only during **Hotline** hours. Please don't call at any other time with your software questions.)

IBM PC® SWAT TABLE FOR: INITIALIZER

LINES	SWAT CODE	LENGTH
10 - 120	ED	339
130 - 220	PX	509
230 - 340	AV	311
350 - 380	SV	115

```
   SS SS SS SS SS SS SS SS SS SS SS
   SS        IBM PC BASIC        SS
   SS         'CHECKING'         SS
   SS   AUTHOR:  LANCE MICKLUS   SS
   SS   TRANSL:  RICH BOUCHARD   SS
   SS     COPYRIGHT (c) 1982     SS
   SS SOFTSIDE PUBLICATIONS, INC SS
   SS SS SS SS SS SS SS SS SS SS SS
100 DEFINT I-L,N,X-Z:DEFDBL B:WIDTH 80:D
EF FNC$(X$)=CHR$(ASC(X$+" ")+32*(ASC(X$+
" ")>96)):DIM A(1200),TI$(35):KEY OFF
140 DLR$="######.##":FX$="$######.##":SP$=
STRING$(79," "):GOSUB 4850:CLS:LOCATE 5,
1:PRINT STRING$(63,"-")
190 PRINT TAB(12);"THE PERSONAL CHECK BA
LANCE PROGRAM OF:":FOR Y=33 TO 35:PRINT
TAB(23);TI$(Y):NEXT Y:PRINT STRING$(63,"
-")
240 CK=0:DF=0:LP=0:CC$="CANCELCK.DAT":DN
$="CHECKING.DAT":SN$="FINANCE.SCO"
300 ZB=ASC(":"):BC=32:Q=0:R=0:S=0:T=0:TD
=0:EN$=DN$
330 ON ERROR GOTO 4380:OPEN"I",1,DN$:ON
ERROR GOTO 0:INPUT #1,DT$:INPUT #1,I,B:I
F I<3 THEN CLOSE 1:GOTO 1000
390 FOR N=0 TO I-1 STEP 3:GOSUB 4810:INP
UT #1,A(N),A(N+1),A(N+2):NEXT N:INPUT #1
,A$
440 IF A$="EOF" THEN 460 ELSE PRINT"FILE
 DATA BAD-Press (RETURN) to continue"
450 GOSUB 4920
460 CLOSE 1:T=1
1000 CLS:PRINT"As of ";DATE$;" your curr
ent"
1020 PRINT USING"CHECKING BALANCE is$$###
##.##";B:PRINT
1030 PRINT" 0 to END session.":IF I>2 TH
EN PRINT" 1 to LIST OUTSTANDING CHECK FI
LE."
1050 IF I<1198 THEN PRINT" 2 to ADD NEW
TRANSACTION(S) to file.":IF I<3 THEN I=0
:GOTO 1090
1070 PRINT" 3 to FIX TRANSACTION(S) with
 data errors.":PRINT" 4 to CANCEL CHECK(
S) received from the bank."
1090 PRINT" 5 to JUSTIFY the ACCOUNT wit
h bank statement."
1100 PRINT" 6 to ESTIMATE total BILLS du
e.":IF I>2 THEN PRINT" 7 to PRINT OUTSTA
NDING CHECK FILE."
1120 PRINT" 8 for OUTSTANDING CHECK STAT
US.":PRINT" 9 to RESET SCREEN.":PRINT:IN
PUT"Enter your CHOICE ";VA$:VA$=FNC$(VA$
)
1160 IF VA$>="0" AND VA$<="9" THEN M=VAL
(VA$) ELSE M=9
1170 CLS:IF (I<3) AND (M>0) AND (M<5) AN
D (M<>2) THEN 1230
1190 IF (I<3 AND M=7) THEN 1230
1200 IF M=1 THEN GOSUB 4110:GOTO 1000
1210 IF M=7 THEN GOSUB 4100:GOTO 1000
1220 ON M+1 GOTO 3240,1230,1240,1240,203
0,2780,2650,1230,3090,1000
1230 GOTO 1000
1240 IF I>1197 AND M=2 THEN 1000 ELSE GO
SUB 4950:IF C=0 THEN 1000
1260 IF (M=2) AND (I<3) THEN N=0:I=0:GOT
O 1820
1270 FOR N=0 TO I-1 STEP 3:IF (A(N)=C) A
ND (M=3) THEN 1410
1290 IF (A(N)=C) AND (M=2) THEN 1370
1300 NEXT N:IF M=2 THEN 1820
1320 CLS:PRINT:PRINT"Transaction #";C;"i
s not in the outstanding check file.":PR
INT:GOTO 1030
1370 GOSUB 3880:LOCATE 5,48:PRINT"Alread
y exists";
1390 LOCATE 12,1:PRINT"NEXT TRANSACTION"
:PRINT:GOTO 1240
```

continued on page 34

THE VOICE OF THE FUTURE . . .
HEAR TODAY

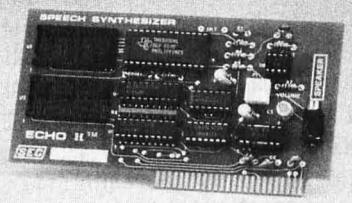
Echo Speech Synthesizers

Now your computer can talk with an ECHO™ speech synthesizer from Street Electronics!
Our SPEAKEASY™ phoneme system provides unlimited vocabulary while using a minimum of memory. The TEXTALKER™ text-to-speech firmware allows you to simply type in a word or phrase and let the computer do the rest. With nearly 400 grammatical rules contained in the system, your computer can properly pronounce most correctly spelled words. Simple commands allow you to select from:

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

The applications of the ECHO speech synthesizer are virtually unlimited, ranging from business and education to games to aiding the handicapped. The ECHO][which plugs into the Apple][is priced at **$150.00** The ECHO-GP is a complete stand alone unit which is compatible with most any computer; it sells for **$300.00**

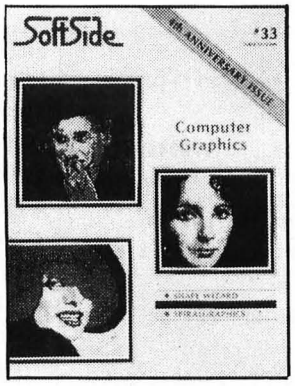
Street Electronics Corporation
1140 Mark
Carpinteria, California 93013
(805) 684-4593

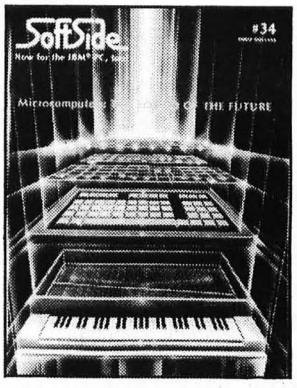


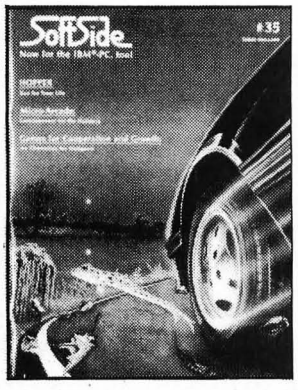
Deluxe Personal Finance *continued*

```
1410 GOSUB 3880:LOCATE 8,10:PRINT USING"
CURRENT BALANCE $######.##";B,:LOCATE 1,5
0:PRINT"OPTIONS";:LOCATE 2,48:PRINT"0=DO
NE";
1450 LOCATE 3,48:PRINT"1=FIX NUMBER";:LO
CATE 4,48:PRINT"2=FIX AMOUNT";:LOCATE 5,
48:PRINT"3=FIX ITEM NAME";
1480 LOCATE 6,48:PRINT"4=VOID";
1490 LOCATE 12,1:PRINT SP$:PRINT SP$:PRI
NT SP$
1500 LOCATE 10,11:GOSUB 5020:INPUT"Enter
 your CHOICE";VA$:A=VAL(VA$):LOCATE 12,1
:IF A=0 THEN IF M=2 THEN 1390 ELSE 1000
1530 IF A=4 THEN S=1:GOTO 1760
1540 IF A=3 THEN S=1:GOTO 1700
1550 IF A=2 THEN S=1:GOTO 1640
1560 IF A<>1 THEN 1490
1570 S=1:GOSUB 4950:IF C=0 OR A(N)=C THE
N 1490
1600 FOR K=0 TO I-1 STEP 3:IF A(K)<>C TH
EN NEXT K:A(N)=C:GOTO 1410
1620 LOCATE 12,1:PRINT SP$;:LOCATE 12,1:
PRINT"Transaction #";C;"already exists."
:GOTO 1500
1640 LINE INPUT"Enter Q to QUIT, or ente
r NEW AMOUNT: $";VA$:IF FNC$(VA$)="Q" TH
EN 1490 ELSE C=INT(VAL(VA$)*100)/100
1660 IF A(N+1)>0 THEN C=-C
1670 B=B-A(N+1)-C:A(N+1)=-C:GOTO 1410
1700 GOSUB 4720
1710 INPUT"Enter Q to QUIT, or enter NEW
 BUDGET NUMBER";VA$:IF FNC$(VA$)="Q" THE
N 1410 ELSE J=VAL(VA$)
1730 IF (J<0) OR (J>31) THEN LOCATE CSRL
IN-1,1:GOTO 1710
1740 A(N+2)=J:GOTO 1410
1760 B=B-A(N+1):FOR K=N TO I-4:A(K)=A(K+
3):NEXT K:I=I-3:GOTO 1000
1820 A(N)=C:LINE INPUT"Enter AMOUNT: $";
VA$:A(N+1)=INT(VAL(VA$)*100)/100
1840 LINE INPUT"CHECK or DEPOSIT (C/D)?"
;B$:B$=FNC$(B$):IF B$="C" THEN A(N+1)=-A
(N+1):GOTO 1890
1860 IF B$="D" THEN 1890
1870 LOCATE CSRLIN-1,1:GOSUB 5020:GOTO 1
840
1890 GOSUB 4720
1900 GOSUB 5020:LINE INPUT"Enter BUDGET
NUMBER: ";VA$:A(N+2)=VAL(VA$)
1920 IF (A(N+2)<0) OR (A(N+2)>32) OR (FN
C$(VA$)<"0") OR (FNC$(VA$)>"9") THEN LOC
ATE CSRLIN-1,1:GOTO 1900
1930 S=1:I=I+3:B=B+A(N+1):GOSUB 3880:LOC
ATE 1,52:PRINT"NEW";:LOCATE 3,50:PRINT"B
ALANCE";:LOCATE 5,50:PRINT USING FX$;B,
2000 LOCATE 8,1:PRINT"Type F to FIX this
 CHECK, any other key to CONTINUE";:A$=I
NPUT$(1):LOCATE 8,1:PRINT SP$;
2020 IF FNC$(A$)="F" THEN 1410 ELSE 1390
2030 IF T=2 THEN 2350
2040 CLS:EN$=CC$:ON ERROR GOTO 4380:OPEN
"I",3,CC$:ON ERROR GOTO 0:IF NOT EOF(3)
THEN INPUT #3,Z
2100 IF EOF(3) OR Z=9999 THEN 2230
2110 CLOSE 3:CLS:LOCATE 3,1:PRINT"Can't
CANCEL CHECKS":PRINT:PRINT"Unprocessed d
ata still in file.":PRINT
2180 PRINT"You must run MONTHLY BUDGET":
PRINT"program to clear the file.":PRINT
2200 LINE INPUT"Do you still wish to CAN
CEL CHECKS (Y/N)? ";A$:A$=FNC$(A$):IF A$
="N" THEN 1000
2220 IF A$<>"Y" THEN LOCATE CSRLIN-1,1:G
OSUB 5020:GOTO 2200
2230 CLOSE 3:OPEN"O",3,CC$:OPEN"O",2,SN$
:CC=-1
2270 CLS:LOCATE 1,6:PRINT"To which month
 should these cancelled checks be charge
d?"
2290 LINE INPUT"   Enter (1-12) ->";VA$:
Z=VAL(VA$):IF Z<1 OR Z>12 THEN 2270
2310 PRINT #3,Z:PRINT #2,Z:T=2:CLS
2350 GOSUB 4950
2360 IF C=0 THEN 1000
2370 FOR N=0 TO I-1 STEP 3:IF A(N)=C THE
N 2450
2390 NEXT N:CLS:PRINT:PRINT"Transaction
#";C;"is not outstanding.":PRINT:GOTO 23
50
2450 GOSUB 3880:PRINT"Enter W if last tr
ansaction cancelled was WRONG, otherwise
":FL=1:GOSUB 4950
2490 IF FNC$(VA$)="W" THEN CLS:GOTO 2350
2500 IF A(N+1)<=0 THEN Q=Q-A(N+1)
2510 IF A(N+1)>0 THEN R=R+A(N+1)
2520 LOCATE 8,15:PRINT"WRITING DISK";:
PRINT #3,A(N);A(N+1);A(N+2):PRINT #2,A(N
);A(N+1);A(N+2):FOR K=N TO I-1:A(K)=A(K+
3)
2570 NEXT K:S=1:I=I-3:IF I>2 THEN 2360
2610 CLS:PRINT"No more transactions in t
he file.":PRINT:GOTO 1020
2650 CLS:PRINT:PRINT:PRINT"Enter your bi
lls to be paid to see what the total is
and"
2680 PRINT"how much money you'll have le
ft over.":D=0
2700 LOCATE 11,1:PRINT"Enter Q to QUIT,
or":PRINT:LINE INPUT"enter BILL: $";VA$:
C=VAL(VA$):IF FNC$(VA$)="Q" THEN 1000
2730 D=D+C:CLS:LOCATE 3,1:PRINT"Your TOT
AL BILLS are $";:PRINT USING DLR$;D
2760 PRINT"which would leave your BALANC
E at $";:PRINT USING DLR$;B-D:GOTO 2700
2780 CLS:LOCATE 8,1:LINE INPUT"Enter bal
ance from last bank statement $";VA$:C=V
AL(VA$):D=0:IF I<3 THEN 2870
2840 FOR N=1 TO I-1 STEP 3:D=D+A(N):NEXT
 N
2870 CLS:LOCATE 5,39:PRINT"CANCELLED CHE
CKS: ";:PRINT USING FX$;Q,:LOCATE 6,37:P
RINT"CANCELLED DEPOSITS: ";:PRINT USING
FX$;R,
2900 LOCATE 5,1:PRINT"BANK STATEMENT ";:
PRINT USING FX$;C,:LOCATE 6,1:PRINT"OUTS
TANDING"," ";:PRINT USING FX$;D,
2920 LOCATE 7,1:PRINT" "," ---------";:L
OCATE 8,58:PRINT"---------";:LOCATE 8,1:
PRINT"TOTALS"," ";:PRINT USING FX$;C+D,
2950 LOCATE 9,57:PRINT USING FX$;R-Q,:LO
CATE 10,1:PRINT" ","CURRENT BALANCE $";:
PRINT USING DLR$;B,:E=ABS(B-(C+D))
2980 IF E>1 THEN 3030
2990 IF INT(E*1000)<>0 THEN 3030
3000 LOCATE 14,1
3010 PRINT"Hit (RETURN) to continue";
3020 GOSUB 4920:GOTO 1000
3030 LOCATE 1,1:PRINT"### ERROR ###","$
";:PRINT USING DLR$;ABS(B-(C+D)),:LOCATE
 14,1
3050 LINE INPUT"Enter NEW BALANCE to be
used by program, if desired $";VA$:IF VA
L(VA$)<>0 THEN B=VAL(VA$)
3070 S=1:GOTO 1000
3090 CLS:LOCATE 1,1:PRINT"CURRENT BALANC
E IS $";:PRINT USING DLR$;B:PRINT:PRINT"
O.C. FILE",
3130 IF S=0 THEN PRINT"UNCHANGED as of "
;DT$
3140 IF S=1 THEN PRINT"UPDATED"
3150 PRINT"# ON FILE",CHR$(29);INT(I/3):
C=0:IF I<3 THEN 3210
3180 FOR N=0 TO I-1 STEP 3:C=C+A(N+1):NE
XT N
3210 PRINT"OUTSTANDING","$";:PRINT USING
 DLR$;C:PRINT:GOTO 3010
3240 CLS
3250 IF T<>2 THEN 3510
3260 PRINT #3,9999;9999;9999:PRINT #2,99
99;9999;9999:PRINT #3,"EOF":PRINT #2,"EO
F":CLOSE 2,3:ZB=ASC("X"):BC=32
3320 OPEN"I",2,SN$:OPEN"I",3,CC$
3330 IF EOF(2) AND EOF(3) THEN 3490
3340 IF EOF(2) OR EOF(3) THEN 3400
3350 GOSUB 4810:LINE INPUT #2,A$:GOSUB 4
810:LINE INPUT #3,B$:IF A$=B$ THEN 3330
3400 LOCATE 1,1:PRINT SPACE$(78);:LOCATE
 1,1:PRINT"### DATA ERROR ###":PRINT"Can
 not verify file with CANCELLED CHECKS"
3420 PRINT"Press (RETURN) to continue. (
CTRL-BREAK) to ABORT"
3430 CLOSE:GOSUB 4920:OPEN"O",3,CC$:PRIN
T #3,9999;9999;9999;9999;"EOF":CLOSE 3:G
OTO 5140
```

continued on page 36







SoftSide Back Issues...Once They're Gone...Are They Gone Forever?

If you like the programs, reviews, and programming information in this issue of *SoftSide*...think of what's waiting for you in past issues!
Exciting games like *Defense*...*Hopper*...*Micro Man*...
Great Graphics like *Shape Wizard*...*Titan*...*Flight of the Bumblebee*...
PLUS...databases, utility programs, educational programs, and more!

It's all here in the Back Issues of ***SoftSide*** magazine. And many of these issues are still available for your enjoyment. But not all. Several issues are SOLD OUT...others are available but supply is limited. So check out the issues and features listed below and order today!

Back Issues.....$5.00 ea.*
Back Issue on cassette.....$12.95 ea.*
Back Issue w/ Enhanced Disk Version
(contains an additional program).....$19.95 ea.*

To order, use the bind-in card facing page 80. Send order card and payment to: *SoftSide* Magazine, 6 South St., Milford, NH 03055 (minimum order - $10.00.)

*Prices good for USA orders only — for foreign pricing see page 69.

October 1981 - *Leyte* - All Systems
Developing Data Base - Apple
Character Generator - ATARI®
EnvyrnTM - TRS-80®

Enhanced Disk Versions
Super Dairy Farming - Apple
Gameplay - TRS-80®

November 1981 - *Flight of the Bumblebee* - All Systems
Music Machine - Apple
Music Programmer - ATARI®
Music Editor - TRS-80®

Enhanced Disk Versions
National Anthems - Apple
Volleyball - ATARI®
Mean Checkers Machine - TRS-80®

December 1981 - *Titan* - All Systems
Aircraft Commander - Apple
Developing Data Base - ATARI®
Electronics Assistant - TRS-80®

Enhanced Disk Version
Bobsledding - Apple
Survive - ATARI®
Konane - TRS-80®

January 1982 - *Gambler* - All Systems
Microtext 1.1 - All Systems
Apple Capture - Apple
Piazza Hotel - ATARI®
TRS-Man - TRS-80®

Enhanced Disk Versions
Nuclear Submarine Adventure - Apple, TRS-80®
Death Star - ATARI®

February 1982 - *Space Rescue* - All Systems
Rubicube - Apple
Defense - ATARI®
Maze Sweep - TRS-80®

Enhanced Disk Versions
Andorra - Apple
Kismet II - ATARI®
Help Package - TRS-80®

March 1982 - *Hexapawn* - All Systems
Magical Shape Machine - Apple
Outer Space Attack - ATARI®
Killer Cars - TRS-80®

Enhanced Disk Versions
PEEKER/POKER - Apple
Curse of the Pharaoh - ATARI®
Warpath - TRS-80

April 1982 - *Microtext* - All Systems
Poster Maker - Apple
ATARI® Banner Machine - ATARI®
Database - TRS-80®

Enhanced Disk Versions
Semaphore - Apple
Renumbering for the ATARI® - ATARI®
Screen Print - TRS-80®

May 1982 - *Solitaire* - All Systems
Micro-Man - Apple
Cross Reference - ATARI®
Ladders - TRS-80®

Enhanced Disk Versions
List Formatter - Apple
Robot Battle - ATARI
Breakthru - TRS-80®

#30 - *Escape from the Dungeons of the Gods* - All Systems
SWAT - All Systems
Code Breaker - Apple
Sabotage - ATARI®
Piazza Hotel - TRS-80®

Enhanced Disk Versions
Dr. Livingston - Apple
Random Access Database - ATARI®
Kriegspiel II - TRS-80

#31 - *C.A.T.S.* - All Systems
Puzzle Jumble - Apple
Tunein - ATARI®
Dots - TRS-80®

Enhanced Disk Versions
Applesoft Extensions - Apple
Menu Plus - ATARI®
Starbase Gunner - TRS-80®

#32 - *Operation Sabotage* - All Systems
C.A.T.S. Computer Assisted Testing System Part II & III - All Systems

Enhanced Disk Versions
Cloze Test - Apple
Neat List - ATARI®
Up Periscope - TRS-80®

#33 - *Shape Wizard* - Apple
SpiralGraphics - ATARI®
Graphic Writer - TRS-80®

Enhanced Disk Versions
Hi-Res Character Generator - Apple
Paranoia - ATARI®
ILIST - TRS-80®

#34 - *Sabotage* - Apple
Pokey Player - ATARI®
PC Blues Box - IBM PC®
Tunein - TRS-80®

Enhanced Disk Versions
Auto Menu - Apple
The Rotberg Synthesizer - ATARI®
Cavern Quest - TRS-80®

#35 - *Hopper* - All Systems
Blockade - Apple
Saucer Formation - ATARI®
Fireman - TRS-80®

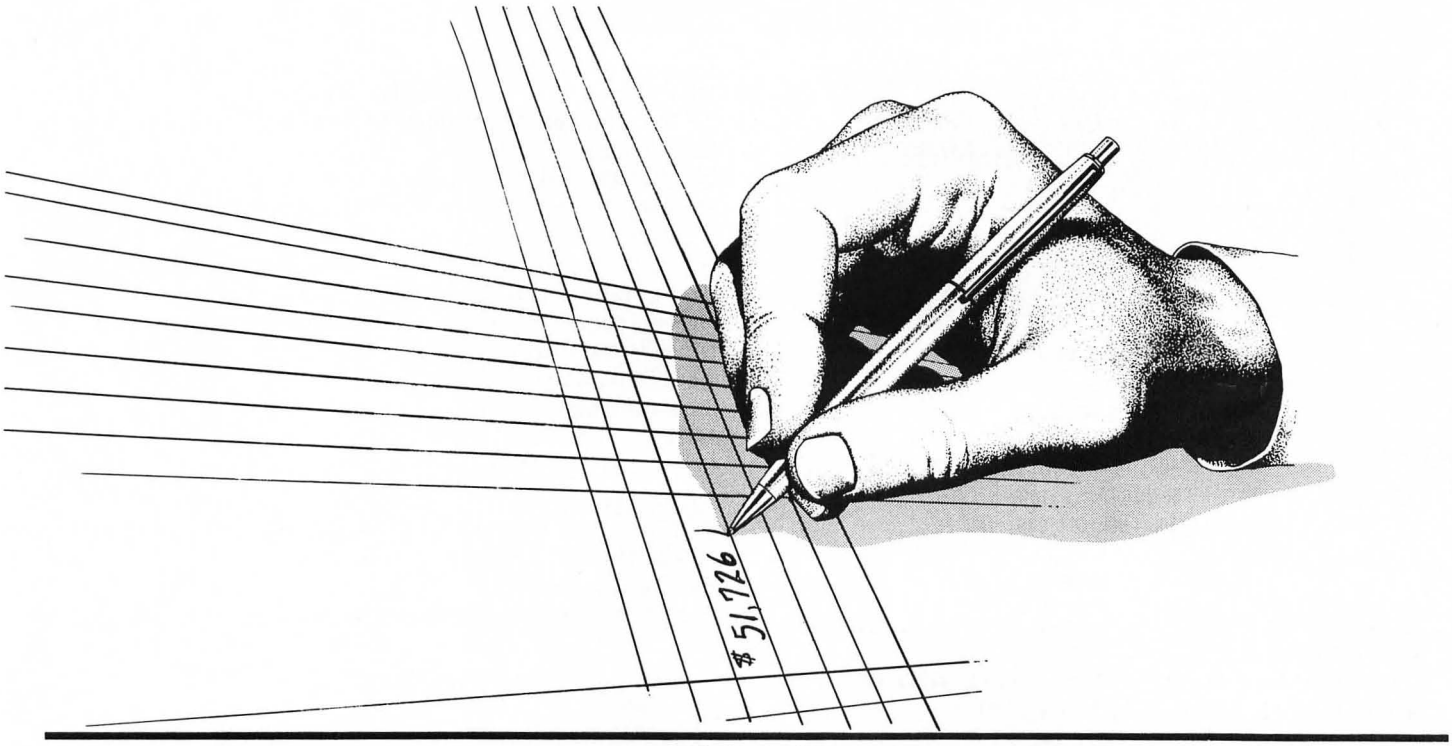
Enhanced Disk Versions
Sub Hunt - Apple
Deadstick Landing - ATARI®
Ping Pong/Hockey - TRS-80®

Deluxe Personal Finance *continued*

```
3490 CLOSE:T=0
3510 IF S=0 OR TD THEN 3860
3520 ZB=ASC("#"):BC=32:ON ERROR GOTO 444
0:FO$=SN$:DT$=DATE$+"  "+TIME$
3560 OPEN"O",1,FO$:T=3:PRINT #1,DT$:PRIN
T #1,I;B:IF I<3 THEN 3650
3610 FOR N=0 TO I-1 STEP 3:GOSUB 4810:PR
INT #1,A(N);A(N+1);A(N+2):NEXT N
3650 PRINT #1,"EOF":CLOSE 1:ON ERROR GOT
O 0
3680 IF FO$<>DN$ THEN FO$=DN$:ZB=ASC("*"
):BC=32:GOTO 3560
3690 IF DF THEN 3860
3700 OPEN"I",1,SN$:OPEN"I",2,DN$:ZB=ASC(
"V"):BC=32
3730 IF EOF(1) AND EOF(2) THEN 3850
3740 IF EOF(1) OR EOF(2) THEN PRINT"DISK
 ###ERROR###":GOTO 3810
3750 GOSUB 4810:LINE INPUT #1,A$:GOSUB 4
810:LINE INPUT #2,B$:IF A$=B$ THEN 3730
3800 PRINT"DATA ###ERROR###"
3810 PRINT"Press (RETURN) to retry"
3820 GOSUB 4920:CLOSE 1,2:GOTO 3520
3850 LOCATE 1,1:PRINT SPACE$(78);:LOCATE
 1,1:PRINT"VERIFIED"
3860 CLOSE 1,2:GOTO 5140
3880 CLS:IF A(N+1)>0 THEN 3980
3900 LOCATE 2,5:PRINT TI$(34);:LOCATE 3,
5:PRINT TI$(35);:LOCATE 2,25:PRINT USING
"CHECK ! #####.#";"#",A(N);
3930 LOCATE 4,5:PRINT STRING$(24,"-");:L
OCATE 4,5:PRINT TI$(A(N+2));" ";:LOCATE
4,31:PRINT USING"$#####.##";-A(N+1);
3960 LOCATE 6,21:PRINT TI$(33);:GOTO 401
0
3980 LOCATE 2,9:PRINT USING"DEPOSIT ! ##
###.#";"#",A(N);:LOCATE 4,9:PRINT USING"
AMOUNT ------------$#####.##";A(N+1)
4000 LOCATE 6,9:PRINT"FOR: ";TI$(A(N+2))
;
4010 LOCATE 1,1:PRINT CHR$(201);STRING$(
40,CHR$(205));CHR$(187);:LOCATE 7,1:PRIN
T CHR$(200);STRING$(40,CHR$(205));CHR$(1
88);
4030 FOR K=2 TO 6:LOCATE K,1:PRINT CHR$(
186);:LOCATE K,42:PRINT CHR$(186);:NEXT
K:LOCATE 9,1:PRINT STRING$(64,CHR$(254))
;
4080 LOCATE 11,1:RETURN
4100 LP=-1:GOTO 4120
4110 LP=0
4120 CLS:IF I<3 THEN RETURN
4140 X=0:X1=0:FOR K=0 TO I-1 STEP 3:IF X
=0 THEN PRINT"TRANSACTION #","   AMOUNT"
,"    ITEMIZED AS       TYPE":PRINT
4170 IF LP AND (X1=0) THEN LPRINT"TRANSA
CTION #","   AMOUNT","    ITEMIZED AS
   TYPE":LPRINT "
4180 A$=TI$(A(K+2)):B$="CHECK":IF A(K+1)
>0 THEN B$="DEPOSIT"
4200 FRMT$="  ######.#     $#####.##
    \             \  \       \":PRINT U
SING FRMT$;A(K),ABS(A(K+1)),A$,B$
4220 IF LP THEN LPRINT USING FRMT$;A(K),
ABS(A(K+1)),A$,B$
4230 X=X+1:X1=X1+1:IF (NOT LP) OR (X1<>5
0) THEN 4270
4250 LPRINT:LPRINT DATE$;TAB(48);"Outsta
nding Check File":LPRINT CHR$(12);:X1=0
4270 IF X<>10 THEN 4330
4280 X=0:IF LP THEN 4320
4290 PRINT:PRINT"Type Q to QUIT, any oth
er key to CONTINUE";:VA$=INPUT$(1):IF FN
C$(VA$)="Q" THEN RETURN
4320 CLS
4330 NEXT K:PRINT:IF NOT LP THEN LINE IN
PUT"** END OF LIST **          (RETURN)"
;B$:RETURN
4360 IF (X1/50)=INT(X1/50) THEN RETURN
4370 LPRINT:LPRINT DATE$;TAB(48);"Outsta
nding Check File";:LPRINT CHR$(12);:RETU
RN
4380 LOCATE 1,1:PRINT"Load disk with ";E
N$;" file on it.":PRINT"Hit (RETURN) to
continue."
4400 GOSUB 4920:LOCATE 1,1:PRINT SPACE$(
79):PRINT SPACE$(79):RESUME
4440 IF ERR<>61 THEN ON ERROR GOTO 0
4450 IF FO$=DN$ AND DF THEN 4620
4460 CLS:LOCATE 3,1:PRINT"DISK FULL-Can'
t verify output":PRINT:PRINT"Press (RETU
RN) to continue without VERIFYING."
4490 PRINT:GOSUB 4920:RESUME 4520
4520 CLOSE:OPEN"O",1,SN$:ON ERROR GOTO 4
610:PRINT #1,9999:CLOSE:DF=-1
4580 FO$=SN$:CLS:GOTO 3680
4610 RESUME NEXT
4620 LOCATE 3,1:PRINT"### E R R O R ###"
:PRINT"DISK FULL-DATA FILE DESTROYED":PR
INT
4650 PRINT"To try to recover, load anoth
er diskette with more":PRINT"file space
on it.  Press (RETURN) to continue."
4670 PRINT"Output will then be placed on
 the new disk.":PRINT:GOSUB 4920:CLOSE:R
ESUME 4580
4720 CLS:Y%=1:Y1%=1:FOR Z%=0 TO 32:LOCAT
E Y%,Y1%:PRINT Z%;:LOCATE Y%,Y1%+4:PRINT
 TI$(Z%);:Y%=Y%+1
4770 IF Y%>11 THEN Y%=Y%-11:Y1%=Y1%+26
4780 NEXT:LOCATE 13,1:RETURN
4810 LOCATE 1,1:IF BC=ZB THEN BC=32 ELSE
 BC=ZB
4820 PRINT CHR$(BC);:RETURN
4840 RETURN
4850 ON ERROR GOTO 4900:FOR Y%=0 TO 35:R
EAD TI$(Y%):NEXT Y%:READ A$:IF A$="EOF"
THEN ON ERROR GOTO 0:RETURN
4900 RESUME 4910
4910 PRINT"BAD DATA":GOTO 5140
4920 A$=INKEY$:IF A$=CHR$(13) THEN RETUR
N ELSE 4920
4930 WT$=TIME$
4940 IF WT$=TIME$ THEN 4940 ELSE RETURN
4950 PRINT"Enter Q to QUIT, or":PRINT
4960 GOSUB 5020:LINE INPUT"enter TRANSAC
TION NUMBER: ";VA$:IF FL=1 AND FNC$(VA$)
="W" THEN 5010
4980 IF FNC$(VA$)="Q" THEN C=0:RETURN EL
SE C=VAL(VA$)
4990 C=INT(C#10)/10:IF ABS(C)<1 OR ABS(C
)>=100000! THEN LOCATE CSRLIN-1,1:GOTO 4
960
5010 FL=1:RETURN
5020 Y%=POS(0):PRINT SPACE$(79-Y%);:LOCA
TE,Y%:RETURN
5050 DATA DENTIST/DOCTOR,MEDICAL AIDS,PH
ARMACY,MEDICAL INS,MED MILEAGE
5060 DATA INTEREST/MTG,INTEREST/OTHER,FI
XED TAXES,OTHER TAXES,DONATIONS
5070 DATA LOSS,EDUCATIONAL EXP,"UNION DU
ES, ETC",CHILD CARE,RENT/MTG PRINC
5080 DATA UTILITIES,INSURANCE,GROCERIES,
LOAN PRINCIPAL,HOME REPAIRS
5090 DATA SAVINGS,AUTO REPAIRS,GAS/OIL,V
AC/ENTERTAINM'T,CLOTHES,FURNISHINGS
5100 DATA HOUSEHOLD ITEM,MISC EXPENSES,O
PEN ITEM,SALARY,MISC DEPOSITS,INTEREST,C
HECKING/CASH
5120 DATA YOUR NAME HERE,123 MAIN STREET
,ANYTOWN USA
5130 DATA EOF
5140 END
```

IBM PC® SWAT TABLE FOR: CHECKING

LINES	SWAT CODE	LENGTH
100 - 390	EF	518
440 - 1100	EI	551
1120 - 1290	RT	483
1300 - 1540	WD	519
1550 - 1740	AI	498
1760 - 2000	ZV	556
2020 - 2270	ZG	507
2290 - 2570	JE	508
2610 - 2870	NW	590
2900 - 3070	WN	514
3090 - 3340	WD	433
3350 - 3680	EE	525
3690 - 3900	ND	412
3930 - 4140	LQ	527
4170 - 4290	WG	504
4320 - 4580	DJ	493
4610 - 4900	BN	510
4910 - 5060	XD	470
5070 - 5140	TK	361



TRS-80®

```
SS SS SS SS SS SS SS SS SS SS SS
SS                              SS
SS       TRS-80 BASIC           SS
SS         'PFINIT'             SS
SS   AUTHOR:  LANCE MICKLUS     SS
SS     COPYRIGHT (c) 1982       SS
SS SOFTSIDE PUBLICATIONS, INC   SS
SS                              SS
SS SS SS SS SS SS SS SS SS SS SS
```

If you don't wish to type this program, it is also included in this month's SoftSide CV and DV.

```
10 CLS:CLEAR500
20 DEFUSR0=673:DEFUSR1=664
30 DEFINTA-Z
40 CD$="CHECKING/DAT"
50 CA$="CANCELCK/DAT"
60 KD$="CKFILE/DAT"
70 SN$="FINANCE/SCO"
80 CLS
90 PRINT@128,"";
100 PRINTTAB(20)"INITIALIZATION PROCEDURE"
110 PRINT
120 PRINT"THIS PROGRAM INITIALIZES A NEW DATA FILE DISK.  IF THI
S IS"
130 PRINT"THE FIRST TIME YOU'VE CREATED A DATA DISK, LOAD A FORM
ATTED"
140 PRINT"BUT OTHERWISE BLANK DISK IN ONE OF YOUR DRIVES.  OTHER
WISE,"
150 PRINT"LOAD A BACKUP COPY OF THE MOST CURRENT DATA DISK.  DO
NOT"
160 PRINT"USE THE ORIGINAL COPY SINCE ALL DATA, EXCEPT THAT FOR
THE"
170 PRINT"CHECKING ACCOUNT PROGRAM WILL BE ERASED.  IF YOU FORGO
T TO"
180 PRINT"MAKE A BACKUP COPY, HIT (BREAK) AND EXIT THIS PROGRAM.
"
190 PRINT"RETURN TO DOS AND MAKE A BACKUP COPY OF THE CURRENT DA
TA DISK"
200 PRINT"TO USE FOR A NEW INITIALIZATION."
210 PRINT
220 INPUT"WHICH DRIVE IS THE DISK IN (0 TO 3)";DR$
230 IFDR$<"0"ORDR$>"3"THEN80
240 IN$=CD$+":"+DR$
250 ONERRORGOTO300
260 OPEN"I",1,IN$
270 ONERRORGOTO0
280 CLOSE1
290 GOTO370
300 RESUME310
310 ONERRORGOTO0
320 CLOSE1
330 OPEN"O",1,IN$
340 PRINT#1,TIME$
350 PRINT#1,0,0,"EOF"
360 CLOSE
370 IN$=KD$+":"+DR$
380 OPEN"O",1,IN$
390 PRINT#1,TIME$
400 PRINT#1,0,"EOF"
410 CLOSE1
420 IN$="BUDGET/DAT"+":"+DR$
430 OPEN"O",1,IN$
440 PRINT#1,TIME$
450 CLOSE1
460 IN$=SN$+":"+DR$
470 OPEN"O",1,IN$
480 PRINT#1,TIME$
490 CLOSE1
500 IN$=CA$+":"+DR$
510 OPEN"O",1,IN$
520 PRINT#1,9999;9999;9999;9999
530 CLOSE1
540 CLS:PRINT"DATA FILE INITIALIZATION COMPLETE":END
```

TRS-80® SWAT TABLE FOR: INITIALIZER

LINES	SWAT CODE	LENGTH
10 - 120	HE	257
130 - 200	BY	501
210 - 320	MS	173
330 - 440	VD	171
450 - 540	HK	174

```
SS SS SS SS SS SS SS SS SS SS SS
SS                           SS
SS       TRS-80 BASIC        SS
SS       'CHECKING'          SS
SS   AUTHOR:  LANCE MICKLUS  SS
SS     COPYRIGHT (c) 1982    SS
SS SOFTSIDE PUBLICATIONS, INC SS
SS                           SS
SS SS SS SS SS SS SS SS SS SS SS
```

```
10 CLS:CLEAR500:DEFUSR0=673:DEFUSR1=664:DEFINTI-L,N,X-Z:DEFDBLB:
DIMA(1200),TI$(35)
20 CRLF$=CHR$(29)+CHR$(26):BIG$=CHR$(23):DLR$="#####.##":FX$="$$
####.##":GOSUB1900:PRINT@320,STRING$(63,"-")
30 PRINTTAB(12);"THE PERSONAL CHECK BALANCE PROGRAM OF:":FORY=33
TO35:PRINTTAB(23);TI$(Y):NEXTY:PRINTSTRING$(63,"-"):CK=0:DF=0
40 LP=0:CC$="CANCELCK/DAT":DN$="CHECKING/DAT":SN$="FINANCE/SCO":
FC$=CC$:FD$=DN$:FL$=LD$:FS$=SN$
50 GOSUB1880:ZB=ASC(":"):Q=0:R=0:S=0:T=0:TD=0:EN$=DN$:ONERRORGOT
01670:OPEN"I",1,FD$:ONERRORGOTO0:INPUT#1,DT$
60 PRINT@965,"LAST FILE UPDATE:",LEFT$(DT$,8),RIGHT$(DT$,8);:INP
UT#1,I,B:IFI<3THENCLOSE1:GOTO110
70 FORN=0TOI-1STEP3:GOSUB1820:INPUT#1,A(N),A(N+1),A(N+2):NEXTN:I
NPUT#1,A$
80 IFA$="EOF"THEN100ELSEPRINT"FILE DATA BAD - PRESS (ENTER) TO C
ONTINUE"
90 GOSUB1910
100 CLOSE1:T=1
110 CLS:GOSUB1880:PRINT"AS OF ";LEFT$(TIME$,8);" YOUR CURRENT"
120 PRINTUSING"CHECKING BALANCE IS$$#####.##";B:PRINT
130 PRINT"0 TO END SESSION.":IFI<3THEN150
140 PRINT"1 TO LIST OUTSTANDING CHECK FILE."
150 PRINT"2 TO ADD NEW CHECK TO FILE.":IFI<3THENI=0:GOTO170
160 PRINT"3 TO FIX CHECKS WITH DATA ERRORS.":PRINT"4 TO CANCEL C
HECKS RECEIVED FROM THE BANK."
170 PRINT"5 TO JUSTIFY THE ACCOUNT WITH BANK STATEMENT."
180 PRINT"6 TO ESTIMATE TOTAL BILLS DUE.":IFI>2PRINT"7 TO PRINT
OUTSTANDING CHECK FILE"
190 PRINT"8 FOR OUTSTANDING CHECK STATUS.":PRINT"9 TO RESET SCRE
EN.":PRINT:INPUT"ENTER YOUR CHOICE ";M:GOSUB1860:CLS
200 IF(I<3)AND(M>0)AND(M<5)AND(M<>2)THEN250
210 IFI<3ANDM=7THEN250
220 IFM=1GOSUB1480:GOTO110
230 IFM=7GOSUB1470:GOTO110
240 ONM+1GOTO1120,250,270,270,640,950,900,250,1070,110
250 GOTO110
260 PRINT"TYPE 'Q' TO QUIT OR"
270 LINEINPUT"ENTER TRANSACTION NUMBER: ";VA$:IFVA$=""THEN270
280 IFLEFT$(VA$,1)="Q"THEN110ELSEC=VAL(VA$)
290 IF(M=2)AND(I<3)LETN=0:I=0:GOTO530
300 FORN=0TOI-1STEP3:IF(A(N)=C)AND(M=3)THEN370
310 IF(A(N)=C)AND(M=2)THEN340
320 NEXTN:IFM=2THEN530
330 CLS:PRINT"TRANSACTION #";CDBL(C$10)/10.0;"IS NOT IN THE OUTS
TANDING CHECK FILE.":PRINT:GOTO130
340 GOSUB1380:FORK=0TO3:PRINT@177,"ALREADY EXISTS";:GOSUB1920:PR
INT@177,"              ";:GOSUB1920:NEXTK
350 PRINT@448,CHR$(30);:PRINT@449,">>> TYPE 'Q' IF NO MORE TRANS
ACTIONS <<<";
360 PRINT@640,CHR$(30);:PRINT@640,"NEXT TRANSACTION":GOTO270
370 GOSUB1380:PRINT@457,"";:PRINTUSING"CURRENT BALANCE $#####.##
";B,:PRINT@49,"OPTIONS";:PRINT@111,"0=DONE";
380 PRINT@175,"1=FIX #";:PRINT@239,"2=FIX AMOUNT";:PRINT@303,"3=
FIX ITEM NAME";:PRINT@367,"4=VOID";:PRINT@640,"";
390 INPUT"ENTER YOUR CHOICE ";A:IFA=0THENIFM=2THEN360ELSE110
400 IFA=4THENS=1:GOTO520
410 IFA=3THENS=1:GOTO490
420 IFA=2THENS=1:GOTO470
430 IFA<>1PRINT"WHAT?":GOTO390
440 S=1:LINEINPUT"ENTER NEW TRANSACTION #";VA$:C=VAL(VA$):FORK=0
TOI-1STEP3
450 IFA(K)=CPRINT"TRANSACTION #";CDBL(C$10)/10.0;"ALREADY EXISTS
.":GOTO390
460 NEXTK:A(N)=C:GOTO370
470 LINEINPUT"ENTER NEW AMOUNT $";VA$:C=VAL(VA$):IFA(N+1)>0THENC
=-C
480 B=B-A(N+1)-C:A(N+1)=-C:GOTO370
490 GOSUB1800
500 INPUT"ENTER NEW BUDGET NUMBER";J:IF(J<0)OR(J>31)PRINT"WHAT?"
:GOTO500
510 A(N+2)=J:GOTO370
520 B=B-A(N+1):FORK=NTOI-4:A(K)=A(K+3):NEXTK:I=I-3:GOTO110
530 A(N)=C:LINEINPUT"ENTER AMOUNT $ ";VA$:A(N+1)=VAL(VA$)
540 LINEINPUT"CHECK OR DEPOSIT?";B$:IFB$=""THEN540
550 IFLEFT$(B$,1)="C"THENA(N+1)=-A(N+1):GOTO580
560 IFLEFT$(B$,1)="D"THEN580
570 PRINT"WHAT?":GOTO540
580 GOSUB1800
590 LINEINPUT"ENTER BUDGET NUMBER #";VA$:IFVA$=""THEN590ELSEA(N+
2)=VAL(VA$)
600 IF(A(N+2)<0)OR(A(N+2)>32)THEN590
610 S=1:I=I+3:B=B+A(N+1):GOSUB1380:PRINT@115,"NEW";:PRINT@177,"B
ALANCE";:PRINT@304,"";:PRINTUSINGFX$;B,
620 PRINT@448,"TYPE F TO FIX THIS CHECK, OR ANY OTHER KEY TO CON
TINUE: ";
630 A$=INKEY$:IFA$=""THEN630ELSEIFA$="F"ORA$=CHR$(102)THEN370ELS
E350
640 IFT=2THEN760
650 CLS:EN$=CC$:ONERRORGOTO1670:OPEN"I",3,FC$:ONERRORGOTO0:IFNOT
EOF(3)THENINPUT#3,Z
660 IFEOF(3)ORZ=9999THEN720
670 CLOSE3:CLS
680 PRINT@128,"WARNING: IF YOU CANCEL CHECKS AT THIS TIME, CERTA
IN DATA WILL":PRINT"BE LOST TO THE BUDGET PROGRAM FOR FUTURE REP
ORTS."
690 PRINT:PRINT:PRINT:LINEINPUT"DO YOU STILL WISH TO CANCEL CHEC
KS (Y/N)? ";A$
700 IFLEFT$(A$,1)="N"THEN110
710 IFLEFT$(A$,1)<>"Y"THENINPUT"Y OR N";A$:GOTO700
720 CLOSE3:OPEN"O",3,FC$:OPEN"O",2,FS$:CC=-1
730 CLS:PRINT@320,"TO WHICH MONTH SHOULD THESE CANCELLED CHECKS
BE CHARGED?":LINEINPUT"ENTER (1-12) -> ";VA$:Z=VAL(VA$)
740 IFZ<1ORZ>12THEN730
750 PRINT#3,Z:PRINT#2,Z:T=2:CLS
760 LINEINPUT"ENTER TRANSACTION NUMBER: ";VA$:IFVA$=""THEN760
770 IFLEFT$(VA$,1)="Q"THEN110ELSEC=VAL(VA$)
780 FORN=0TOI-1STEP3:IFA(N)=CTHEN800
790 NEXTN:CLS:PRINT"TRANSACTION #";CDBL(C$10)/10.0;"IS NOT OUTST
ANDING.":PRINT:GOTO760
```

```
800 GOSUB1380
810 PRINT"IF WRONG TRANSACTION, TYPE 'W' TO UNCANCEL.":PRINT"IF
RIGHT TRANSACTION, ENTER NEXT TRANSACTION NUMBER.":PRINT"TO EXIT
, TYPE 'Q'."
820 PRINT
830 LINEINPUT"ENTER TRANSACTION NUMBER: ";VA$:IFVA$=""THEN830
840 IFLEFT$(VA$,1)="W"THENCLS:GOTO760ELSEC=VAL(VA$)
850 IFA(N+1)<=0THENQ=Q-A(N+1)
860 IFA(N+1)>0THENR=R+A(N+1)
870 PRINT@462,"*WRITINGDISK*";:PRINT#3,CDBL(A(N));A(N+1);A(N+2):
PRINT#2,CDBL(A(N));A(N+1);A(N+2):FORK=NTOI-1:A(K)=A(K+3):NEXTK
880 S=1:I=I-3:IFI>2THEN770
890 CLS:PRINT"NO MORE TRANSACTIONS IN THE FILE.":PRINT:GOTO120:C
LS
900 CLS:GOSUB1880:PRINT:PRINT:PRINT"ENTER YOUR BILLS TO BE PAID
TO SEE WHAT THE TOTAL IS AND"
910 PRINT"HOW MUCH MONEY YOU'LL HAVE LEFT OVER.":D=0
920 PRINT@640,"ENTER 'Q' TO STOP.":LINEINPUT"ENTER BILL $";VA$:I
FLEFT$(VA$,1)THEN110ELSEC=VAL(VA$)
930 D=D+C:CLS:PRINT@128,"YOUR TOTAL BILLS ARE $";:PRINTUSINGDLR$
;D
940 PRINT"WHICH WOULD LEAVE YOUR BALANCE AT $";:PRINTUSINGDLR$;B
-D:GOTO920
950 CLS:GOSUB1880:PRINT@448,"";:LINEINPUT"ENTER BALANCE FROM LAS
T BANK STATEMENT $";VA$:C=VAL(VA$):D=0:IFI<3THEN970
960 FORN=1TOI-1STEP3:D=D+A(N):NEXTN:CLS
970 PRINT@290,"CANCELLED CHECKS: ";:PRINTUSINGFX$;Q,:PRINT@352,"
CANCELLED DEPOSITS: ";:PRINTUSINGFX$;R,
980 PRINT@256,"BANK STATEMENT",:PRINTUSINGFX$;C,:PRINT@320,"OUTS
TANDING",:PRINTUSINGFX$;D,:PRINT@384," ","---------";
990 PRINT@436,"---------";:PRINT@448,"TOTALS",:PRINTUSINGFX$;C+D
,:PRINT@500,"";:PRINTUSINGFX$;R-Q,
1000 PRINT@576," ","CURRENT BALANCE $";:PRINTUSINGDLR$;B,:E=ABS(
B-(C+D)):IFE>1THEN1040
1010 IFINT(E*1000)<>0THEN1040
1020 PRINT@832,"";
1030 PRINT"HIT (ENTER) TO CONTINUE";:GOSUB1910:GOTO110
1040 PRINT@0,"*** ERROR ***","$";:PRINTUSINGDLR$;ABS(B-(C+D)),:P
RINT@832,"";
1050 LINEINPUT"ENTER NEW BALANCE TO BE USED BY PROGRAM $";VA$:IF
VA$<>""THENB=VAL(VA$)
1060 S=1:GOTO110
1070 CLS:GOSUB1880:PRINT@0,"CURRENT BALANCE IS $";:PRINTUSINGDLR
$;B:PRINT"O.C. FILE",:IFS=0PRINT"UNCHANGED AS OF ";DT$
1080 IFS=1PRINT"UPDATED"
1090 PRINT"# ON FILE",INT(I/3):C=0:IFI<3THEN1110
1100 FORN=0TOI-1STEP3:C=C+A(N+1):NEXTN
1110 PRINT"OUTSTANDING","$";:PRINTUSINGDLR$;C:PRINT:GOTO1030
1120 CLS:FR$=RT$:GOSUB1880:IFT<>2THEN1210
1130 PRINT#3,9999;9999;9999:PRINT#2,9999;9999;9999:PRINT#3,"EOF"
:PRINT#2,"EOF":CLOSE2,3:ZB=ASC("X"):OPEN"I",2,FS$:OPEN"I",3,FC$
1140 IFEOF(2)ANDEOF(3)THEN1200
1150 IFEOF(2)OREOF(3)THEN1170
1160 GOSUB1820:LINEINPUT#2,A$:GOSUB1830:LINEINPUT#3,B$:IFA$=B$TH
EN1140
1170 PRINT@0,CHR$(30);"*** DATA ERROR ***":PRINT"CAN NOT VERIFY
FILE WITH CANCELLED CHECKS"
1180 PRINT"PRESS (ENTER) TO CONTINUE. - (BREAK) TO ABORT":CLOSE:
GOSUB1910:OPEN"O",3,FC$:PRINT#3,9999;9999;9999;9999;"EOF"
1190 CLOSE3:EN$=RT$:F1$=FR$:GOTO1360
1200 CLOSE:T=0
1210 IFS=0ORTDTHEN1350
1220 ZB=ASC("#"):ONERRORGOTO1690:FO$=FS$:DT$=TIME$
```

continued on page 40

Is our rating system too hard on software?

Peelings II The magazine of Apple software and hardware evaluation

Yes, we're critical. Too critical? No. We don't pass on to you reworked ad copy; we present balanced and complete criticism.

Fair and Factual

In recent issues our experienced reviewers have examined 6502-based word processors, utilities, graphics packages, and assemblers. Our future issues will be about the latest data management programs, business and financial applications, and communication software. We explain how well a piece of software or hardware does what it was designed to do, and we often rate it against comparable products. We're not afraid to tell you which products do not measure up. And our highly respected rating system will tell you everything you need and want to know about new programs, everything from AAA to F. From our reviews, to our editorials, to our special comparative feature columns, you'll find *Peelings II* fair and factual.

Find Out What Thousands of Other Apple Owners Already Know

We take the time and devote our resources to staying current with the latest releases for the Apple II, so you can look in one publication for information you need to make informed choices.

Peelings II™

THE MAGAZINE OF APPLE SOFTWARE AND HARDWARE EVALUATION

2260 Oleander, Las Cruces, New Mexico 88001

Special Introductory Offer. 9 Issues for $18.95 Offer Expires March 1, 1983

Subscribe Today!

Send to:
Name _____
Address _____
City _____ State _____ Zip _____
Charge to:
☐ Master Card ☐ Visa _____
Exp. Date: _____
Signature _____

☐ Send First Class Mail (Extra $15.00 per year) required for Canada, Mexico, APO, FPO
☐ South American and European subscribers add $27.00 per year for Air Mail Post
☐ —All other foreign add $36.00 per year for Air Mail Post.

Apple is a registered trademark of Apple Computer, Inc.

Deluxe Personal Finance *continued*

```
1230 OPEN"O",1,FO$:T=3:PRINT#1,DT$:PRINT#1,1;B:IFI<3THEN1250
1240 FORN=0TOI-1STEP3:GOSUB1820:PRINT#1,CDBL(A(N)*10)/10;A(N+1);
A(N+2):NEXTN
1250 PRINT#1,"EOF":CLOSE1:ONERRORGOTO0
1260 IFFO$<>FD$LETFO$=FD$:ZB=ASC("*"):GOTO1230
1270 IFDFTHEN1350
1280 OPEN"I",1,FS$:OPEN"I",2,FD$:ZB=ASC("V")
1290 IFEOF(1)ANDEOF(2)THEN1340
1300 IFEOF(1)OREOF(2)PRINT"DISK ***ERROR***":GOTO1330
1310 GOSUB1820:LINEINPUT#1,A$:GOSUB1830:LINEINPUT#2,B$:IFA$=B$TH
EN1290
1320 PRINT"DATA ***ERROR***"
1330 PRINT"PRESS (ENTER) TO RETRY":GOSUB1910:CLOSE1,2:GOTO1220
1340 PRINT@0,CHR$(30);"VERIFIED"
1350 CLOSE1,2
1360 CLS:PRINT"BE SURE TO RUN THE BUDGET PROGRAM IF YOU WANT THE
 RESULTS OF":PRINT"THIS SESSION INCLUDE IN FUTURE BUDGET REPORTS
 ":PRINT:PRINT
1370 END
1380 CLS:IFA(N+1)>0THEN1430
1390 PRINT@68,TI$(34);:PRINT@132,TI$(35);:PRINT@88,"";:PRINTUSIN
G"CHECK ! ######.#";"#",CDBL(A(N)*10)/10.0;
1400 PRINT@196,STRING$(24,"-");:PRINT@196,TI$(A(N+2));" ";:PRINT
@222,"";:PRINTUSING"$####.##";-A(N+1);:FORK=7TO33
1410 IFRND(2)=1THENSET(K,15)
1420 NEXTK:PRINT@340,TI$(33);:GOTO1450
1430 PRINT@72,"";:PRINTUSING"DEPOSIT ! ######.#";"#",CDBL(A(N)*1
0)/10.0;
1440 PRINT@200,"";:PRINTUSING"AMOUNT ----------$####.##";A(N+1
):PRINT@328,"FOR: ";TI$(A(N+2));
1450 PRINT@0,STRING$(42,CHR$(140));:PRINT@384,STRING$(42,CHR$(13
1));:FORK=1TO18:SET(0,K):SET(1,K):SET(84,K):SET(85,K):NEXT
1460 PRINT@512,STRING$(64,CHR$(160));:PRINT@640,"";:RETURN
1470 LP=-1:GOTO1490
1480 LP=0
1490 CLS:IFI<3THENRETURN
1500 X=0:FORK=0TOI-1STEP3:IFX<>0THEN1530
1510 PRINT"TRANSACTION #","  AMOUNT","ITEMIZED AS","TYPE":IFLPTH
ENLPRINT"TRANSACTION #","  AMOUNT","ITEMIZED AS","TYPE":LPRINT
1520 PRINT
1530 A$=TI$(A(K+2)):B$="CHECK":IFA(K+1)>0B$="DEPOSIT"
1540 FRMT$="  ######.#     $$#####.##        %              %%
   %":PRINTUSINGFRMT$;CDBL(A(K)*10)/10.0,ABS(A(K+1)),A$,B$
1550 IFLPTHENLPRINTUSINGFRMT$;CDBL(A(K)),ABS(A(K+1)),A$,B$
1560 X=X+1:IFX<11THEN1650
1570 PRINT:IFLP=0THEN1610
1580 LPRINTLEFT$(TIME$,8);TAB(48);"OUTSTANDING CHECK FILE"
1590 IFLP=-1THENLPRINT" ":LPRINT" ":LPRINT" ":LP=-2ELSELPRINTCHR
$(12);:LP=-1
1600 GOTO1640
1610 PRINT"TYPE '1' TO CONTINUE OR '2' TO ABORT.";
1620 VA$=INKEY$:IFVA$=""THEN1620
1630 IFVA$="2"THENRETURN
1640 CLS:X=0
1650 NEXTK:PRINT:IFLPTHENLPRINTCHR$(12);:RETURN
1660 LINEINPUT"** END OF LIST **       HIT (ENTER)";B$:RETURN
1670 PRINT@0,"LOAD DISK WITH ";EN$;" FILE ON IT.":PRINT"HIT (ENT
ER) TO CONTINUE.":GOSUB1910:PRINT@0,CHR$(30):PRINTCHR$(30)
1680 RESUME
1690 IFERR/2+1<>62THENONERRORGOTO0
1700 IFFO$=FD$ANDDFTHEN1770
1710 CLS:PRINT@128,"DISK FULL - CAN'T VERIFY OUTPUT"
1720 PRINT:PRINT"PRESS (ENTER) TO CONTINUE WITHOUT VERIFYING."
1730 PRINT:GOSUB1910:RESUME1740
1740 CLOSE:OPEN"O",1,FS$:ONERRORGOTO1760:PRINT#1,9999:CLOSE:DF=-
1
1750 FO$=FS$:CLS:GOTO1260
1760 RESUMENEXT
1770 PRINT@128,"*** E R R O R ***":PRINT"DISK FULL - DATA FILE D
ESTROYED":PRINT
1780 PRINT"TO TRY TO RECOVER, LOAD ANOTHER DISK WITH MORE":PRINT
"FILE SPACE ON IT.  PRESS (ENTER) TO CONTINUE."
1790 PRINT"OUTPUT WILL THEN BE PLACED ON THE NEW DISK.":PRINT:GO
SUB1910:CLOSE:RESUME1750
1800 CLS:Y%=0:FORZ%=0TO32:PRINT@Y%,Z%;:PRINT@(Y%+4),TI$(Z%);:Y%=
Y%+64:IFY%>703THENY%=Y%-683
1810 NEXT:PRINT@768,"";:RETURN
1820 IFPEEK(15360)=32THENPOKE15360,ZBELSEPOKE15360,32
1830 IFPEEK(293)<>73THENPRINT@3,"TRACK:";PEEK(&H37ED);"  SECTOR:
";PEEK(&H37EE);
1840 IFPEEK(293)=73THENPRINT@3,"TRACK:";INP(&HF1);"  SECTOR:";IN
P(&HF2);
1850 RETURN
1860 IFPEEK(293)=73THENZ=USR0(0)
1870 RETURN
1880 IFPEEK(293)=73THENZ=USR1(0)
1890 RETURN
1900 FORY%=0TO35:READTI$(Y%):NEXTY%:RETURN
1910 A$=INKEY$:IFA$=CHR$(13)THENRETURNELSE1910
1920 WT$=TIME$
1930 IFWT$=TIME$THEN1930ELSERETURN
1940 DATA DENTIST/DOCTOR,MEDICAL AIDS,PHARMACY,MEDICAL INS,MED M
ILEAGE,INTEREST/MTG,INTEREST/OTHER,FIXED TAXES,OTHER TAXES,DONAT
IONS,LOSS,EDUCAT'L EXP,UNION DUES ETC,CHILD CARE
1950 DATA RENT/MTG PRINC,UTILITIES,INSURANCE,GROCERIES,LOAN PRIN
CIPAL,HOME REPAIRS,SAVINGS,AUTO REPAIRS,GAS/OIL,VAC/ENTERTN'MT,C
LOTHES,FURNISHINGS,HOUSEHOLD ITEM,MISC EXP,OPEN ITEM
1960 DATA SALARY,MISC DEPOSITS,INTEREST,CHECKING/CASH,JOE FINANC
E,123 MAIN STREET,MILFORD NH
```

TRS-80® SWAT TABLE FOR: CHECKING

LINES	SWAT CODE	LENGTH
10 - 70	XH	544
80 - 180	DP	514
190 - 300	GK	419
310 - 390	EY	543
400 - 510	GH	395
520 - 630	ZY	516
640 - 750	GL	518
760 - 870	LM	563
880 - 960	DH	511
970 - 1050	RC	519
1060 - 1170	OT	525
1180 - 1290	FA	424
1300 - 1400	CP	508
1410 - 1520	WL	485
1530 - 1640	KV	433
1650 - 1760	HK	402
1770 - 1860	DD	515
1870 - 1960	KD	558

```
SS SS SS SS SS SS SS SS SS SS SS
SS                             SS
SS        ATARI  BASIC         SS
SS          'PFINIT'           SS
SS   AUTHOR:  LANCE MICKLUS    SS
SS    TRANSL: ALAN J. ZETT     SS
SS     COPYRIGHT (c) 1982      SS
SS SOFTSIDE PUBLICATIONS, INC SS
SS                             SS
SS SS SS SS SS SS SS SS SS SS SS
```

If you don't wish to type this program, it is also included in this month's SoftSide CV and DV.

```
10 GRAPHICS 0
20 CLR :DIM CA$(12),CD$(12),BD$(12),KD
$(12),DT$(20),A$(20)
30 CD$="CHECKING.DAT":CA$="CANCELCK.DA
T":KD$="CKFILE.DAT":BD$="BUDGET.DAT"
40 DT$="00/00/00 00:00 AM"
50 CLOSE #1:OPEN #1,4,0,"K"
100 ? "INITIALIZATION PROCEDURE":?
110 ? "THIS PROGRAM INITIALIZES A NEW
DATA":? "FILE DISK. IF THIS IS THE FIR
ST TIME"
120 ? "YOU'VE CREATED A DATA DISK, LOA
D A":? "BLANK FORMATTED DISK INTO ONE
OF YOUR"
130 ? "DRIVES; OTHERWISE, LOAD A BACKU
P COPY OF THE MOST RECENT DATA DISK."
140 POKE 752,1:? :?
150 ? CHR$(28);"WHAT DRIVE IS THE DISK
 IN? (1-4)":GET #1,A:A$=CHR$(A):IF A<4
9 OR A>52 THEN 150
160 ? :?
170 ? CHR$(28);"DRIVE ";CHR$(A);"? (Y/
N)":GET #1,A:IF A<>89 THEN ? CHR$(125)
;:GOTO 140
180 A$(2)=A$(1):A$(3)=":*.*":A$(1,2)="
D":TRAP 270:CLOSE #2:OPEN #2,6,0,A$:CL
OSE #2
190 ? CHR$(125):? "WRITING TO DISK - D
O NOT DISTURB!"
200 TRAP 220:A$(4)=CD$:OPEN #2,4,0,A$
210 TRAP 33333:CLOSE #2:GOTO 230
220 TRAP 33333:CLOSE #2:OPEN #2,8,0,A$
:? #2;DT$:? #2;"0,0":? #2;"EOF":CLOSE
#2
230 A$(4)=KD$:OPEN #2,8,0,A$:? #2;DT$:
? #2;"0":? #2;"EOF":CLOSE #2
240 A$(4)=BD$:OPEN #2,8,0,A$:? #2;DT$:
CLOSE #2
250 A$(4)=CA$:OPEN #2,8,0,A$:? #2;"999
9,9999,9999,9999":CLOSE #2
260 POKE 752,0:? :? "DATA FILE INITIAL
IZATION COMPLETE":CLOSE #1:END
270 TRAP 33333:? CHR$(253);CHR$(125):?
 "DRIVE DOES NOT EXIST!!":GOTO 140
```

ATARI® SWAT TABLE FOR: INITIALIZER

LINES	SWAT CODE	LENGTH
10 - 130	HA	512
140 - 210	CA	506
220 - 270	ZC	526

```
SS SS SS SS SS SS SS SS SS SS SS
SS                             SS
SS        ATARI  BASIC         SS
SS          'CHECKING'         SS
SS   AUTHOR:  LANCE MICKLUS    SS
SS    TRANSL: ALAN J. ZETT     SS
SS     COPYRIGHT (c) 1982      SS
SS SOFTSIDE PUBLICATIONS, INC SS
SS                             SS
SS SS SS SS SS SS SS SS SS SS SS
```

```
10 GRAPHICS 0:CLR :POKE 752,1:?
20 N0=0:N1=1:N2=2:N3=3:N4=4:N5=5:N6=6:
N7=7:N8=8:N9=9:N10=10:N11=11:N12=12:N1
3=13:N14=14:N15=15:N16=16:N17=17
30 N19=19:N23=23:N25=25:N32=32:N33=33:
N39=39:N48=48:N100=100:N128=128:N256=2
56:N462=462
40 N2030=2030:N2200=2200:N2310=2310:N2
570=2570:N3480=3480
50 SET=N0:TIME=20:UP=28:CLS=125:BELL=2
53:RESET=255:CRS=752:TEXT=755:KEY=764
60 CHECK=5240:IO=8060:TITLES=9030:WAIT
=9530:USING=9600:USING2=9620:CLEAR=333
33
70 SCR=PEEK(88)+PEEK(89)*N256:MAX1=999
99.99:MAX2=999999.99
100 DIM A$(N15),B$(N10),CC$(N15),DF$(N
17),DN$(N15),DT$(N17),EN$(N15),FL$(N15
),TF$(N17),CD$(N17),VA$(N10)
110 DIM N$(N25),S$(N25),C$(N25),TI$(N4
62),A(1200):TI$(N462)=" ":TI$(N1,N1)="
 ":TI$(N2)=TI$(N1)
120 A$=TI$:B$=A$:CC$=A$:DF$=A$:DN$=A$:
EN$=A$:FL$=A$:TF$=A$:VA$=A$:CD$=TI$
```

continued on page 42

WHY PAY HIGH RETAIL PRICES

FOR BUSINESS, EDUCATIONAL AND ENTERTAINMENT SOFTWARE, BOOKS AND PERIPHERALS FOR YOUR...

apple ATARI Commodore COMPUTER IBM TRS-80

ALL SOFTWARE AT LEAST 20% OFF! WE WELCOME INQUIRIES FOR HARD-TO-FIND ITEMS. WRITE!!

apple

LIST PRICE $39.95- OUR PRICE $31.95
MasterType — Aztec Adventure
Adam and Eve Paddles
Tubeway II — Zoom Grafix

LIST PRICE $34.95- OUR PRICE $27.95
Choplifter — Frogger — Serpentine
Knight of Diamonds — Transylvania
Cannonball Blitz — Crisis Mountain

LIST PRICE $29.95- OUR PRICE $23.95
Snack Attack — Sea Fox — Mars Cars — Autobahn
A2-PB1 Pinball — Castle Wolfenstein
Genetic Drift — Spy's Demise — Crazy Mazey

VisiCalc 3.3 or VisiFile.....$199.00
VisiTrend/VisiPlot.....$239.95
DB Master.....$179.95
Word Handler.....$159.00
Screen Writer II.....$103.95
Hayes Micromodem II w/term. program...$309.00
PFS,PFS:Graph, Super-Text 40-56-70.....$99. each
PFS:Report or Master Diagnostics Plus..$74.95
Arcade Machine.....$43.95
Home Accountant.....$59.95
Magic Window.....$79.95
Graphics Magician.....$47.95
Wizardry.....$39.95
Star Blazer.....$25.50
DOS Boss.....$19.00
Kids and the Apple (book).....$15.95

APPLE III USERS: WE SERVE YOU, TOO! WRITE!

ATARI

Sub Commander (cart.) or Deadline (D)..$39.95

LIST PRICE $39.95- OUR PRICE $31.95
Zork I,II, or III(D) — Starcross (D)
Compu-Math Decimals (D) or C-M Fractions (D)

LIST PRICE $34.95- OUR PRICE $27.95
Choplifter (D) — Frogger (D or C) — FaceMaker(D)
Mouskattack (D)

LIST PRICE $29.95- OUR PRICE $23.95
Crossfire (D,C) — Jawbreaker (D,C)
Preppie (D) — Labyrinth (D,C) — Genetic Drift D,C)
Compu-Read 3.0 (D) — C-M Fracs. or Dec. (C)

Golf Challenge (C) or Preppie (C).....$19.95
Compu-Read 3.0 (C).....$15.95

WRITE ABOUT PADDLES, JOYSTICKS, RAM MODULES

IBM

Call to Arms or Space Strike.....$23.95
Zork I,II or III or Starcross.....$31.95
Deadline.....$39.95
Face Maker.....$27.95
Adam and Eve Game Paddles.....$31.95
Snack Attack II.....$31.00
dBase II.....$525.00 Versaform.....$295.00
SuperCalc.....$225.00 Easy Writer II.$262.00
Home Accountant Plus..$99. Easy Filer.$299.00
Easy Speller..$135. Easy Planner...$189.00
VisiCalc256K, VisiDex or VisiLink.....$189.00
VisiTrend/Plot, VisiFile or Desktop Pln$230.00

TRS-80

WE CAN SUPPLY PRODUCTS FROM THE FOLLOWING MFRS: ACORN-ADVENTURE INTL.-AUTOMATED SIMULATIONS-AVALON HILL-BIG FIVE-BRODERBUND-CORNSOFT-EAGLE-EDUWARE EAST-HAYDEN-INFOCOM-MICROSOFT-ODESTA-QUALITY-STRATEGIC SIMULATIONS-SUBLOGIC-VISICORP-VOYAGER. **TELL US WHAT YOU WANT!**

VISA — No Credit Card Penalty — MasterCard

Send check, money order, Visa or Master Card # s with signature and expiration date. Personal checks will delay order. U.S. and Canada add $2.00 for shipping and handling.

Mail orders to:

If you have a favorite program that isn't listed send us your name address and program name and we will send you our special introductory price

SOFTWARE PLUS!
P.O. BOX 654
N. HAMPTON, NH 03862

Deluxe Personal Finance *continued*

```
180 CLOSE #N2:OPEN #N2,N4,N0,"K"
200 GOSUB 9360
240 FOR Y=N6 TO N14 STEP N8:POSITION N
0,Y:? "################################
#########";:NEXT Y
260 POSITION N3,N8:? "PERSONAL CHECK B
ALANCE PROGRAM OF:"
280 POSITION N10,N10:? N$
300 POSITION N10,N11:? S$
320 POSITION N10,N12:? C$
360 CK=N0:DF=N0:LP=N0
420 CC$="D:CANCELCK.DAT"
440 DN$="D:CHECKING.DAT"
600 POKE CRS,SET:POSITION N2,N16:? "EN
TER DATE & TIME?MM/DD/YY HH:MM AM"
610 POSITION N19,N16:INPUT CD$
620 IF CD$(N1,N6)="MM/DD/" THEN CD$="0
0/00/00 00:00 AM"
640 IF LEN(CD$)<>N17 THEN 600
660 POKE CRS,RESET:?
700 Q=N0:R=N0:S=N0:T=N0:TD=N0
720 EN$=DN$
730 TRAP IO:RESUME=720
750 CLOSE #N1:OPEN #N1,N4,N0,DN$
800 INPUT #N1,DT$
820 POSITION N2,N16:? "LAST UPDATE ON:
 ";DT$(N1,N8);" @ ";DT$(N10,N17)
840 TRAP CLEAR:INPUT #N1,I,B
860 IF I<N3 THEN CLOSE #N1:GOTO 1100
880 FOR N=N0 TO I-N1 STEP N3
900 GOSUB 9140
920 INPUT #N1,A,AZ,ZA:A(N)=A:A(N+N1)=A
Z:A(N+N2)=ZA:NEXT N:INPUT #N1,A$
940 IF A$="EOF" THEN 1020
960 POSITION N2,N0
980 ? "BAD DATA - PRESS RETURN TO CONT
INUE"
1000 GOSUB 9490
1020 CLOSE #N1
1040 T=N1
1100 POSITION N13,N23:? "PRESS ANY KEY
";:GET #N2,M
2030 ? CHR$(CLS);
2050 ? "AS OF ";DT$(N1,N8);" YOUR CURR
ENT":? "CHECKING BALANCE IS ";:DF=B:FW
=SET:GOSUB USING:? DF$:?
2060 REM LINE 2070 HAS CTRL CODES
2070 ? "qrrrrrrrrrrrrrrrrrrrrrrrrrrrre
":? "¦PERSONAL FINANCE MAIN MENU¦":? "ar
wrrrrrrrrrrrrrrrrrrrrrrrrrrrd"
2080 ? "¦0¦EXIT CHECKING PROGRAM  ¦":
IF I<N3 THEN 2100
2090 ? "¦1¦LIST OUTSTANDING CHECKS ¦"
2100 IF I<1198 THEN ? "¦2¦ADD NEW CHEC
KS TO FILE  ¦"
2110 IF I<N3 THEN I=N0:GOTO 2140
2120 ? "¦3¦FIX CHECKS WITH ERRORS  ¦"
2130 ? "¦4¦CANCEL BANK CHECKS  ¦"
2140 ? "¦5¦JUSTIFY BANK STATEMENT  ¦"
2150 ? "¦6¦ESTIMATE TOTAL BILLS  ¦"
2160 IF I>N2 THEN ? "¦7¦PRINT OUTSTAND
ING CHECKS¦":? "¦8¦OUTSTANDING CHECK S
TATUS¦"
2165 REM LINE 2170 HAS CTRL CODES
2170 ? "zrxrrrrrrrrrrrrrrrrrrrrrrrrrrc"
2190 ? :?
2200 POKE CRS,RESET:? CHR$(UP);
2220 ? "ENTER YOUR CHOICE":GET #N2,M
2230 M=M-N48:IF I<N3 AND M>N0 AND M<N5
 AND M<>N2 THEN GOTO N2200
2240 IF I<N3 AND M>N6 THEN GOTO N2200
2250 IF M<N0 OR M>N8 THEN GOTO N2200
2255 IF M=N2 AND I>1197 THEN GOTO N220
0
2260 IF M=N1 THEN GOSUB 5560:GOTO N203
0
2265 IF M=N7 THEN GOSUB 5550:GOTO N203
0
2270 ? CHR$(CLS):ON M+N1 GOTO 4500,N20
30,N2310,N2310,3150,3980,3820,N2030,43
20
2310 POKE CRS,SET:? :IF I>1197 AND M=N
2 THEN GOTO N2030
2315 TRAP 2315:? CHR$(UP);"QUIT OR TRA
NSACTION #";:INPUT VA$
2320 IF VA$(N1,N1)="Q" THEN GOTO N2030
2325 C=VAL(VA$):C=INT(C$N10)/N10:IF C<
N1 OR C>MAX1 THEN 2315
2330 IF M=N2 AND I<N3 THEN N=N0:I=N0:G
OTO 2950
2335 TRAP CLEAR
2340 FOR N=N0 TO I-N1 STEP N3
2350 IF A(N)=C AND M=N3 THEN GOTO N257
0
2360 IF A(N)=C AND M=N2 THEN 2440
2370 NEXT N
2380 IF M=N2 THEN 2950
2390 ? CHR$(CLS);"CHECK #";
2410 DF=C:FW=RESET:GOSUB USING2:? DF$;
" IS NOT OUTSTANDING"
2420 ?
2430 GOTO 2070
2440 GOSUB CHECK
2450 POSITION N4,N0:? CHR$(BELL);"ALRE
ADY EXISTS!":POSITION N5,N10:? "PRESS
 ANY KEY"
2460 FOR K=N0 TO N2:POKE TEXT,N2:GOSUB
 WAIT:POKE TEXT,N0:GOSUB WAIT:NEXT K:P
OKE TEXT,N2:GOSUB 9490
2480 ? CHR$(CLS):GOTO N2310
2520 POSITION N2,N10:? "FIX CHECK OR A
NY KEY";:GET #N2,AZ:IF AZ<>70 THEN ? C
HR$(CLS):GOTO N2310
2570 GOSUB CHECK
2580 POSITION N2,N0:FW=SET:DF=B:? "BAL
ANCE IS ";:GOSUB USING:? DF$
2585 REM LINE 2590 HAS CTRL CODES
2590 POSITION N23,N0:? "qrrrrrrrrrrrre
":POSITION N23,N1:? "¦ FIX CHECKS ¦":P
OSITION N23,N2:? "arwrrrrrrrrrrd"
2600 POSITION N23,N3:? "¦0¦DONE  ¦
":POSITION N23,N4:? "¦1¦FIX NUMBER¦":P
OSITION N23,N5:? "¦2¦FIX AMOUNT¦"
2620 REM LINE 2630 HAS CTRL CODES
2630 POSITION N23,N6:? "¦3¦FIX NAME  ¦
":POSITION N23,N7:? "¦4¦VOID  ¦":P
OSITION N23,N8:? "zrxrrrrrrrrrrc"
2650 POKE CRS,RESET
2660 POSITION N2,N10:? "ENTER YOUR CHO
ICE ";:GET #N2,A
2670 A=A-N48
2680 IF A<N0 OR A>N4 THEN 2660
2690 ? "[";CHR$(A+N48);"]":S=N1:ON A+N
1 GOTO 2700,2730,2790,2840,2890
2700 IF M<>2 THEN GOTO N2030
2710 ? CHR$(CLS):GOTO N2310
2730 POKE CRS,SET:?
2735 TRAP 2735:? CHR$(UP);"QUIT OR NEW
 TRANSACTION #";
2740 INPUT VA$:IF VA$(N1,N1)="Q" THEN
GOTO N2570
2750 C=VAL(VA$):C=INT(C$N10)/N10
2755 IF C<N1 OR C>MAX1 THEN 2735
2760 TRAP CLEAR:FOR K=N0 TO I-N1 STEP
N3
2770 IF A(K)=C THEN ? CHR$(BELL);"TRAN
SACTION #";:DF=C:FW=RESET:GOSUB USING2
:? DF$;" ALREADY EXISTS"
2775 IF A(K)=C THEN ? CHR$(UP);:GOTO 2
735
2780 NEXT K:A(N)=C:GOTO N2570
2790 POKE CRS,SET:?
2795 TRAP 2795:? CHR$(UP);"QUIT OR NEW
 AMOUNT";:INPUT VA$:IF VA$(N1,N1)="Q"
THEN GOTO N2570
2800 C=VAL(VA$):C=INT(C$N100)/N100:IF
C<-MAX1 OR C>MAX2 THEN 2795
2810 IF A(N+N1)>N0 THEN C=-C
2815 TRAP CLEAR:B=B-A(N+N1)-C
2820 A(N+N1)=-C:GOTO N2570
2840 GOSUB TITLES
2845 POKE CRS,SET:?
2850 TRAP 2850:? CHR$(UP);"QUIT OR NEW
 BUDGET #";:INPUT VA$:IF VA$(N1,N1)="Q
" THEN GOTO N2570
2860 J=INT(VAL(VA$))
2865 IF J<N0 OR J>31 THEN GOTO 2850
2870 TRAP CLEAR:A(N+N2)=J
2880 GOTO N2570
2890 B=B-A(N+N1)
2900 FOR K=N TO I-N4
2910 A(K)=A(K+N3)
2920 NEXT K
2930 I=I-N3:IF M<>2 THEN GOTO N2030
2940 ? CHR$(CLS):GOTO N2310
2950 A(N)=C
```

```
2955 POKE CRS,SET:?
2960 TRAP 2960:? CHR$(UP);"AMOUNT";
2965 INPUT VA$
2970 AZ=VAL(VA$):AZ=INT(AZ*N100)/N100:
A(N+N1)=AZ:IF AZ<-MAX1 OR AZ>MAX2 THEN
 2960
2975 POKE CRS,SET:?
2980 TRAP 2980:? CHR$(UP);"CHECK OR DE
POSIT";:INPUT B$
2985 IF B$(N1,N1)="C" THEN A(N+N1)=-A(
N+N1):GOTO 3020
2990 IF B$(N1,N1)<>"D" THEN 2980
3020 TRAP CLEAR:GOSUB TITLES
3025 POKE CRS,SET:?
3030 TRAP 3030:? CHR$(UP);"BUDGET #";
3035 INPUT VA$
3040 C=INT(VAL(VA$)):IF C<NO OR C>N32
THEN 3030
3045 A(N+N2)=C
3050 S=N1:I=I+N3:B=B+A(N+N1)
3055 TRAP CLEAR:GOSUB CHECK
3060 POSITION N2,NO:? "NEW BALANCE: ";
:DF=B:FW=SET:GOSUB USING:? DF$
3065 GOTO 2520
3150 IF T=N2 THEN GOTO N3480
3160 ? CHR$(CLS);
3170 EN$=CC$
3180 TRAP IO:RESUME=3170
3190 CLOSE #N3:OPEN #N3,N4,NO,CC$
3210 INPUT #N3,Z
3220 TRAP CLEAR:IF Z=9999 THEN 3360
3230 CLOSE #N3
3240 ? CHR$(CLS):? "NOTE: IF YOU CANCE
L CHECKS BEFORE":? "RUNNING BUDGET, CE
RTAIN DATA WILL BE"
3250 ? "UNAVAILABLE FOR FUTURE BUDGET
REPORTS"
3330 POKE CRS,RESET:? :?
3335 ? CHR$(UP);"STILL WISH TO CANCEL
CHECKS? (Y/N)":GET #N2,AZ
3340 IF AZ<>89 THEN GOTO N2030
3345 EN$=CC$
3350 TRAP IO:RESUME=3345
3360 CLOSE #N3:OPEN #N3,N8,NO,CC$
3370 CC=N1:POKE CRS,SET:? CHR$(CLS)
3380 TRAP 3380:POSITION N2,N1:? "CHARG
E CANCELLED CHECKS":? "TO WHAT MONTH (
1-12)";
3390 INPUT VA$:IF VA$(N1,N1)="Q" THEN
GOTO N2030
3400 Z=INT(VAL(VA$))
3410 IF Z<N1 OR Z>N12 THEN 3380
3420 TRAP CLEAR:? #N3,Z
3430 T=N2
3440 ? CHR$(CLS)
3480 POKE CRS,SET:?
3485 TRAP 3485:? CHR$(UP);"QUIT OR TRA
NSACTION #";:INPUT VA$
3490 IF VA$(N1,N1)="Q" THEN GOTO N2030
3510 C=VAL(VA$):C=INT(C*N10)/N10
3520 IF C<NO OR C>MAX1 THEN 3485
3530 TRAP CLEAR:FOR N=NO TO I-N1 STEP
N3
3540 IF A(N)=C THEN 3580
3550 NEXT N
3560 ? CHR$(CLS);CHR$(BELL);"TRANSACTI
ON #";:DF=C:FW=RESET:GOSUB USING2:? DF
$;" NOT OUTSTANDING"
3570 ? :GOTO N3480
3580 GOSUB CHECK
3585 POKE CRS,SET:?
3590 TRAP 3590:? CHR$(UP);"TYPE WRONG
TO SKIP, QUIT FOR MENU, OR THE NEXT TR
ANSACTION #";
3595 INPUT VA$:IF VA$(N1,N1)="W" THEN
? CHR$(CLS):GOTO N3480
3640 IF A(N+N1)<=NO THEN Q=Q-A(N+N1)
3650 IF A(N+N1)>NO THEN R=R+A(N+N1)
3670 ? :? " *WRITING DISK*";
3680 ? #N3;A(N);",";A(N+N1);",";A(N+N2
)
3690 FOR K=N TO I-N1
3700 A(K)=A(K+N3)
3710 NEXT K
3720 S=N1
3730 I=I-N3
3740 IF I>N2 THEN 3490
3750 ? CHR$(CLS);"NO MORE TRANSACTIONS
 IN FILE":?
3760 GOTO 2050
3820 ? CHR$(CLS)
3840 ? :?
3850 ? "ENTER THE COST OF YOUR BILLS T
O SEE   WHAT THE TOTAL IS AND HOW MUCH
 MONEY  YOU'LL HAVE LEFT OVER."
3870 D=NO
3880 POSITION N2,N8:POKE CRS,SET:?
3890 TRAP 3890:? CHR$(UP);"QUIT OR BIL
L AMOUNT";:INPUT VA$
3900 IF VA$(N1,N1)="Q" THEN GOTO N2030
3910 C=VAL(VA$):C=INT(C*N100)/N100:IF
C<-MAX1 OR C>MAX2 THEN 3890
3920 D=D+C
3930 TRAP CLEAR:? CHR$(CLS):?
3940 ? "YOUR TOTAL BILLS SO FAR: ";:DF
=D:FW=SET:GOSUB USING:? DF$:?
3945 ? :? "THAT LEAVES A BALANCE OF: "
;:DF=B-D:FW=SET:GOSUB USING:? DF$
3950 GOTO 3880
3980 ? CHR$(CLS)
3990 POKE CRS,SET:POSITION N2,N12
4000 TRAP 3990:? "QUIT OR BANK STATEME
NT BALANCE";
4010 INPUT VA$:IF VA$(N1,N1)="Q" THEN
GOTO N2030
4020 C=VAL(VA$):C=INT(C*N100)/N100:IF
C<-MAX1 OR C>MAX2 THEN 3990
4030 TRAP CLEAR:D=NO:POKE CRS,RESET
4040 IF I<N3 THEN 4080
4050 FOR N=N1 TO I-N1 STEP N3
4060 D=D+A(N)
4070 NEXT N
4080 ? CHR$(CLS)
4090 DF=Q:FW=N11:GOSUB USING:? DF$;" C
ANCELLED CHECKS"
4100 DF=R:GOSUB USING:? DF$;" CANCELLE
D DEPOSITS"
4110 ? "-----------":DF=R-Q:GOSUB USIN
G:? DF$;" TOTAL":? :?
4120 DF=C:GOSUB USING:? DF$;" BANK STA
TEMENT"
4130 DF=D:GOSUB USING:? DF$;" OUTSTAND
ING"
4140 ? "-----------":DF=C+D:GOSUB USIN
G:? DF$;" TOTAL":? :?
4150 DF=B:GOSUB USING:? DF$;" CURRENT
BALANCE":? :?
4180 E=ABS(B-C-D):IF E>N1 THEN 4240
4200 IF INT(E*1000)<>NO THEN 4240
4220 ? "PRESS ANY KEY FOR MENU";:GOSUB
 9490:GOTO N2030
4240 DF=ABS(B-C-D):GOSUB USING:? CHR$(
BELL);DF$;" *** ERROR ***":? :?
4250 POKE CRS,SET:?
4260 TRAP 4260:? CHR$(UP);"QUIT OR NEW
 BALANCE";:INPUT VA$:IF VA$(N1,N1)="Q"
 THEN GOTO N2030
4270 B=VAL(VA$):B=INT(B*N100)/N100:IF
B<-MAX1 OR B>MAX2 THEN 4260
4280 TRAP CLEAR:S=N1:GOTO N2030
4320 C=NO:FOR N=N1 TO I-N1 STEP N3:C=C
+A(N):NEXT N:POSITION N2,N7
4330 ? 1/3;" OUTSTANDING TRANSACTION";
CHR$(27+56*(1<>3)):?
4340 DF=C:FW=SET:GOSUB USING:? "OUTSTA
NDING CHECK TOTAL: ";DF$:?
4350 ? "CHECKING DATA ";:IF S THEN ? "
HAS BEEN ";:GOTO 4370
4360 ? "REMAINS UN";
4370 ? "MODIFIED":? :? "PRESS ANY KEY
FOR MENU";:GOSUB 9490:GOTO N2030
4500 ? CHR$(CLS):? "SURE YOU WANT TO?
(Y/N)":GET #N2,AZ:IF AZ<>89 THEN GOTO
N2030
4510 ? :? "WRITING TO DISK - DO NOT DI
STURB":TRAP 4550:IF T=N2 THEN ? #N3;"9
999,9999,9999,9999":T=NO
4520 IF  NOT S THEN 4600
4530 CLOSE #N1:OPEN #N1,N8,NO,DN$:? #N
1;CD$:? #N1;I;",";B:IF I<N3 THEN ? #N1
;"EOF":GOTO 4600
4540 FOR N=NO TO I-N1 STEP N3:? #N1;A(
N);",";A(N+N1);",";A(N+N2):NEXT N:? #N
1;"EOF":GOTO 4600
4550 ? CHR$(CLS);CHR$(BELL):? "*** DIS
K I/O ERROR ***":? :? "PRESS ESC TO EX
IT, ANY KEY TO RETRY":GET #N2,AZ
```

```
4560 IF AZ<>27 THEN 4510
4600 TRAP CLEAR:FOR X=N1 TO N4:CLOSE #
X:NEXT X:POKE CRS,SET:? :END
5240 POKE CRS,RESET:? CHR$(CLS)
5245 REM LINE 5250 HAS CTRL CODES
5250 ? :? "qrrrrrrrrrrrrrrrre":FOR Y=
N0 TO N4:? "¦                  ¦":NEXT
Y:? "zrrrrrrrrrrrrrrrrc"
5260 POSITION N4,N4:IF A(N+N1)>N0 THEN
 ? "DEPOSIT";:GOTO 5280
5270 ? "CHECK";
5280 ? " #";:DF=A(N):FW=RESET:GOSUB US
ING2:? DF$
5285 POSITION N4,N5:? TI$((A(N+N2)+N1)
*N14-N13,(A(N+N2)+N1)*N14);:POSITION N
4,N6:DF=ABS(A(N+N1))
5290 FW=SET:GOSUB USING:? DF$;:POSITIO
N N2,N10:RETURN
5550 LP=N1:CLOSE #N4:GOTO 5570
5560 LP=N0
5570 IF LP THEN TRAP 5550:OPEN #N4,N8,
N0,"P"
5580 IF I<N3 THEN RETURN
5590 TRAP CLEAR:X=N0
5600 FOR K=N0 TO 1-N1 STEP N3:IF X<>N0
 THEN 5650
5610 IF LP AND K<>N0 THEN 5640
5615 REM LINE 5620 HAS CTRL CODES
5620 ? CHR$(CLS);"qrrrrrrrwrrrrrrrrrrr
wrwrrrrrrrrrrrrre¦ENTRY #¦ENTRY VALUE
¦T¦ENTRY CATEGORY¦";
5625 REM LINE 5630 HAS CTRL CODES
5630 ? "arrrrrrrsrrrrrrrrrrrsrsrrrrrrr
rrrrrrrd";
5640 IF LP THEN ? #N4;" ":? #N4;" ":?
#N4;" ":? #N4;" "
5645 IF LP THEN ? #N4;"ENTRY #  ENTRY
VALUE  T  ENTRY CATEGORY":? #N4;"-----
--  -----------  -  --------------"
5650 AZ=A(K+N2)+N1:A$=TI$(AZ*N14-N13,A
Z*N14):B$=CHR$(67+(A(K+N1)>N0))
5660 ? "¦";:DF=A(K):FW=SET:GOSUB USING
2:? DF$;"¦";:DF=ABS(A(K+N1)):FW=N11:GO
SUB USING:? DF$;"¦";B$;"¦";A$;"¦";
5670 IF  NOT LP THEN 5700
5680 DF=A(K):FW=SET:GOSUB USING2:? #N4
;DF$;"  ";:DF=ABS(A(K+N1)):FW=N11:GOSU
B USING:? #N4;DF$;"  ";B$;"  ";A$
5700 A=N1:X=X+N1
5710 IF (X<N17 AND  NOT LP) OR (X<N48
AND LP) THEN 5810
5715 REM LINE 5720 HAS CTRL CODES
5720 IF  NOT LP THEN ? "zrrrrrrrxrrrrr
rrrrrrxrxrrrrrrrrrrrrrrc":GOTO 5760
5730 A=N0:? #N4;" ":? #N4;" ":? #N4;CD
$(1,8),"OUTSTANDING CHECK FILE";CHR$(N
12);:GOTO 5790
5760 POSITION N3,N23:? "PRESS ANY KEY
FOR MORE, ESC TO ABORT";:GET #N2,AZ:IF
 AZ=27 THEN RETURN
5790 X=N0
5810 NEXT K
5815 REM LINE 5820 HAS CTRL CODES
5820 ? "zrrrrrrrxrrrrrrrrrrrxrxrrrrrrr
rrrrrrrc":IF LP AND A=N0 THEN ? #4;" "
5830 IF LP AND A THEN ? #N4;" ":? #N4;
" ":? #N4;CD$(N1,N8),"OUTSTANDING CHEC
K FILE";CHR$(N12)
5840 POSITION N3,N23:? "END OF LIST, P
RESS ANY KEY FOR MENU";:GOSUB 9490
5850 RETURN
8060 POKE CRS,RESET:POSITION N2,N0:? "
LOAD DISK WITH FILE '";EN$;"'":POSITIO
N N8,N2
8080 ? "HIT RETURN TO CONTINUE"
8100 GOSUB 9490
8120 COLOR N32:FOR Y=N0 TO N2:PLOT N0,
Y:DRAWTO N39,Y:NEXT Y
8140 GOTO RESUME
9030 ? CHR$(CLS);:POKE CRS,RESET
9040 FOR Z=N0 TO N16:POSITION N2,Z:? Z
:POSITION N5,Z:? TI$((Z+N1)*N14-N13,(Z
+N1)*N14):NEXT Z
9050 FOR Z=N17 TO N32:POSITION 21,Z-N1
7:? Z:POSITION 24,Z-N17:? TI$((Z+N1)*N
14-N13,(Z+N1)*N14):NEXT Z
9060 POSITION N2,18:RETURN
9140 AZ=PEEK(SCR):AZ=AZ+N128*(AZ=N0)-N
128*(AZ<>N0):POKE SCR,AZ:RETURN
9360 RESTORE 10000
9380 FOR Y=N1 TO N33
9390 READ A$
9400 TI$(Y*N14-N13,Y*N14)=A$
9410 NEXT Y
9420 READ N$,S$,C$
9430 RETURN
9490 POKE KEY,RESET
9500 IF PEEK(KEY)=RESET THEN 9500
9510 POKE KEY,RESET:RETURN
9530 POKE TIME,N0
9540 IF PEEK(TIME)<N15 THEN 9540
9550 RETURN
9600 IF  NOT DF THEN DF$="0.00":GOTO 9
604
9602 UA=ABS(DF)+1.0E-03:DF$=STR$(UA):D
F$=DF$(N1,LEN(DF$)-N1)
9604 TF$="":IF DF<N0 THEN TF$="-"
9606 IF  NOT FW THEN TF$(LEN(TF$)+N1)=
"$":TF$(LEN(TF$)+N1)=DF$:DF$=TF$:RETUR
N
9608 IF DF>=N0 THEN TF$=" "
9610 TF$(N17)=" ":TF$(N2,N3)="$ ":TF$(
N4)=TF$(N3):TF$(FW-LEN(DF$(N2)))=DF$:D
F$=TF$:RETURN
9620 IF  NOT DF THEN DF$="0.0":GOTO 96
24
9622 UA=ABS(DF)+0.01:DF$=STR$(UA):DF$=
DF$(N1,LEN(DF$)-N1)
9624 IF FW THEN RETURN
9626 TF$=" ":TF$(N17)=" ":TF$(N2)=TF$(
N1):TF$(N7-LEN(DF$(N2)))=DF$:DF$=TF$:R
ETURN
10000 DATA DENTIST/DOCTOR,MEDICAL AIDS
,PHARMACY,MEDICAL INS,MED MILEAGE,INTE
REST/MTG,INTEREST/OTHER,FIXED TAXES
10010 DATA OTHER TAXES,DONATIONS,LOSS,
EDUCAT'L EXP,UNION DUES ETC,CHILD CARE
,RENT/MTG PRINC,UTILITIES,INSURANCE
10020 DATA GROCERIES,LOAN PRINCIPAL,HO
ME REPAIRS,SAVINGS,AUTO REPAIRS,GAS/OI
L,VAC/ENTERTN'MT,CLOTHES
10030 DATA FURNISHINGS,HOUSEHOLD ITEM,
MISC EXP,OPEN ITEM,SALARY,MISC DEPOSIT
S,INTEREST,CHECKING/CASH
10040 DATA JOE FINANCE,123 MAIN STREET
,MILFORD NH  03055
```

ATARI® SWAT TABLE FOR: CHECKING

LINES	SWAT CODE	LENGTH
10 - 50	AU	559
60 - 280	GT	501
300 - 720	RS	282
730 - 980	WS	295
1000 - 2110	BG	453
2120 - 2240	GY	393
2250 - 2340	GZ	351
2350 - 2480	VS	283
2520 - 2680	AL	493
2690 - 2780	CZ	379
2790 - 2870	IQ	327
2880 - 2970	LW	217
2975 - 3055	AR	273
3060 - 3250	PQ	311
3330 - 3420	TO	304
3430 - 3570	TE	273
3580 - 3720	HE	296
3730 - 3910	LT	348
3920 - 4040	AK	336
4050 - 4180	TB	367
4200 - 4360	QF	505
4370 - 4600	FY	519
5240 - 5580	KZ	392
5590 - 5660	GM	544
5670 - 5820	GV	514
5830 - 9060	AU	428
9140 - 9530	MB	168
9540 - 9626	OA	397
10000 - 10040	SJ	453

```
SS SS SS SS SS SS SS SS SS SS SS
SS                             SS
SS      APPLESOFT  BASIC       SS
SS        'INITIALIZER'        SS
SS   AUTHOR:  LANCE MICKLUS    SS
SS   TRANSL: KERRY SHETLINE    SS
SS     COPYRIGHT (c) 1982      SS
SS SOFTSIDE PUBLICATIONS, INC SS
SS                             SS
SS SS SS SS SS SS SS SS SS SS SS
```

If you don't wish to type this program, it is also included in this month's SoftSide CV and DV.

```
10  HOME : PRINT : PRINT "INITIALI
    ZATION PROCEDURE": PRINT : PRINT
    "THIS PROGRAM INITIALIZES NEW
     DATA": PRINT "FILES, 'CHECKI
    NG/DAT' AND 'CANCELCK/DAT'FOR
     THE 'CHECKING' PROGRAM, AND"
20  PRINT "'BUDGET/DAT' AND 'CKFIL
    E/DAT FOR THE": PRINT "'BUDGE
    T' PROGRAM. PLEASE INSERT AN"
    : PRINT "INITIALIZED DISK TO
    BE USED WITH YOUR": PRINT "PE
    RSONAL FINANCE PROGRAMS.": PRINT
    : PRINT
30  PRINT "PRESS 'RETURN' WHEN REA
    DY"
40  GET A$: IF  ASC (A$) <  > 13 THEN
     40
100  PRINT :D$ =  CHR$ (4): PRINT
     D$"OPEN CHECKING/DAT": PRINT
     D$"WRITE CHECKING/DAT": PRINT
     "01/01/83": PRINT 0: PRINT 0:
      PRINT D$"CLOSE"
110  PRINT D$"OPEN CANCELCK/DAT": PRINT
     D$"WRITE CANCELCK/DAT": PRINT
     0: PRINT D$"CLOSE"
120  PRINT D$"OPEN BUDGET/DAT": PRINT
     D$"WRITE BUDGET/DAT": PRINT "
     01/01/83": PRINT 0: PRINT D$"
     CLOSE"
130  PRINT D$"OPEN CKFILE/DAT": PRINT
     D$"WRITE CKFILE/DAT": PRINT "
     01/01/83": PRINT "EOF": PRINT
     D$"CLOSE"
```

APPLE™ SWAT TABLE FOR:
INITIALIZER

LINES	SWAT CODE	LENGTH
10 - 110	KT	503
120 - 130	FA	73

```
SS SS SS SS SS SS SS SS SS SS SS
SS                             SS
SS      APPLESOFT  BASIC       SS
SS          'CHECKING'         SS
SS   AUTHOR:  LANCE MICKLUS    SS
SS   TRANSL: KERRY SHETLINE    SS
SS     COPYRIGHT (c) 1982      SS
SS SOFTSIDE PUBLICATIONS, INC SS
SS                             SS
SS SS SS SS SS SS SS SS SS SS SS
```

```
10  DIM A(1200),TI$(35),MN$(11):D$
     =  CHR$ (4):B$ =  CHR$ (7):F
    F$ =  CHR$ (12):ESC$ =  CHR$
    (27):B8$ = "          ":OP$ = D
    $ + "OPEN ":RD$ = D$ + "READ
    ":WR$ = D$ + "WRITE ":CL$ = D
    $ + "CLOSE": DEF  FN H(X) =  SGN
    (X) *  INT ( ABS (X) * 100 +
    .01) / 100
20 Q = 0:R = Q:S = Q:T = Q
30 PS = 1:PL = 58
40  FOR X = 0 TO 35: READ TI$(X): NEXT
     X
50  DATA "DENTIST/DOCTOR","MEDICAL
     AIDS","PHARMACY","MEDICAL IN
    S","MED MILEAGE","INTEREST/MT
    G","INTEREST/OTHER","FIXED TA
    XES","OTHER TAXES","DONATIONS
    ","LOSS"
60  DATA "EDUCAT'L EXP","UNION DUE
    S,ETC","CHILD CARE","RENT/MTG
     PRINC","UTILITIES","INSURANC
    E","GROCERIES","LOAN PRINCIPA
    L","HOME REPAIRS","SAVINGS","
    AUTO REPAIRS"
70  DATA "GAS/OIL","VAC/ENTERTN'MT
    ","CLOTHES","FURNISHINGS","HO
    USEHOLD ITEM","MISC EXP","OPE
    N ITEM","SALARY","MISC DEPOSI
    TS","INTEREST","CHECKING/CASH
    "
80  DATA "JOE FINANCE","128 HENRY
    STREET","MILFORD, NH 03055"
90  FOR X = 0 TO 11: READ MN$(X): NEXT
     X
100  DATA "JANUARY","FEBRUARY","MA
    RCH","APRIL","MAY","JUNE","JU
    LY","AUGUST","SEPTEMBER","OCT
    OBER","NOVEMBER","DECEMBER"
110  FOR X = 768 TO 859: READ N: POKE
     X,N: NEXT X: POKE 1013,76: POKE
     1014,0: POKE 1015,3
120  DATA 72,32,177,0,104,166,118,
    224,255,208,3,76,11,227,201,7
    6,240,3,76,192,222,169,73,32,
    192,222,169,78,32,192,222,169
    ,69,32,192,222,169,132,32,192
    ,222,32,227,223,32,108,221,16
    9,128,133,51,32,111,253,142,2
    55,2,138,32,213
130  DATA 227,172,255,2,240,11,136
    ,185,0,2,41,127,145,158,152,2
    08,245,165,157,145,131,200,16
    5,158,145,131,200,165,159,145
    ,131,96
140 FD$ = "CHECKING/DAT":FC$ = "CA
    NCELCK/DAT": ONERR  GOTO 210
150  HOME : TEXT : VTAB 4: PRINT "
    CHECK DATA FOR:": PRINT : PRINT
    : FOR X = 33 TO 35: PRINT TI$
    (X): NEXT X
160  PRINT OP$;FD$: PRINT RD$;FD$:
     & LINE INPUT DT$: PRINT D$
170 OM =  VAL ( LEFT$ (DT$,2)):OD =
     VAL ( MID$ (DT$,4,2)):OY =  VAL
    ( RIGHT$ (DT$,2))
180  PRINT : PRINT : PRINT "LAST F
    ILE UPDATE: "DT$
190  PRINT RD$;FD$: INPUT I,B: PRINT
    D$: IF I < 3 THEN  PRINT CL$:
     GOTO 220
200  PRINT RD$;FD$: FOR N = 0 TO I
     - 1 STEP 3: INPUT A(N),A(N +
    1),A(N + 2): NEXT N: PRINT CL
    $: POKE 216,0: GOTO 220
210  POKE 216,0: PRINT : PRINT CL$
    : HOME : VTAB 7: PRINT "DOS E
    RROR " PEEK (222): PRINT "MAK
    E SURE CHECKING/DAT IS PROPER
    LY": PRINT "INITIALIZED.": END
220  POKE 216,0: VTAB 15: CALL  -
    958: PRINT "PLEASE ENTER THE
    DATE": PRINT "(MM/DD/YY): ";:
     & LINE INPUT CD$: IF  LEN (C
    D$) < 6 THEN 220
230 CM$ =  LEFT$ (CD$,2): IF  MID$
    (CD$,2) < "0" THEN CM$ = "0" +
     LEFT$ (CD$,1):CD$ = "0" + CD
    $
240 CX$ =  MID$ (CD$,4,2): IF  MID$
    (CD$,5) < "0" THEN CX$ = "0" +
     MID$ (CD$,4,1)
250 CD$ = CM$ + "/" + CX$ + "/" +
     RIGHT$ (CD$,2):CM =  VAL (CM
    $):CD =  VAL (CX$):CY =  VAL
    ( RIGHT$ (CD$,2))
260  IF CM = 2 THEN ML = 28 + ( INT
    (CY / 4) = CY / 4): GOTO 280
270 M = CM - (CM > 7):ML = 30 + M -
    INT (M / 2) * 2
```

```
280 IF CY < 80 OR CM < 1 OR CM >
    12 OR CD < 1 OR CD > ML THEN 220
290 IF CY < OY OR (CY = OY AND CM
     < OM) OR (CY = OY AND CM = O
    M AND CD < OD) THEN  PRINT : PRINT
    "THIS DATE IS EARLIER THAN TH
    E LAST.": PRINT "DO YOU WISH
    TO USE IT (Y/N)? ";: GET A$: IF
    A$ <  > "Y" THEN 220
300 HOME : PRINT : PRINT "AS OF "
    CD$" YOUR CURRENT":DF = B:FW =
    0: GOSUB 1370: PRINT "CHECKIN
    G BALANCE IS "DF$: PRINT
310 PRINT : PRINT "0) END SESSION
    ": IF I > 2 THEN  PRINT "1) L
    IST OUTSTANDING CHECK FILE"
320 IF I < 1998 THEN  PRINT "2) A
    DD NEW CHECK TO FILE"
330 IF I > 2 THEN  PRINT "3) FIX
    CHECKS WITH DATA ERRORS": PRINT
    "4) CANCEL CHECKS RECEIVED FR
    OM BANK"
340 PRINT "5) JUSTIFY THE ACCOUNT
     WITH THE BANK": PRINT "  ST
    ATEMENT": PRINT "6) ESTIMATE
    TOTAL BILLS DUE"
350 IF I > 2 THEN  PRINT "7) PRIN
    T OUTSTANDING CHECK FILE": PRINT
    "8) OUTSTANDING CHECK STATUS"
360 PRINT : PRINT : PRINT "ENTER
    YOUR CHOICE: ";
370 GET M$: IF M$ < "0" OR M$ > "
    8" THEN 370
380 M =  VAL (M$): IF I < 3 AND (M
     = 1 OR M = 3 OR M = 4 OR M >
    6) OR M = 2 AND I > 1997 THEN
    370
390 PRINT : HOME : ON M + 1 GOTO
    1110,1220,400,400,760,1020,99
    0,1210,1320
400 GOSUB 1460: IF EX THEN 300
410 IF M = 2 AND I < 3 THEN N = 0
    : GOTO 660
420 FOR N = 0 TO I - 1 STEP 3: IF
    A(N) = C AND M = 3 THEN 480
430 IF A(N) = C AND M = 2 THEN 47
    0
440 NEXT N
450 IF M = 2 THEN 660
460 PRINT : PRINT G$;G$"TRANSACTI
    ON #"C" IS NOT IN THE": PRINT
    "OUTSTANDING CHECK FILE.": GOSUB
    1440: GOTO 400
470 PRINT G$;G$: GOSUB 1180: PRINT
    "ALREADY EXISTS": GOSUB 1440:
     HOME : GOTO 400
480 GOSUB 1180:DF = B:FW = 0: GOSUB
    1370: PRINT "CURRENT BALANCE:
     "DF$: PRINT : PRINT "OPTIONS
    : ": PRINT : PRINT "0) DONE":
     PRINT "1) FIX TRANSACTION NU
    MBER": PRINT "2) FIX AMOUNT":
     PRINT "3) FIX ITEM NAME": PRINT
    "4) VOID"
490 VTAB 20: CALL  - 958: PRINT "
    ENTER YOUR CHOICE: ";
500 GET A$: IF A$ < "0" OR A$ > "
    4" THEN 500
510 A =  VAL (A$): IF  NOT A AND M
     = 2 THEN  HOME : GOTO 400
520 IF A = 0 THEN 300
530 IF A < 1 OR A > 4 THEN 490
540 HOME :S = 1: ON A GOTO 550,58
    0,600,640
550 GOSUB 1460: IF EX THEN 480
560 FOR K = 0 TO I - 1 STEP 3: IF
    A(K) = C THEN  PRINT G$;G$"TR
    ANSACTION #"C" ALREADY EXISTS
    .": GOSUB 1440: GOTO 480
570 NEXT K:A(N) = C: GOTO 480
580 VTAB 7: PRINT "ENTER 'Q' TO E
    XIT OR": PRINT "NEW AMOUNT: $
    ";: & LINE INPUT VA$: IF  LEFT$
    (VA$,1) = "Q" THEN 480
590 C =  FN H( VAL (VA$)) * ( SGN
    (A(N + 1)) + ( NOT A(N + 1)))
    :B =  FN H(B - A(N + 1) + C):
    A(N + 1) = C: GOTO 480
600 GOSUB 1450
610 VTAB 21: CALL  - 958: PRINT "
    ENTER 'Q' TO EXIT OR": PRINT
    "NEW BUDGET NUMBER: ";: & LIN
    E INPUT VA$: IF  LEFT$ (VA$,1
    ) = "Q" THEN 480
620 J =  INT ( VAL (VA$)): IF J <
    0 OR J > 32 THEN 610
630 A(N + 2) = J: GOTO 480
640 B =  FN H(B - A(N + 1)): FOR K
     = N TO I - 4:A(K) = A(K + 3)
    : NEXT K:I = I - 3: IF M = 2 THEN
     HOME : GOTO 400
650 GOTO 300
660 A(N) = C: VTAB 10: PRINT "ENTE
    R AMOUNT: $";: & LINE INPUT V
    A$:C =  FN H( VAL (VA$))
670 VTAB 11: CALL  - 958: PRINT "
    CHECK OR DEPOSIT (C/D): ";: &
    LINE INPUT B$:B$ =  LEFT$ (B$
    ,1): IF B$ = "C" THEN C =  -
    C: GOTO 690
680 IF B$ <  > "D" THEN 670
690 A(N + 1) = C: GOSUB 1450
700 VTAB 21: PRINT "ENTER BUDGET
    NUMBER: ";: & LINE INPUT VA$:
    J =  INT ( VAL (VA$)): IF J <
    0 OR J > 32 THEN 700
710 S = 1:I = I + 3:B =  FN H(B +
    C):A(N + 2) = J: GOSUB 1180
720 PRINT "PRESS 'F' TO FIX OR AN
    Y": PRINT "OTHER KEY TO CONTI
```

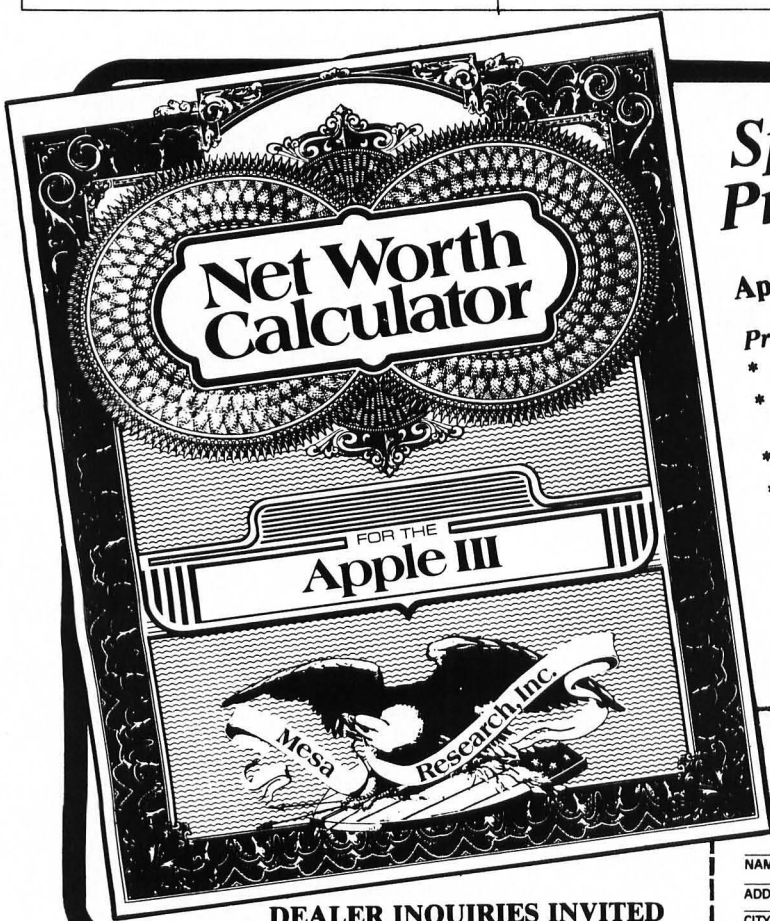
SoftTakes

```
    NUE: ";: GET B$: IF B$ = "F" THEN
    480
730 IF I > 1997 THEN  PRINT "OUTS
    TANDING CHECK FILE FULL.": GOSUB
    1440: GOTO 300
740 HOME : GOTO 400
750 GOTO 480
760 ONERR  GOTO 980
770 PRINT OP$;FC$: PRINT RD$;FC$:
     INPUT MN: PRINT CL$: IF  NOT
    MN THEN 840
780 IF T THEN 860
790 VTAB 7: PRINT "THIS FILE CONT
    AINS CANCELLED CHECKS": PRINT
    "FOR THE MONTH OF "MN$(MN - 1
    )".": PRINT : PRINT "IF YOU C
    ANCEL CHECKS BEFORE RUNNING":
     PRINT "'BUDGET', CERTAIN DAT
    A WILL BE": PRINT "UNAVAILABL
    E FOR FUTURE BUDGET REPORTS."
800 PRINT : PRINT : PRINT "DO YOU
     STILL WISH TO": PRINT "CANCE
    L CHECKS (Y/N)? ";
810 GET A$: IF A$ <  > "Y" AND A$
     <  > "N" THEN 810
820 IF A$ = "N" THEN 300
830 HOME
840 VTAB 4: CALL  - 958: PRINT "W
    HICH MONTH SHOULD THESE CANCE
    LLED": PRINT "CHECKS BE CHARG
    ED TO (1-12)? ";: & LINE INPUT
    VA$:MN =  INT ( VAL (VA$)): IF
    MN < 1 OR MN > 12 THEN 840
850 PRINT OP$;FC$: PRINT WR$;FC$:
     PRINT MN: PRINT 0: PRINT CL$
860 HOME : GOSUB 1460: IF EX THEN
     POKE 216,0: GOTO 300
870 FOR N = 0 TO I - 1 STEP 3: IF
    A(N) = C THEN 890
880 NEXT N: PRINT : PRINT G$;G$"T
    RANSACTION #"C" IS NOT IN THE
    ": PRINT "OUTSTANDING CHECKFI
    LE.": GOSUB 1440: HOME : GOTO
    860
890 GOSUB 1180: PRINT "PRESS 'W'
    IF WRONG TRANSACTION, OR": PRINT
    "ANY OTHER KEY TO CANCEL CHEC
    K.";: GET A$: IF A$ = "W" THEN
    860
900 PRINT : PRINT OP$;FC$: PRINT
    RD$;FC$: INPUT A: IF T = 0 THEN
    920
910 FOR X = 1 TO T: INPUT A,A,A: NEXT
    X
920 PRINT WR$;FC$: PRINT A(N): PRINT
    A(N + 1): PRINT A(N + 2): PRINT
    0: PRINT CL$
930 S = 1: IF A(N + 1) > 0 THEN R =
    R + A(N + 1): GOTO 950
940 Q = Q - A(N + 1)
950  FOR K = N TO I - 4:A(K) = A(K
     + 3): NEXT K:I = I - 3:T = T
     + 1
960  IF I < 3 THEN  POKE 216,0: GOTO
     300
970  GOTO 860
980  POKE 216,0: PRINT : PRINT CL$
     : HOME : VTAB 7: PRINT "DOS E
    RROR " PEEK (222): PRINT : PRINT
    "MAKE SURE CANCELCK/DAT IS PR
    OPERLY": PRINT INITIALIZED.":
    GOSUB9700:GOTO2060
990 D = 0: POKE 34,9: VTAB 4: PRINT
     "ENTER YOUR BILLS TO BE PAID
    TO SEE WHAT THE TOTAL IS AND
    HOW MUCH MONEY YOU'LL  HAVE L
    EFT OVER.": PRINT : PRINT "EN
    TER 'Q' TO STOP": PRINT
1000  PRINT "ENTER BILL: $";: & LI
     NE INPUT VA$: IF  LEFT$ (VA$,
    1) = "Q" THEN  POKE 34,0: GOSUB
     1440: GOTO 300
1010 D =  FN H( VAL (VA$) + D):VP =
     PEEK (37) + 1: VTAB 1:DF = D
    :FW = 0: GOSUB 1370: PRINT "Y
    OUR TOTAL BILLS ARE: "DF$:DF =
     FN H(B - D): GOSUB 1370: PRINT
    "REMAINING BALANCE WOULD BE:
    "DF$: VTAB VP: GOTO 1000
1020  VTAB 7: PRINT "ENTER 'Q' TO
    QUIT OR": PRINT "BALANCE FROM
     LAST": PRINT "BANK STATEMENT
    : $";: & LINE INPUT VA$: IF  LEFT$
    (VA$,1) = "Q" THEN 300
1030 C =  FN H( VAL (VA$)):D = 0: IF
     I < 3 THEN 1050
1040  FOR N = 1 TO I - 1 STEP 3:D =
     D + A(N): NEXT N
1050  HOME : VTAB 7: PRINT "BANK S
    TMT ";:DF = C:FW = 10: GOSUB
    1370: PRINT DF$" CAN CHKS ";:
    DF = Q: GOSUB 1370: PRINT DF$
    "OUTSTNDNG ";:DF = D: GOSUB 1
    370: PRINT DF$" CAN DEPS ";:D
    F = R: GOSUB 1370: PRINT DF$;
     TAB( 12)"---------"; TAB( 32
    )"---------";
1060 DF =  FN H(C + D): GOSUB 1370
    : PRINT  TAB( 11);DF$;:DF =  FN
    H(R - Q): GOSUB 1370: PRINT  TAB(
    31);DF$:DF = B:FW = 0: GOSUB
    1370: PRINT "CURRENT BALANCE:
     "DF$
1070 E =  ABS ( FN H(B - C - D)): IF
     NOT E THEN  GOSUB 1440: GOTO
    300
1080  VTAB 1:DF = E: GOSUB 1370: PRINT
    G$;G$"### ERROR ### "DF$
1090  VTAB 16: PRINT "ENTER 'Q' TO
     EXIT WITHOUT CHANGING": PRINT
    "BALANCE, OR NEW BALANCE: $";
    : & LINE INPUT VA$: IF VA$ =
    "" OR  LEFT$ (VA$,1) = "Q" THEN
    300
1100 S = 1:B =  FN H( VAL (VA$)): GOTO
    300
1110  IF  NOT S THEN  HOME : END
1120  HOME : VTAB 7: PRINT "WRITIN
    G TO DISK.": PRINT : INVERSE
    : PRINT "DO NOT DISTURB!": NORMAL
    : ONERR  GOTO 1160
1130  PRINT OP$;FD$: PRINT WR$;FD$
    : PRINT CD$: PRINT I: PRINT B
    : IF I < 3 THEN 1150
1140  FOR N = 0 TO I - 1 STEP 3: PRINT
    A(N): PRINT A(N + 1): PRINT A
    (N + 2): NEXT N
1150  POKE 216,0: PRINT CL$: HOME
    : END
1160  POKE 216,0: PRINT CL$: HOME
    : VTAB 7: FLASH : PRINT G$;G$
    "DISK I/O ERROR": NORMAL : PRINT
    : PRINT "PRESS ESC TO EXIT OR
     ANY": PRINT "OTHER KEY TO RE
    -TRY FILE SAVE.";: GET A$: IF
    A$ <  > ESC$ THEN 1110
1170  HOME : END
1180  HOME : INVERSE : PRINT  SPC(
    25): PRINT : FOR X = 1 TO 6: PRINT
    " ";: HTAB 25: PRINT " ": NEXT
    X: NORMAL : VTAB 3: HTAB 3: IF
    A(N + 1) > 0 THEN  PRINT "DEP
    OSIT";: GOTO 1200
1190  PRINT "CHECK";
1200  PRINT " #"A(N): HTAB 3: PRINT
    TI$(A(N + 2)):DF =  ABS (A(N +
    1)):FW = 0: GOSUB 1370: HTAB
    3: PRINT DF$: INVERSE : VTAB
    8: PRINT  SPC( 25): PRINT : NORMAL
    : PRINT : RETURN
1210  PRINT D$"PR#"PS:GA = PL:LP =
    1: GOTO 1230
1220 GA = 18:LP = 0
1230 AP = 0
1240 LC = 2: HOME : PRINT "TRANS.
    #  BUDGET" TAB( 26)"AMOUNT": PRINT
    "-------- --------------- ---
    ----------"
1250 DF =  ABS (A(AP + 1)):FW = 11
    : GOSUB 1370:TA = A(AP):TA$ =
     STR$ ( SGN (TA) *  INT ( ABS
    (TA))):TD$ = "": IF TA <  > INT
    (TA) THEN TD$ =  STR$ ( INT (
    ( ABS (TA) -  INT ( ABS (TA))
     + .001) * 10) / 10)
1260  PRINT  SPC( 6 -  LEN (TA$));
```

```
    TA$;TD$; TAB( 10);TI$(A(AP +
    2)); TAB( 25);DF$" ";: IF A(A
    P + 1) < 0 THEN  PRINT "CHK":
     GOTO 1280
1270  PRINT "DEP"
1280  IF AP + 3 = I THEN  PRINT FF
    $: PRINT D$"PR#0": GOSUB 1440
    : GOTO 300
1290 AP = AP + 3:LC = LC + 1: IF G
    A <  > LC THEN 1250
1300  IF LP THEN  PRINT FF$;: GOTO
    1240
1310  GOSUB 1440: GOTO 1240
1320 C = 0: FOR N = 0 TO I - 1 STEP
    3:C =  FN H(C + A(N + 1)): NEXT
    N: VTAB 7: PRINT I / 3" OUTST
    ANDING TRANSACTION" CHR$ (83 *
    (I <  > 3))".": PRINT :DF = C
    :FW = 0: GOSUB 1370: PRINT "O
    UTSTANDING CHECK TOTAL: "DF$
1330  PRINT : PRINT "CHECKING DATA
     ";: IF S THEN  PRINT "HAS BE
    EN ";: GOTO 1350
1340  PRINT "REMAINS UN";
1350  PRINT "MODIFIED": GOSUB 1440
    : GOTO 300
1360  STOP
1370  IF  NOT DF THEN DF$ = "0.00"
    : GOTO 1390
1380 UA =  ABS (DF) + .001:DF$ =  STR$
    (UA):DF$ =  LEFT$ (DF$, LEN (
    DF$) - 1)
1390 SN$ = "": IF DF < 0 THEN SN$ =
    "-"
1400  IF  NOT FW THEN DF$ = SN$ +
    "$" + DF$: RETURN
1410  IF DF >  = 0 THEN SN$ = " "
1420 BL = FW -  LEN (DF$) - 1: IF
    BL < 1 THEN DF$ = "%" + SN$ +
    "$" + DF$: RETURN
1430 DF$ =  LEFT$ (B8$,BL) + DF$:D
    F$ = SN$ + "$" +  MID$ (DF$,2
    ): RETURN
1440  VTAB 23: INVERSE : PRINT "PR
    ESS ANY KEY TO CONTINUE";: NORMAL
    : GET A$: PRINT : RETURN
1450  HOME : PRINT : FOR X = 0 TO
    15: PRINT  SPC( X < 10);X") "
    TI$(X): NEXT X: VTAB 2: FOR X
     = 16 TO 32: HTAB 21: PRINT X
    ") "TI$(X): NEXT X: PRINT : PRINT
    : RETURN
1460 EX = 0: VTAB 7: CALL  - 958: PRINT
    "ENTER 'Q' TO QUIT OR": PRINT
    "TRANSACTION NUMBER: ";: & LI
    NE INPUT VA$: IF  LEFT$ (VA$,
    1) = "Q" THEN EX = 1: RETURN
1470 C =  VAL (VA$):C =  INT (C *
    10 + .01) / 10: IF C < 1 OR C
     > 99999.9 THEN 1460
1480  VTAB 8: HTAB 21: PRINT C;: CALL
     - 958: PRINT : PRINT : RETURN
```

APPLE™ SWAT TABLE FOR:
CHECKING

LINES	SWAT CODE	LENGTH
10 - 60	SI	547
70 - 120	KM	614
130 - 210	OU	546
220 - 290	KV	510
300 - 380	FX	541
390 - 500	ZP	516
510 - 620	LJ	475
630 - 730	HO	502
740 - 840	FD	510
850 - 960	MZ	485
970 - 1020	KK	572
1030 - 1090	VY	513
1100 - 1200	UN	514
1210 - 1320	TC	592
1330 - 1440	VQ	387
1450 - 1480	CT	252



DEALER INQUIRIES INVITED

Special Pre-release Price

(VALID UNTIL JANUARY 1, 1983)

Apple II - $50.00 Apple III - $75.00

Prepares and Prints...

* *A personal financial statement.*
* *All five schedules - with itemized & non-itemized assets and liabilities.*
* *Posts from schedules to financial statement.*
* *Shows totals by category to be transferred.*
* *All schedules are open-ended with open code. (Personal backups possible).*
* *All reports are well formatted with pagination, headings and columnar data.*

Release Date: December 1, 1982 for:

Apple II 48 K system operating with DOS 3.3
Apple III 96 K system operating with SOS 1.1 & Business Basic

MESA RESEARCH, INC. • Rt. 1, Box 1456A • Waco, Tx. 76710
Phone 817/848-5272

Please send me _____ copies of Net Worth Calculator, Apple II/Apple III (scratch one).
Enclosed is $_____ which includes tax and $2.00 each for shipping & handling charges.

NAME _____

ADDRESS _____ A-1182

CITY _____ STATE _____ ZIP _____

GENERAL INFORMATION

Concerning SoftSide line listings, SWAT & Magnetic Media

Follow these procedures unless otherwise instructed by the documentation in the magazine. Back issues may differ in some details.

SWAT TABLES

At the conclusion of each line listing of a ***SoftSide*** program, we include a *SWAT* (*Strategic Weapon Against Typos*) Table. *SWAT* was published in issue #30 of ***SoftSide*** and is available as a free reprint. Please send a self-addressed, stamped envelope to ***SoftSide*** Publications, Inc., Dept. *SWAT*, 6 South Street, Milford, NH 03055.

APPLE™

Disks are in 16-sector format, created under DOS 3.3. To use, just boot the disk. A cover/menu program will run automatically.

Tapes LOAD in the normal manner. Advance the tape to the beginning of the lead-in tone; stop the tape; insert the plug into the EAR jack; type LOAD; start the tape; and press RETURN. Side two of the tape is a duplicate of side one, unless one or more Integer BASIC programs are included, in which case side two contains the Integer programs.

ATARI®

Line Listings use the following conventions in representing unprintable characters, unless otherwise noted:

Characters (including blank spaces) which are underlined should be typed in inverse video.

When graphics or control characters are to be included in a string (between quotation marks), it will be noted in a nearby REMark. In such cases, graphics characters are represented by the corresponding lower-case letter, and control characters are represented by the corresponding unshifted key symbol. For example: The lower-case letter s represents a graphics cross, entered by holding down the CTRL key and then pressing the S key. The symbol = represents a control-down-arrow, entered by first pressing and releasing the ESC key, then holding down the CTRL key and pressing the = key. (See Appendix F, and the back cover, of the *ATARI® BASIC Reference Manual.)*

The one exception to the above practice is that a clear-screen character (ESC CTRL-↰) is represented in listings by a right-hand brace, which looks like this: }

A shifted = is represented in the listings by a vertical line with a small gap in it: ¦

SWAT — Before appending *SWAT* to a program in memory, the program to be SWATed must first be LISTed to disk or cassette (using LIST "D:FILENAME" for disk or LIST "C:" for tape). Next, turn the computer off, then on again, to clear the system and ENTER the program back into memory (using ENTER "D:filename" for disk or ENTER "C:" for tape). Because of the unique method in which *ATARI® BASIC* stores variables in a program, the variable table must always be in the same order to produce accurate *SWAT* codes. LISTing and ENTERing the program is the only known way to rebuild the variable table in a specific order so that *SWAT* codes can match.

Disks do not contain DOS.SYS files, and are therefore not bootable by themselves. First boot a disk which contains any version of DOS, then insert the ***SoftSide*** disk and RUN "D:COVER" (*Adventure of the Month* — RUN "D:INTRO").

Tapes CLOAD in the normal manner. If you have difficulty, try this procedure:
(1) Type POKE 54018,54 and press RETURN.
(2) Turn up the volume on your TV.
(3) Type CLOAD and press RETURN once.
(4) Press the PLAY button and listen.
(5) When you hear a steady lead-in tone, press RETURN again.
Side two of the tape is a duplicate of side one.

IBM® PC

DV is available by subscription or individual order. There is no CV at this time.

TRS-80®

Disks are available in Model I or Model III format. They contain the DOS PLUS operating system, and a cover program which automatically runs upon booting. Back issues prior to May, 1982, are available only in Model I format, and may be converted using the TRSDOS CONVERT utility on a two-drive Model III. Older back issues (with Model I TRSDOS) require you to enter BASIC and then type RUN "COVER".

Tapes CLOAD in the normal manner on Model I's, and at low speed (500 baud) on Model III's. The first program is a cover/menu program. Side two of the tape is a duplicate of side one.

NOTES ABOUT MAGNETIC MEDIA

SoftSide disks and tapes are duplicated by reliable, professional duplication services; bad copies are very rare. However, the trip through the mail occasionally wreaks havoc with sensitive magnetic media. If, after a reasonable number of tries and a careful check and cleaning of your equipment, you are not able to load a program from a tape or disk, please return it to us with an exact description of the problem. If we cannot duplicate the problem on our systems, we will advise you when we send the replacement copy.

We use no copy-protection on our media. We urge you to make a backup copy of every disk or tape as soon as you receive it (and at the same time resist the urge to give copies to friends). Our replacement policy does not extend beyond 30 days.

PC/SIDE page 50

TRS-80®/SIDE page 57

ATARI®/SIDE page 76

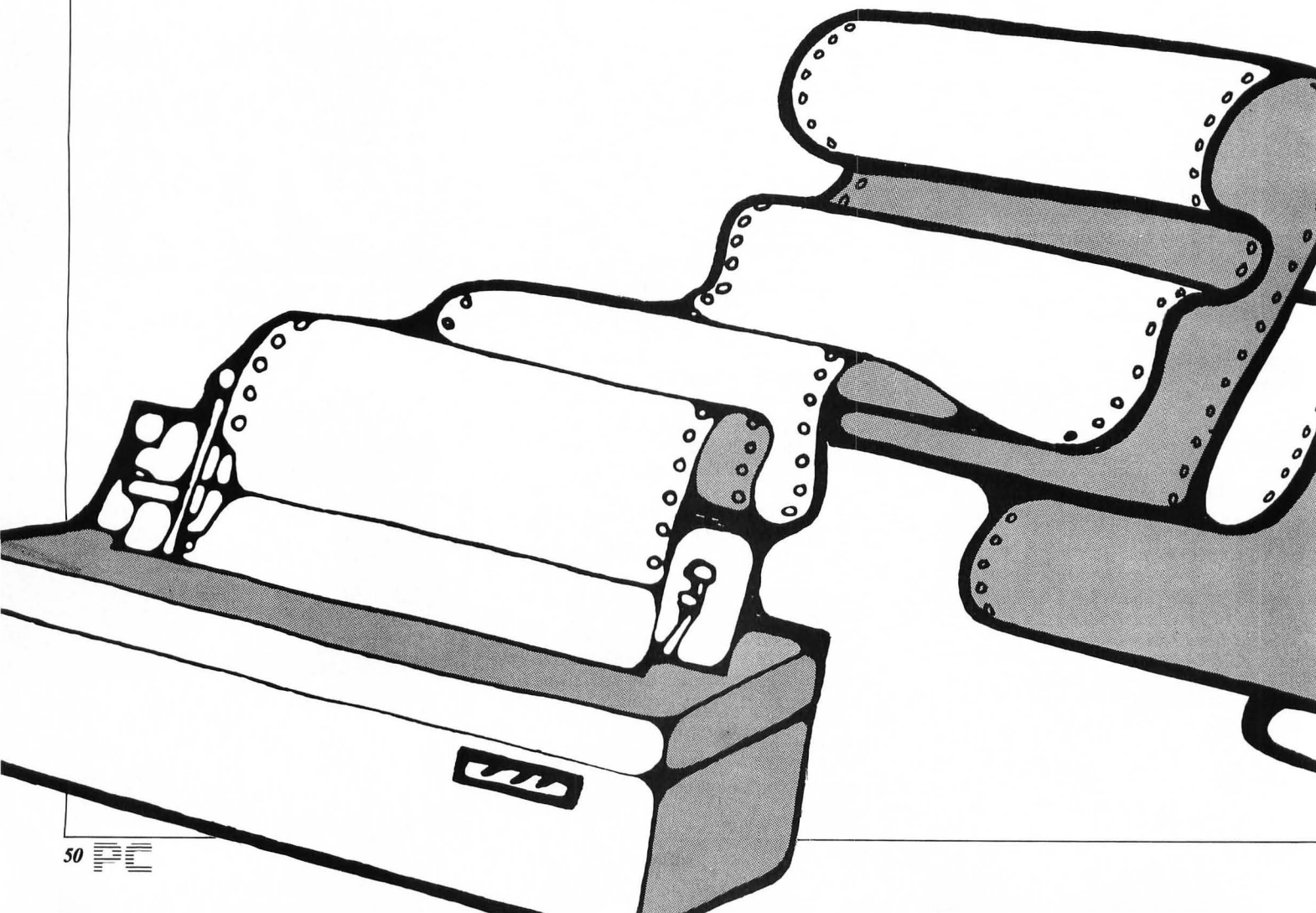
APPLE™/SIDE page 94

***Poster Maker* is a large-character printing program for an IBM® PC with 16K RAM and a printer.**

Poster Maker

by Fred J. Condo

Banner-making programs have probably been around as long as computers, but the banners have always run *along* the paper rather than *across* it. *Poster Maker* makes posters with words that run the same way ordinary printing would. You can input a whole phrase, and it will print it automatically, one word centered on each line. If you have an Epson MX-80 (or IBM 80-column) printer, choose the size of character you want by specifying the number of banner-sized characters per line. If any word won't fit, the program puts in a hyphen (but not necessarily in the grammatically correct position). If you remember the program starts a new line wherever you type a space, you can insert the hyphen in a long word where you want to divide it. To top it off, you can specify



whether each large character should be made up of the small characters it represents, or of another character or even a string of characters.

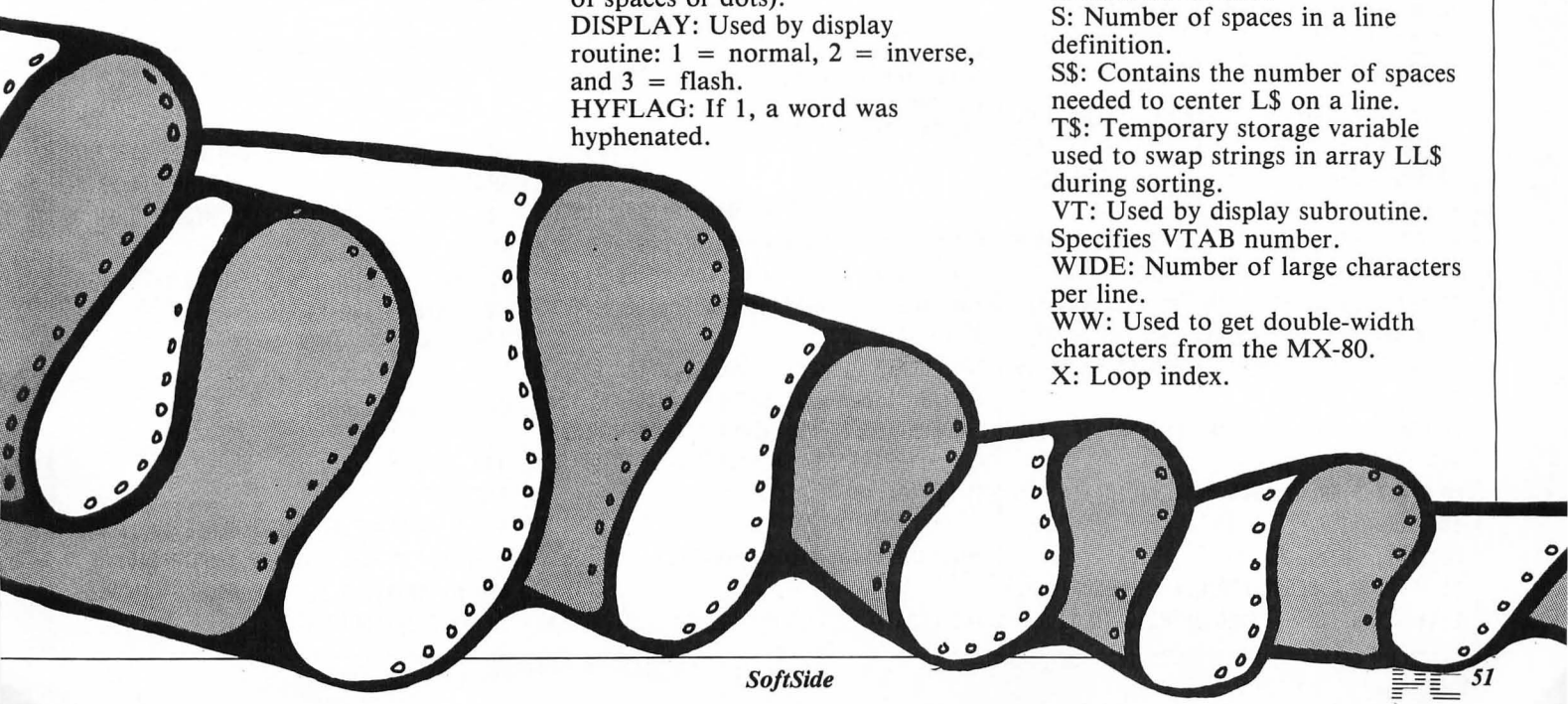
The program provides all the upper-case letters plus many of the punctuation characters. The DATA statements contain the character definitions, beginning with line 10000. To add or modify character definitions, use this data organization: First, designate the character being defined. If in doubt, put this character in quotation marks. Next, create 13 line definitions. You must have exactly 13, because the characters are composed of a 12 by 13 matrix. Each line definition is READ by the program (line 119) in number pairs, so each line definition must contain an even number of numeric values. Each of the line definitions must end with a pair of -1s, since they act as the end-of-line flag. (A blank line is designated by the pair of -1s alone.)

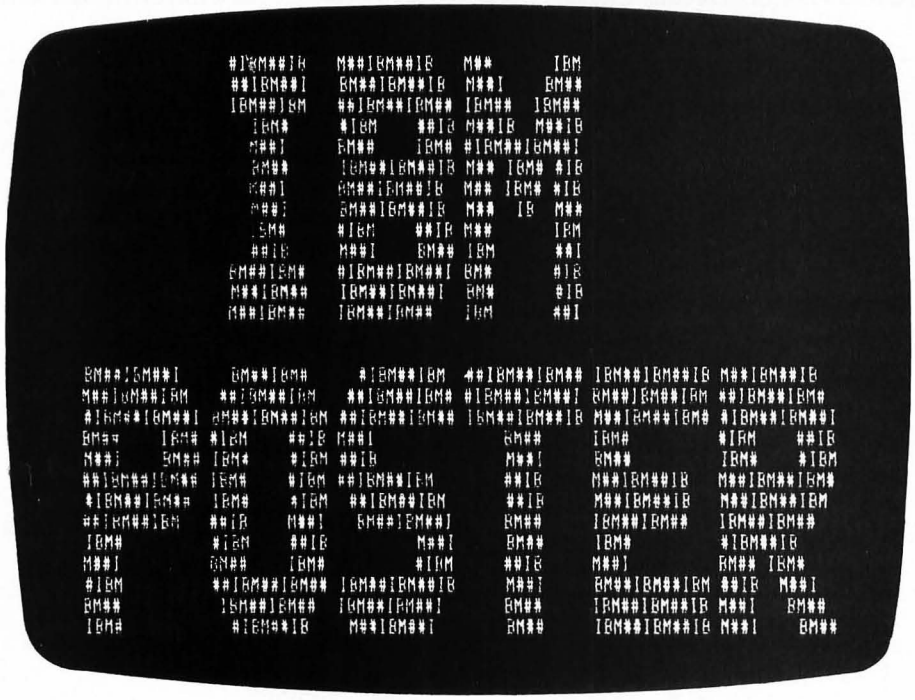
Each line definition consists of up to three pairs of numbers (plus the terminal -1s). The first number of a pair is always a number of spaces to be printed; the second is a number of dots. Together, these spaces and dots specify the printing pattern for a particular line of a character. The total number of dots plus spaces in each line must be no more than 12. The program does not check for faulty character definitions, so errors will show up either as error messages or as scrambled lettering on the posters. If you add characters, you will have to increase the subscripts in the DIM statement in line 30 from 55 to the actual number of characters you have defined.

If you don't have an Epson printer, always choose a width of six large characters. If your printer can handle alternate densities, then you'll need to make appropriate modifications of the character codes printed in lines 375-430.

Variables

A: Loop index in sorting routine.
A$: Allows display subroutine to display centered lines on the monitor screen. Also used for Y/N replay at the end of printing.
B: Loop index in sorting routine.
C: The number of dots (characters) in a line definition.
CFLAG: If 1, user has specified a character to make up the large ones.
CH$: Contains a character specified by the user to make up the larger characters. If null, characters are made up of their small analogs.
CP%: Pointer into the internal message.
CS$: Used in making the internal message.
DESC%(x,y,z): Mnemonic for "DESCription;" contains the character definitions. X subscript is the character; y is the line number; and z is the item (number of spaces or dots).
DISPLAY: Used by display routine: 1 = normal, 2 = inverse, and 3 = flash.
HYFLAG: If 1, a word was hyphenated.
I: Item number during DATA READing; character number in P1$ during printing; also a general loop index.
II: Item number during printing.
J, KK: Loop indices.
KK$: Used in the input-simulation subroutine to accept each character as it is typed at the keyboard.
L: Saves LEN(L$).
L$: Contains each word of the input string as it is cut apart. (A word consists of the characters between spaces.)
L$(x): Contains the characters defined in the DATA. X is analogous to the x subscript in the DESC% array.
LL$(x): Same as L$(x), but sorted into ASCII order. Used to present the available characters.
LTR: Number of characters defined, plus 1.
M$: Used to save the results of MID$ function calls.
OL$: Mnemonic for "Old L$". Contains the previous word printed to determine if a hyphen was used. If so, centering of the next line is suppressed.
P$: String input by user to be printed.
P1$: Same as P$, but with undefined characters removed. This is the string which is cut up and put, piece by piece, into L$.
QQ: Number of characters defined. Used to sort and display LL$(x).
R$: String returned by the input-simulation routine.
S: Number of spaces in a line definition.
S$: Contains the number of spaces needed to center L$ on a line.
T$: Temporary storage variable used to swap strings in array LL$ during sorting.
VT: Used by display subroutine. Specifies VTAB number.
WIDE: Number of large characters per line.
WW: Used to get double-width characters from the MX-80.
X: Loop index.





```
SS SS SS SS SS SS SS SS SS SS SS
SS                             SS
SS          PC BASIC           SS
SS       'Poster Maker'        SS
SS     Author: Fred Condo      SS
SS     Copyright (c) 1982      SS
SS SoftSide Publications, Inc  SS
SS                             SS
SS SS SS SS SS SS SS SS SS SS SS
```

If you don't wish to type this program, it is also included in this month's SoftSide DV.

Initialize screen, printer, lower-to-upper case conversion function FNU$, and data arrays. The CHR$(18);CHR$(20) clears out the special character mode of the Epson MX-80/IBM printer.

```
10 HEREAGAIN=0:KEY OFF:TROFF:SCREEN 0,0,
0:WIDTH 40:LOCATE,,0:CLS:OPEN"LPT1:" FOR
 OUTPUT AS #1:WIDTH #1,132:PRINT #1,CHR$
(18);CHR$(20):VT=1:DISPLAY=2:A$="
POSTER MAKER        ":GOSUB 760:VT=2:DISP
LAY=1:A$="BY F.J. CONDO":GOSUB 760
30 DEF FNU$(A$)=CHR$(ASC(A$)+32*(A$>="a"
 AND A$<="z")):VT=12:A$="READING DATA":D
ISPLAY=3:GOSUB 760:DIM DESC%(55,13,8),L$
(55),LL$(55):LTR=1
70 READ L$(LTR):LL$(LTR)=L$(LTR):IF L$(L
TR)="XXX" GOTO 160
80 VT=13:DISPLAY=1:A$="<"+L$(LTR)+">":GO
SUB 760:L=1
100 I=1
110 READ S,C:DESC%(LTR,L,I)=S:DESC%(LTR,
L,I+1)=C:IF S<>-1 THEN I=I+2:GOTO 110
```

```
140 L=L+1:IF L<=13 GOTO 100
150 LTR=LTR+1:GOTO 70
```

Sort the data in LL$(B) in ascending ASCII order for later display of available characters.

```
160 LOCATE 12,1:PRINT SPACE$(120);:VT=12
:DISPL=3:A$="SORTING DATA":GOSUB 760:QQ=
LTR-1:FOR A=1 TO QQ-1:FOR B=1 TO QQ-A:IF
 LL$(B)>LL$(B+1) THEN T$=LL$(B):LL$(B)=L
L$(B+1):LL$(B+1)=T$
190 NEXT B,A
```

Input the maximum number of large characters allowed per line. For 80 columns, this should be six. Display this number at the top of the screen, and display the available characters.

```
200 LOCATE 12,1:PRINT SPACE$(120);:LOCAT
E 12,1:INPUT"How many large characters w
ide do you  want each line to be (max.
10)";WIDE:WIDE=INT(WIDE):IF WIDE<1 OR WI
DE>10 GOTO 200
220 LOCATE 2,1:PRINT SPC(40):VT=2:DISPLA
Y=2:A$="Lines will be"+STR$(WIDE)+" long
":GOSUB 760:IF NOT HEREAGAIN THEN HEREAG
AIN=-1:PRINT"USE ONLY":FOR A=1 TO QQ:PRI
NT LL$(A);:NEXT A
```

Input the phrase to be printed.

```
260 LOCATE 12,1:PRINT SPACE$(120);:LOCAT
E 12,1:LINE INPUT"Phrase: ";P$:IF P$=""
 THEN SCREEN 0,0,0:KEY ON:WIDTH 80:CLS:CL
OSE #1:END
```

Convert to upper case, then remove any undefined characters. After this, P1$ contains the phrase actually to be printed.

```
290 P1$="":FOR I=1 TO LEN(P$):M$=FNU$(MI
D$(P$,I,1)):MID$(P$,I,1)=M$:FOR J=1 TO L
TR-1:IF M$<>L$(J) THEN NEXT J
330 IF J<LTR THEN P1$=P1$+M$
340 NEXT I:IF P1$="" THEN PRINT SPACE$(1
20);:LOCATE 12,1:PRINT"No data available
 for your input string.Please try again,
 using only the charac-ters listed above
.":LOCATE,,1:KK$=INPUT$(1):GOTO 260
```

Input the character or internal phrase from which to construct the large characters. All blank spaces are removed from this.

```
360 CFLAG=0:PRINT"Internal message to pr
int or simply hit <RETURN> to have the b
ig letters made upof their small counter
parts: ";:LINE INPUT"";CH$
370 IF CH$="" GOTO 375
372 CFLAG=-1:CS$="":FOR RE=1 TO LEN(CH$)
:M$=MID$(CH$,RE,1):IF M$<>" " THEN CS$=C
S$+M$
373 NEXT RE:CH$=CS$
```

Generate the proper control codes for the Epson MX-80/IBM printer. CHR$(15) turns on the compressed font; setting WW to 14 doubles the width of whatever font is currently active.

```
375 CP%=1:P1$=P1$+" ":LOCATE 12,1:PRINT
SPACE$(120);:LOCATE 12,1:PRINT P1$:PRINT
 #1,CHR$(27);CHR$(48);:IF WIDE>6 THEN PR
INT #1,CHR$(15)
420 WW=0:IF WIDE<6 AND WIDE>3 THEN PRINT
 #1,CHR$(15):WW=14
430 IF WIDE<=3 THEN WW=14
```

Cut P1$ into words.

```
440 PRINT #1,:I=1
460 L$=""
470 IF I>LEN(P1$) THEN LOCATE 12,1:PRINT
 SPACE$(120);:LOCATE 12,1:GOTO 810
480 M$=MID$(P1$,I,1):IF M$<>" " THEN L$=
L$+M$:I=I+1:GOTO 470
```

Center each word, unless the last character of the previous word was a hyphen. To center every word, regardless of hyphenation, delete everything in line 500 that is after the second quotation mark.

```
500 I=I+1:L=LEN(L$):S$="":IF RIGHT$(OL$,
1)="-" THEN HYFLAG=-1
530 IF L>WIDE THEN I=I-(L-WIDE)-2:L=WIDE
:L$=LEFT$(L$,WIDE-1)+"-"
540 IF L<WIDE-1 AND NOT HYFLAG THEN FOR
X=1 TO (WIDE-L)/2:S$=S$+" ":NEXT X
```

Print the current word, and go on to the next one.

```
550 OL$=L$:HYFLAG=0:L$=S$+L$:FOR LIN3=1
TO 13:FOR Q=1 TO LEN(L$):M$=MID$(L$,Q,1)
:FOR R=1 TO LTR-1:IF L$(R)<>M$ THEN NEXT
 R
610 II=1:SCOUNT=0
620 S=DESC%(R,LIN3,II):C=DESC%(R,LIN3,II
+1):IF NOT CFLAG THEN CP$=L$(R)
632 IF S=-1 GOTO 650
634 PRINT #1,CHR$(WW);SPC(S);:SCOUNT=SCO
UNT+S:FOR X=1 TO C:IF CFLAG THEN CP$=MID
$(CH$,CP%,1):CP%=CP%+1:IF CP%>LEN(CH$) T
HEN CP%=1
638 PRINT #1,CP$;:NEXT X:SCOUNT=SCOUNT+C
:II=II+2:GOTO 620
650 IF SCOUNT<13 THEN FOR X=SCOUNT TO 12
:PRINT #1," ";:NEXT X
660 NEXT Q:PRINT #1,:NEXT LIN3:PRINT #1,
:PRINT #1,:GOTO 460
```

Subroutine to place messages centered on line VT in normal, flashing, or inverse text.

```
760 ON DISPLAY GOTO 770,780,790
770 COLOR 7,0:GOTO 800
780 COLOR 0,7:GOTO 800
790 COLOR 23,0
800 LOCATE VT,20-LEN(A$)/2:PRINT A$:COLO
R 7,0:RETURN
810 '
```

Done with this poster; do another?

```
830 LINE INPUT"More? ";A$:IF FNU$(LEFT$(
A$,1))="N" THEN SCREEN 0,0,0:KEY ON:WIDT
H 80:CLS:CLOSE #1:END ELSE PRINT #1,CHR$
(18);CHR$(20):LOCATE 12,1:PRINT SPACE$(6
0);SPACE$(255):GOTO 200
```

Data for characters. See article for data format.

```
10000 DATA A,4,4,-1,-1,3,6,-1,-1,2,8,-1,
-1,1,10,-1,-1,0,12,-1,-1,0,4,4,4,-1,-1,0
,4,4,4,-1,-1,0,12,-1,-1,0,12,-1,-1,0,12,
-1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,4,4,
4,-1,-1
10001 DATA E,0,12,-1,-1,0,12,-1,-1,0,12,
-1,-1,0,4,-1,-1,0,4,-1,-1,0,10,-1,-1,0,1
0,-1,-1,0,10,-1,-1,0,4,-1,-1,0,4,-1,-1,0
,12,-1,-1,0,12,-1,-1,0,12,-1,-1
10002 DATA " ",-1,-1,-1,-1,-1,-1,-1,-1,-
1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1
,-1,-1,-1
10003 DATA I,2, 8,-1,-1,2, 8,-1,-1,2, 8,
-1,-1,4,4,-1,-1,4,4,-1,-1,4,4,-1,-1,4,4,
-1,-1,4,4,-1,-1,4,4,-1,-1,4,4,-1,-1,2, 8
,-1,-1,2, 8,-1,-1,2, 8,-1,-1
10004 DATA O,2,8,-1,-1,1,10,-1,-1,0,12,-
1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,4,4,4
,-1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,4,4
,4,-1,-1,0,4,4,4,-1,-1,0,12,-1,-1,1,10,-
1,-1,2,8,-1,-1
10005 DATA U,0,4,4,4,-1,-1,0,4,4,4,-1,-1
,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,
-1,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,4,4,4,-
1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,12,-1
,-1,1,10,-1,-1,2,8,-1,-1
10006 DATA "-",-1,-1,-1,-1,-1,-1,-1,-1,-
1,-1,-1,-1,1,10,-1,-1,1,10,-1,-1,-1,-1,-
1,-1,-1,-1,-1,-1,-1,-1
10007 DATA C,2,10,-1,-1,1,11,-1,-1,0,12,
-1,-1,0,4,-1,-1,0,4,-1,-1,0,4,-1,-1,0,4,
-1,-1,0,4,-1,-1,0,4,-1,-1,0,4,-1,-1,0,4,
-1,-1,0,4,-1,-1,0,4,-1,-1,0,12
,-1,-1,1,11,-1,-1,2,10,-1,-1
10008 DATA R,0,10,-1,-1,0,11,-1,-1,0,12,
-1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,12,-
1,-1,0,11,-1,-1,0,10,-1,-1,0,8,-1,-1,0,4
,1,4,-1,-1,0,4,2,4,-1,-1,0,4,3,4,-1,-1,0
,4,4,4,-1,-1
10009 DATA S,2,9,-1,-1,1,11,-1,-1,0,12,-
1,-1,0,4,-1,-1,0,4,-1,-1,0,10,-1,-1,1,10
,-1,-1,2,10,-1,-1,8,4,-1,-1,8,4,-1,-1,0,
12,-1,-1,0,11,-1,-1,1,9,-1,-1
10010 DATA N,0,4,4,4,-1,-1,0,5,3,4,-1,-1
,0,6,2,4,-1,-1,0,7,1,4,-1,-1,0,7,1,4,-1,
-1,0,4,1,7,-1,-1,0,4,1,7,-1,-1,0,4,2,6,-
1,-1,0,4,3,5,-1,-1,0,4,4,4,-1,-1,0,4,4,4
,-1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,-1
10011 DATA B,0,10,-1,-1,0,11,-1,-1,0,12,
-1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,12,-
1,-1,0,11,-1,-1,0,11,-1,-1,0,4,4,4,-1,-1
,0,4,4,4,-1,-1,0,12,-1,-1,0,11,-1,-1,0,1
0,-1,-1
10012 DATA D,0,10,-1,-1,0,11,-1,-1,0,12,
-1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,4,4,
4,-1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,4,
4,4,-1,-1,0,4,4,4,-1,-1,0,12,-1,-1,0,11,
-1,-1,0,10,-1,-1
10013 DATA F,0,12,-1,-1,0,12,-1,-1,0,12,
-1,-1,0,4,-1,-1,0,4,-1,-1,0,8,-1,-1,0,8,
-1,-1,0,8,-1,-1,0,4,-1,-1,0,4,-1,-1,0,4,
-1,-1,0,4,-1,-1,0,4,-1,-1
10014 DATA H,0,4,4,4,-1,-1,0,4,4,4,-1,-1
,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,
-1,0,12,-1,-1,0,12,-1,-1,0,12,-1,-1,0,4,
4,4,-1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,
4,4,4,-1,-1,0,4,4,4,-1,-1
10015 DATA J,8,4,-1,-1,8,4,-1,-1,8,4,-1,
-1,8,4,-1,-1,8,4,-1,-1,8,4,-1,-1,8,4,-1,
-1,8,4,-1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,-1
,0,12,-1,-1,1,10,-1,-1,2,8,-1,-1
10016 DATA T,0,12,-1,-1,0,12,-1,-1,0,12,
-1,-1,4,4,-1,-1,4,4,-1,-1,4,4,-1,-1,4,4,
-1,-1,4,4,-1,-1,4,4,-1,-1,4,4,-1,-1,4,4,
-1,-1,4,4,-1,-1,4,4,-1,-1
10017 DATA Y,0,4,4,4,-1,-1,0,4,4,4,-1,-1
,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,
-1,1,10,-1,-1,2,8,-1,-1,4,4,-1,-1,4,4,-1
,-1,4,4,-1,-1,4,4,-1,-1,4,4,-1,-1,4,4,-1
,-1
10018 DATA V,1,3,4,3,-1,-1,1,3,4,3,-1,-1
,1,3,4,3,-1,-1,1,3,4,3,-1,-1,1,3,4,3,-1,
-1,1,3,4,3,-1,-1,1,3,4,3,-1,-1,1,3,4,3,-
1,-1,1,3,4,3,-1,-1,2,3,2,3,-1,-1,3,2,2,2
,-1,-1,4,4,-1,-1,5,2,-1,-1
10019 DATA W,0,3,6,3,-1,-1,0,3,6,3,-1,-1
,0,3,6,3,-1,-1,0,3,6,3,-1,-1,0,3,6,3,-1,
-1,0,3,2,2,2,3,-1,-1,0,3,1,4,1,3,-1,-1,0
,3,1,4,1,3,-1,-1,0,12,-1,-1,0,5,2,5,-1,-
1,0,5,2,5,-1,-1,0,4,4,4,-1,-1,0,3,6,3,-1
,-1
10020 DATA L,0,4,-1,-1,0,4,-1,-1,0,4,-1,
-1,0,4,-1,-1, 0,4,-1,-1,0,4,-1,-1,0,4,-1
,-1,0,4,-1,-1,0,4,-1,-1,0,4,-1,-1,0,12,-
1,-1,0,12,-1,-1,0,12,-1,-1
10021 DATA G,2,10,-1,-1,1,11,-1,-1,0,12,
-1,-1,0,4,-1,-1,0,4,-1,-1,0,4,-1,-1,0,4,
3,5,-1,-1,0,4,3,5,-1,-1,0,4,5,3,-1,-1,0,
4,5,3,-1,-1,0,12,-1,-1,1,10,-1,-1,2,8,-1
,-1
10022 DATA Z,0,12,-1,-1,0,12,-1,-1,0,12,
-1,-1,8,4,-1,-1,7,4,-1,-1,6,4,-1,-1,5,4,
-1,-1,4,4,-1,-1,3,4,-1,-1,2,4,-1,-1,1,11
,-1,-1,0,12,-1,-1,0,12,-1,-1
10023 DATA M,0,3,6,3,-1,-1,0,4,4,4,-1,-1
```

```
,0,5,2,5,-1,-1,0,5,2,5,-1,-1,0,12,-1,-1,
0,3,1,4,1,3,-1,-1,0,3,1,4,1,3,-1,-1,0,3,
2,2,2,3,-1,-1,0,3,6,3,-1,-1,0,3,6,3,-1,-
1,0,3,6,3,-1,-1,0,3,6,3,-1,-1,0,3,6,3,-1
,-1
10024 DATA Q,1,8,-1,-1,0,10,-1,-1,0,10,-
1,-1,0,3,4,3,-1,-1,0,3,4,3,-1,-1,0,3,4,3
,-1,-1,0,3,4,3,-1,-1,0,3,4,3,-1,-1,0,3,2
,5,-1,-1,0,3,3,4,-1,-1,0,11,-1,-1,1,11,-
1,-1,2,6,2,2,-1,-1
10025 DATA P,0,10,-1,-1,0,11,-1,-1,0,12,
-1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,12,-
1,-1,0,11,-1,-1,0,10,-1,-1,0,4,-1,-1,0,4
,-1,-1,0,4,-1,-1,0,4,-1,-1,0,4,-1,-1
10026 DATA K,0,4,4,4,-1,-1,0,4,3,5,-1,-1
,0,4,2,5,-1,-1,0,4,1,4,-1,-1,0,8,-1,-1,0
,6,-1,-1,0,7,-1,-1,0,8,-1,-1,0,4,1,4,-1,
-1,0,4,2,4,-1,-1,0,4,3,4,-1,-1,0,4,4,4,-
1,-1,0,4,5,3,-1,-1
10027 DATA X,0,4,4,4,-1,-1,0,4,4,4,-1,-1
,0,4,4,4,-1,-1,1,3,4,3,-1,-1,2,3,2,3,-1,
-1,3,6,-1,-1,4,4,-1,-1,4,4,-1,-1,3,6,-1,
-1,2,3,2,3,-1,-1,0,4,4,4,-1,-1,0,4,4,4,-
1,-1,0,4,4,4,-1,-1
10028 DATA "'",4,5,-1,-1,4,5,-1,-1,4,5,-
1,-1,4,5,-1,-1,3,5,-1,-1,2,5,-1,-1,-1,-1
,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1
10029 DATA !,3,5,-1,-1,3,5,-1,-1,3,5,-1,
-1,3,5,-1,-1,3,5,-1,-1,3,5,-1,-1,3,5,-1,
-1,3,5,-1,-1,4,3,-1,-1,-1,-1,3,5,-1,-1,3
,5,-1,-1,3,5,-1,-1
10030 DATA &,5,3,-1,-1,5,3,-1,-1,2,10,-1
,-1,1,11,-1,-1,0,5,-1,-1,1,9,-1,-1,3,7,-
1,-1,1,9,-1,-1,0,5,-1,-1,1,11,-1,-1,2,10
,-1,-1,5,3,-1,-1,5,3,-1,-1
10031 DATA ".",-1,-1,-1,-1,-1,-1,-1,-1,-
1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,0,4,-
1,-1,0,4,-1,-1,0,4,-1,-1
10032 DATA ",",-1,-1,-1,-1,-1,-1,-1,-1,-
1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,2,3,-
1,-1,1,3,-1,-1,0,3,-1,-1
10033 DATA "+",-1,-1,-1,-1,-1,-1,4,3,-1,
-1,4,3,-1,-1,4,3,-1,-1,0,11,-1,-1,0,11,-
1,-1,4,3,-1,-1,4,3,-1,-1,4,3,-1,-1,-1,-1
,-1,-1
10034 DATA =,-1,-1,-1,-1,-1,-1,-1,-1,0,1
2,-1,-1,0,12,-1,-1,-1,-1,0,12,-1,-1,0,12
,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1
10035 DATA ;,-1,-1,-1,-1,-1,-1,2,4,-1,-1
,2,4,-1,-1,2,4,-1,-1,-1,-1,-1,-1,-1,-1,2
,4,-1,-1,2,4,-1,-1,1,4,-1,-1,0,4,-1,-1
10036 DATA /,10,2,-1,-1,10,2,-1,-1,9,3,-
1,-1,8,3,-1,-1,7,3,-1,-1,6,3,-1,-1,5,3,-
1,-1,4,3,-1,-1,3,3,-1,-1,2,3,-1,-1,1,3,-
1,-1,0,3,-1,-1,0,2,-1,-1
10037 DATA %,0,4,6,2,-1,-1,0,4,6,2,-1,-1
,0,4,5,3,-1,-1,0,4,4,3,-1,-1,7,3,-1,-1,6
,3,-1,-1,5,3,-1,-1,4,3,-1,-1,3,4,-1,-1,2
,3,3,4,-1,-1,1,3,4,4,-1,-1,0,3,5,4,-1,-1
,0,2,6,4,-1,-1
10038 DATA (,5,2,-1,-1,4,2,-1,-1,3,2,-1,
-1,2,2,-1,-1,1,2,-1,-1,0,3,-1,-1,0,2,-1,
-1,0,3,-1,-1,1,2,-1,-1,2,2,-1,-1,3,2,-1,
-1,4,2,-1,-1,5,2,-1,-1
10039 DATA ),5,2,-1,-1,6,2,-1,-1,7,2,-1,
-1,8,2,-1,-1,9,2,-1,-1,9,3,-1,-1,10,2,-1
,-1,9,3,-1,-1,9,2,-1,-1,8,2,-1,-1,7,2,-1
,-1,6,2,-1,-1,5,2,-1,-1
10040 DATA ":",-1,-1,-1,-1,2,4,-1,-1,2,4
,-1,-1,2,4,-1,-1,-1,-1,-1,-1,-1,-1,2,4,-
1,-1,2,4,-1,-1,2,4,-1,-1,-1,-1,-1,-1
10041 DATA 1,4,4,-1,-1,4,4,-1,-1,3,5,-1,
-1,2,6,-1,-1,5,3,-1,-1,5,3,-1,-1,5,3,-1,
-1,5,3,-1,-1,5,3,-1,-1,5,3,-1,-1,3,7,-1,
-1,3,7,-1,-1,3,7,-1,-1
10042 DATA 2,3,6,-1,-1,2,8,-1,-1,1,3,3,4
,-1,-1,8,3,-1,-1,8,3,-1,-1,7,3,-1,-1,6,3
,-1,-1,5,3,-1,-1,4,3,-1,-1,3,3,-1,-1,2,1
0,-1,-1,1,11,-1,-1,1,11,-1,-1
10043 DATA 0,2,8,-1,-1,1,10,-1,-1,0,12,-
1,-1,0,4,3,5,-1,-1,0,4,2,6,-1,-1,0,4,2,6
,-1,-1,0,4,1,2,1,4,-1,-1,0,4,1,2,1,4,-1,
-1,0,6,2,4,-1,-1,0,5,3,4,-1,-1,0,12,-1,-
1,1,10,-1,-1,2,8,-1,-1
10044 DATA 4,7,3,-1,-1,6,4,-1,-1,5,5,-1,
-1,4,6,-1,-1,3,3,1,3,-1,-1,2,3,2,3,-1,-1
,1,3,3,3,-1,-1,0,12,-1,-1,0,12,-1,-1,0,1
2,-1,-1,7,3,-1,-1,7,3,-1,-1,7,3,-1,-1
10045 DATA 8,2,8,-1,-1,1,10,-1,-1,0,12,-
1,-1,0,3,6,3,-1,-1,0,3,6,3,-1,-1,1,10,-1
,-1,2,8,-1,-1,1,2,6,2,-1,-1,0,3,6,3,-1,-
1,0,3,6,3,-1,-1,0,12,-1,-1,1,10,-1,-1,2,
8,-1,-1
10046 DATA 3,2,8,-1,-1,1,10,-1,-1,0,12,-
1,-1,0,3,6,3,-1,-1,9,3,-1,-1,4,7,-1,-1,4
,6,-1,-1,8,3,-1,-1,9,3,-1,-1,0,3,6,3,-1,
-1,0,12,-1,-1,1,10,-1,-1,2,8,-1,-1
10047 DATA 5,0,11,-1,-1,0,11,-1,-1,0,11,
-1,-1,0,4,-1,-1,0,4,-1,-1,0,10,-1,-1,0,1
1,-1,-1,0,12,-1,-1,8,4,-1,-1,0,4,4,4,-1,
-1,0,4,4,4,-1,-1,1,10,-1,-1,2,8,-1,-1
10048 DATA 6,2,8,-1,-1,1,10,-1,-1,0,12,-
1,-1,0,4,4,4,-1,-1,0,4,-1,-1,0,10,-1,-1,
0,11,-1,-1,0,12,-1,-1,0,4,4,4,-1,-1,0,4,
4,4,-1,-1,0,12,-1,-1,1,10,-1,-1,2,8,-1,-
1
10049 DATA 9,2,8,-1,-1,1,10,-1,-1,0,12,-
1,-1,0,4,4,4,-1,-1,0,4,4,4,-1,-1,0,12,-1
,-1,1,11,-1,-1,2,10,-1,-1,8,4,-1,-1,0,4,
4,4,-1,-1,0,12,-1,-1,1,10,-1,-1,2,8,-1,-
1
10050 DATA 7,0,12,-1,-1,0,12,-1,-1,0,12,
-1,-1,8,4,-1,-1,7,4,-1,-1,6,4,-1,-1,5,4,
-1,-1,4,4,-1,-1,3,4,-1,-1,3,4,-1,-1,3,4,
-1,-1,3,4,-1,-1,3,4,-1,-1
10051 DATA ?,2,8,-1,-1,1,10,-1,-1,0,4,4,
4,-1,-1,0,4,4,4,-1,-1,8,4,-1,-1,4,7,-1,-
1,4,6,-1,-1,4,4,-1,-1,4,4,-1,-1,-1,-1,4,
4,-1,-1,4,4,-1,-1,4,4,-1,-1
10052 DATA >,2,3,-1,-1,3,3,-1,-1,4,3,-1,
-1,5,3,-1,-1,6,3,-1,-1,7,3,-1,-1,8,3,-1,
-1,7,3,-1,-1,6,3,-1,-1,5,3,-1,-1,4,3,-1,
-1,3,3,-1,-1,2,3,-1,-1
10053 DATA <,7,3,-1,-1,6,3,-1,-1,5,3,-1,
-1,4,3,-1,-1,3,3,-1,-1,2,3,-1,-1,1,3,-1,
-1,2,3,-1,-1,3,3,-1,-1,4,3,-1,-1,5,3,-1,
-1,6,3,-1,-1,7,3,-1,-1
63999 DATA XXX
```

IBM PC® SWAT TABLE FOR:
POSTER MAKER
(Modified Parameters: NU = 3
B = 500)

LINES	SWAT CODE	LENGTH
10 - 70	YE	365
80 - 110	IU	125
140 - 160	ER	196
190 - 220	MF	296
260 - 330	QP	197
340 - 370	WC	314
372 - 375	LJ	188
420 - 440	NO	83
460 - 480	TS	110
500 - 540	LZ	172
550 - 620	XI	188
632 - 638	QS	172
650 - 760	KY	115
770 - 790	HB	43
800 - 830	UF	160
10000 - 10002	WY	398
10003 - 10005	TR	494
10006 - 10008	TW	410
10009 - 10011	CH	498
10012 - 10014	UI	494
10015 - 10017	UM	448
10018 - 10020	WI	523
10021 - 10023	LP	502
10024 - 10026	GZ	500
10027 - 10029	RG	420
10030 - 10032	BA	342
10033 - 10035	DJ	342
10036 - 10038	PG	448
10039 - 10041	QQ	389
10042 - 10044	QY	476
10045 - 10047	YS	466
10048 - 10050	FB	455
10051 - 10053	VA	419
63999 - 63999	JW	10

Home Finance Program For The IBM®PC

Reviewed by Katherine Ackerman and Glen N. Ackerman, M.D.

from Design Data Systems Corporation, 5270 N. Park Place N.E., Cedar Rapids, IA 52402. System requirements: IBM PC with 64K RAM, monochrome or color display, one disk drive and printer (optional). Suggested retail price: $100.00.

Attention Average Homeowner and IBM-PC Owner!

The "Home Finance Program For the IBM Personal Computer" is an early answer to the current shortage of IBM PC-specific software. Our initial impressions? A no-nonsense, no-frills, flexible series of four programs: budget analysis, checking account analysis, savings account analysis and a loan amortization program. The package includes two disks (a program disk and a formatted data disk) along with a looseleaf notebook of 61 pages of easy-to-read instructions.

Since there are some frustrating traps for the new user, we suggest you begin by entering only brief amounts of data into the programs until you understand thoroughly how each program works.

Budget Analysis

The budget program makes a tedious chore fun, even for those who otherwise might never have constructed a budget for home use. After creating a new file to contain the budget, fourteen major categories for monthly expenditures are viewed. Although other programs offer as many as 24 categories, fourteen seems to be adequate if you use the "miscellaneous" category. A great deal of flexibility exists to tailor the budget categories to any particular set of circumstances. For each major category, (e.g. "Food and Groceries," "Shelter," "Vices") the user inputs subcategories — thus the flexibility. One can subdivide "Food and Groceries" into "Eating Out," "Pet Food" and so on. Editing and revising these subcategories is easy. In fact, all editing and revising in this particular program are easy.

"A no-nonsense, no-frills, flexible series of four programs: budget analysis, checking account analysis, savings account analysis and a loan amortization program."

The next step of the budget program is to enter either predicted or actual expenses. We found it more useful to enter the actual amounts to see where the money went, instead of predicting expenditures. Even a computer cannot predict when the car will break down! However, many people will want to compare their predicted and actual expenditures. Figures are always entered for only one month at a time into the subcategories already established. After entering the amounts, you have several options: reviewing past budgets, changing actual or predicted expenditures, viewing year-to-date expenditures, or ending the program and returning to the main menu after storing data. A nice feature is that the function keys control these multiple options.

We found this program the most satisfying of the four, but it did have some problems. Designed for single disk drive systems, there is the inconvenience of switching back and forth from program disk to data disk. Another annoyance is the speed with which the categories and subcategories scroll across the screen. They are impossible to read, and it is a challenge to press the "CTRL NUMLOCK" keys before half of the categories have disappeared. The most frustrating problem is the inadequate warning that stopping the program or switching disks before using the F10 funtion key, ("end the program" key), causes loss of all data.

Checking Account Analysis

This section allows you to maintain a checkbook register. As with a manual system, you record checks written, deposits made and interest paid. Entry of each month's cancelled checks allows the computer to bring the account up to date. The

searching criteria for checks written or deposits made are the check numbers or dates. As each check is entered, the question "Is this check tax deductible?" appears on the screen, a handy feature at tax time. By using several data disks, you can maintain more than one account.

All of this sounds good, right? Unfortunately, it's not that simple. Balancing the checkbook is a tedious chore anyway, and a computer program must have special appeal to make the task faster or more enjoyable. This program doesn't tempt us to abandon a pen, calculator and the check register. One program that did, though, was the checkbook program included in the FriendlyWare P.C. Introductory set, used as a comparison for this review.

The first major drawback to Design Data's checkbook program is the 27 pages of instructional material. True, they are short, double-spaced pages, but this compares with less than 1 1/2 pages in the FriendlyWare program. The FriendlyWare approach is a graphic presentation on the screen. This saves the trouble of flipping through a manual and is "friendlier" for those of us who don't use computers for strictly financial applications.

The FriendlyWare process is also better for labeling checks. Numerical codes can be assigned to each check to indicate the type of expense incurred. (tax deductions travel expenses, etc.) This "coding" is useful in answering such questions as "How much did we spend on computer software this year?" Design Data only identifies checks as tax deductible or non-deductible.

The Design Data program allows you to search the check register by date and check number. More useful, though, is the ability to search checks by recipient. It allows the user to review the computer version of the register in much the same way as the more familiar paper register. This is a feature of many checkbook programs, but not Design Data's.

Loading data into the checkbook program is time-consuming. The user is hampered by a typo which tells you to choose the appropriate function key, when you actually need a numerical key. (A minor problem, but annoying just the same.) The FriendlyWare program has another feature which we would like to see on the Design Data programs. This is the message, "Entries will not be saved if you escape now — is that O.K.?" This is comforting, especially since we both lost data on the Design Data programs. Finally, there is the frustrating dilemma of data scrolling much too fast to read.

The Savings Account Analysis and Loan Amortization Programs

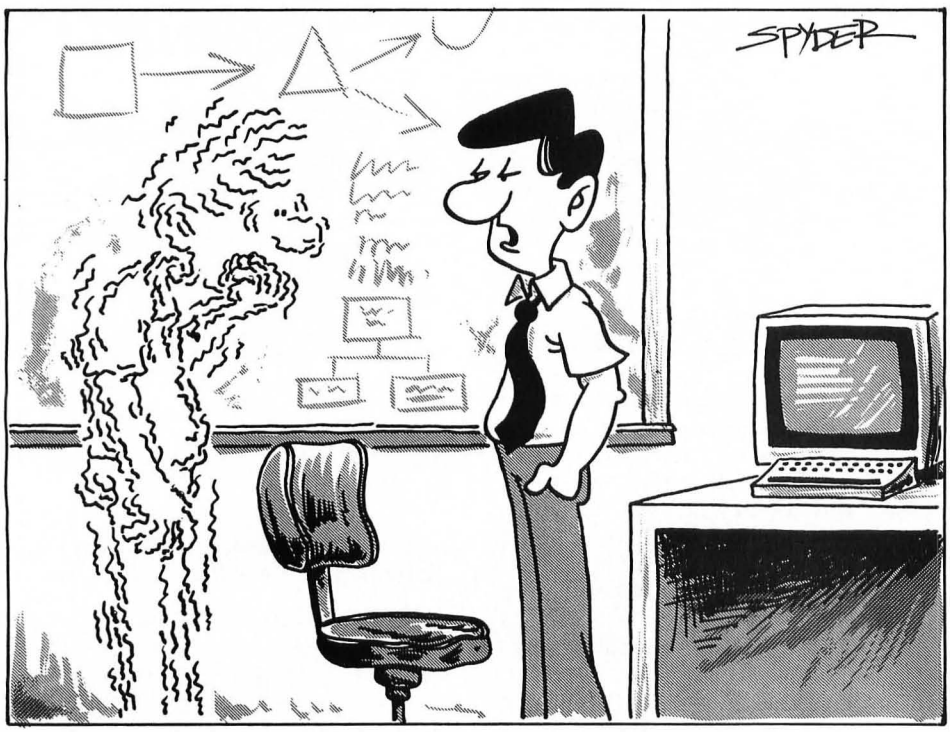
The savings account program allows you to enter combinations of principal, interest rates, and duration of a savings account or savings certificate. You can then calculate interest income, length of time needed to achieve a certain amount of interest, and so on. It's a useful tool for examining the benefit of any savings account, money market, or savings certificate.

The loan amortization program is similar in nature — allowing you to estimate home mortgage payments with a pleasing display of monthly payments broken into interest, principal, and outstanding balance based on interest rates. Our only complaint here is the vast quantity of data on the screen with no mechanism for easy escape. (A twenty-five year mortgage generates 300 lines of figures!)

Our overall impressions? Design Data offers a good budget analysis program that is quite flexible and useful for most people's home finances. The checkbook program is fair — it offers no graphics, and is not very interesting to use. The savings account analysis program and the loan amortization program are useful, but are really "filler" programs easily found in other places (like on your DOS disk.) The suggested retail price for the Design Data program is around $100. Judging from our personal budget analysis, this is somewhat expensive.

However, if we were to borrow $100 at 10.9% interest.....

SoftTakes



"WE'RE A TEENY BIT AFRAID OF COMPUTERS, AREN'T WE?"

TURRET AND TRACK

by Ron Potkin

***Turret and Track* is a two-player computer war-game for a TRS-80® Model I or III with at least 32K RAM and one disk drive. It is included as the bonus program on issue 37 TRS-80 DV. See the Bind-in Card elsewhere in this issue to order this issue's disk.**

Imagine yourself and an opponent sitting in Battle Headquarters at a control console. From this angle, you can observe your tanks and those of your opponent. Both of you control your tanks from the console. You win when you hit the opposing headquarters and deactivate your opponent's tanks.

The first campaign takes place in a town which provides plenty of shelter for both sides; the second and last occurs in a desert. This change of scene permits each side to exercise strategic abilities and cunning.

The Pieces

Each commander has five pieces — four tanks and headquarters. The initial strength and capability count of each tank follows:

Movement:	9 squares
Range:	9 squares
Shots:	3 rounds per turn
Armor:	9 units

Squares do not appear on the screen, but the town blocks are broken up to indicate the size of each square. Headquarters has no offensive or defensive capability, but you may move it one square during each round.

The Board

The tanks and the headquarters line up on the west and east sides of the town. The bottom three lines handle the input of orders and messages. The bottom left and right-hand corners show the total strength of each side.

West's tanks are named W1, W2, W3 and W4. East's tanks are similarly named E1, E2 and so on. Against each is shown a strength quotient, i.e. 9939 — movement, range, shots and armor (MRSA). The six orders you may give to each tank are indicated by the six periods displayed after each tank's strength.

Order of Play

1. West gives orders to each of his tanks. (See Fig. 1)
2. His orders are removed from the screen.
3. East gives his orders. (See Fig. 1)
4. His orders are removed.

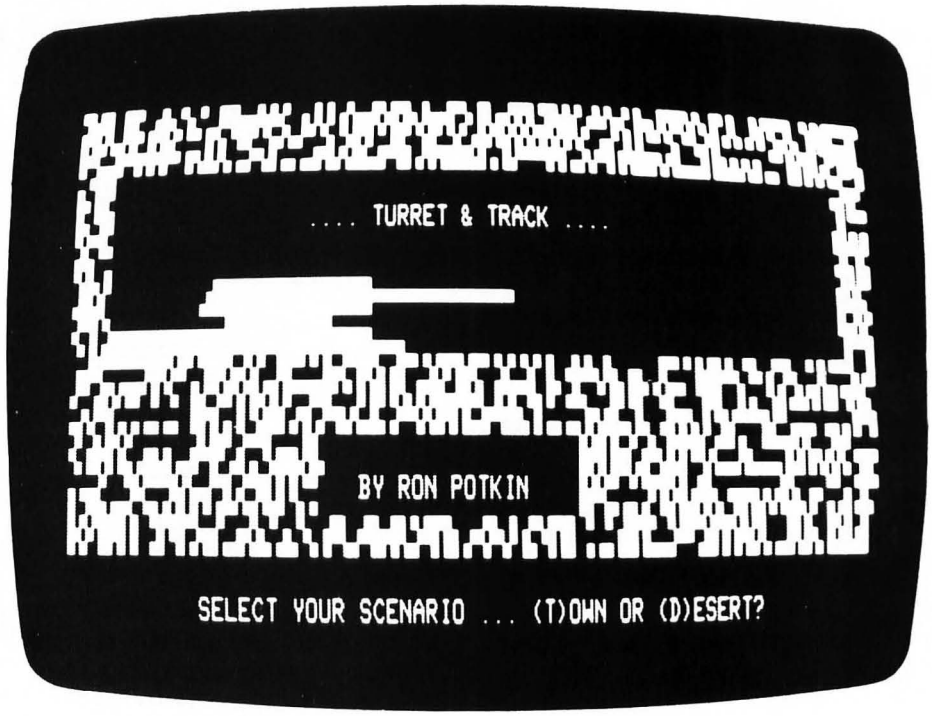
5. West moves his Headquarters, then East moves his.
6. The computer calculates a random sequence # and the tanks move in that order.
7. The game ends if you hit a headquarters or the strength of one side drops below 20.
8. Steps one to seven repeat.

Movement

Once both commanders complete tank orders, the game determines the order of movement and so indicates at the bottom of the screen. A "move" is defined as: 1) Moving one square, 2) Firing one shot, 3) One delay, or 4) One change of direction. Moving four squares counts as four moves.

A piece will stop and cease following orders if: 1) It hits an obstruction, 2) It is destroyed, or 3) A hit cripples it so that it cannot complete its orders. If this occurs, an "*" will appear in place of the next move.

The degree of damage you suffer from a tank attack depends on the distance between tanks. A frontal hit hurts less than a broadside strike which is, in turn, less dangerous than a hit in the rear. The design notes explain how to calculate the chance of a hit and the degree of damage. Whatever the type of shot, the armor will suffer some damage. When you reduce this value to zero, you destroy the piece and reduce all remaining values to zero.



The game moves quickly, as you can maneuver in over 100 different ways. You can single-step by pressing "@". A move is then carried out each time you press the spacebar. Press ENTER to return to normal operation.

Design Notes

In this game, when two opposing tanks meet face-to-face, devastation is inevitable. Does one stay and fight to the death? It is suicide to turn and thus take more punishment. One way out of this dilemma is to arrange a truce and mutually agree to move away. But can you trust your opponent?

A player can shoot at his own tanks and even at his own headquarters — which means instant loss. You must therefore be very precise in the timing of your orders.

A hit change is calculated as follows: 85% minus 5% for every square of distance.

Figure 1

Giving Orders

An arrow will appear against W1 and a flashing hyphen will prompt for input. The tanks will stop while you give orders, but the arrow will move to indicate position and direction.

You may give the following orders up to a maximum of six per turn.

Numbers 1 to 9 move that number of squares

Direction (N,E,W,S)	Change of direction. You can use the four arrow keys instead.
Delay (D)	Do nothing for one move.
Fire (R)	Fire at the turret of an enemy. A successful hit will affect the enemy tank's Range.
Fire (M)	Fire at the tracks (affect Movement).
Rescind (B)	i.e. Backspace.
< ENTER>	Completes orders to a tank.

Examples

W1 4S3EM2<ENTER>	Move 4 squares Turn South Move 3 squares Turn East Fire at Track Move 2 squares
W2 D1NRM <ENTER>	Delay 1 move Move 1 square Turn North Fire at Turret Fire at Track
W3 1SR <ENTER>	Move 1 square Turn South Fire at Turret

Damage Points

The degree of damage depends on the distance, and the thickness of the target tank's armor. Refer to the following tables:

Table 1

FRONTAL

D = Destroyed, M = Minor Damage

Armour \ Range	1	2	3	4	5	6	7	8	9
1	D	D	D	D	D	D	D	D	D
2	D	D	D	D	D	D	1	1	1
3	D	D	2	2	2	1	1	1	M
4	3	2	2	2	1	1	1	M	M
5	2	2	2	1	1	1	M	M	M
6	2	2	1	1	1	M	M	M	M
7	2	1	1	1	M	M	M	M	M
8	1	1	1	M	M	M	M	M	M
9	1	1	M	M	M	M	M	M	M

Table 2

SIDE

D = Destroyed, M = Minor Damage

Armour \ Range	1	2	3	4	5	6	7	8	9
1	D	D	D	D	D	D	D	D	D
2	D	D	D	D	D	D	D	D	1
3	D	D	D	D	2	2	2	1	1
4	3	3	3	2	2	2	1	1	1
5	3	3	2	2	2	1	1	1	M
6	3	2	2	2	1	1	1	M	M
7	2	2	2	1	1	1	M	M	M
8	2	2	1	1	1	M	M	M	M
9	2	1	1	1	M	M	M	M	M

Table 3

REAR

D = Destroyed, M = Minor Damage

Armour \ Range	1	2	3	4	5	6	7	8	9
1	D	D	D	D	D	D	D	D	D
2	D	D	D	D	D	D	D	D	D
3	D	D	D	D	D	D	2	2	2
4	D	3	3	3	3	2	2	2	1
5	3	3	3	3	2	2	2	1	1
6	3	3	3	2	2	2	1	1	1
7	3	3	2	2	2	1	1	1	M
8	3	2	2	2	1	1	1	M	M
9	2	2	2	1	1	1	M	M	M

ILIST PATCH PROGRAM

by Joseph Iwanski

***ILIST/BAS* is an update to *ILIST/CMD*, published on issue 33 TRS-80® DV. You must have *ILIST/CMD* to run this program. It requires 48K and one disk drive and assumes, but does not require, a printer. It is included as a bonus program on this issue's TRS-80 DV. See the Bind-in Card elsewhere in this issue to order this month's disk.**

ILIST/BAS is a BASIC program executed to patch *ILIST/CMD* for the following options:

(1) Change all lower-case video output to upper-case;

(2) Change Epson control codes to service GRAFTRAX-80;

(3) Change the Standard Option List;

It also squashes a harmless, but odious bug.

How to run *ILIST/BAS*

1. Have *ILIST/CMD* available on a disk drive. The patch program uses a generic filespec, so it will search all drives for *ILIST/CMD*. If you have copies in more than one drive, it will always access the copy on the lowest drive number.

2. RUN "ILIST/BAS". The program is self-explanatory; it gives you the option of running any of three different patch routines. You can run this patch program over and over again on the same copy of *ILIST*. If you make a mistake, just run it again.

3. *Important!* — Terminate the patch program by executing Option 4, the "Termination" option. This will ensure that *ILIST/CMD* is closed properly.

Patch Options

Automatic: Bug Out. The original copy of *ILIST* listed statements containing the token LINE INPUT as LNE INPUT. Running the patch program automatically eliminates this bug; you don't have to specify this option.

(1) Upper-Case. The original *ILIST* works fine on the Model I without the lower-case hardware modification, but displays many lower-case characters on the screen, resulting in an unsightly display. The Upper-Case patch changes all video displays to upper-case.

(2)Epson Modification. The original *ILIST* is configured for a vanilla Epson MX printer — without Graftrax. Running it with a Graftrax printer creates an ugly printout. The patch routine allows you to change *ILIST* to "GRAFTRAX-80 EPSON", and back to "Vanilla EPSON", according to your mood and equipment.

(3) Option Modification. *ILIST* was published with a built-in standard option set. It may be convenient to have a different set of standard options. For example, if you don't have an Epson, the standard option should indicate EPSON — NO. The patch routine in *ILIST/BAS* allows configuration of the standard option to your specifications.

(4) Exit Program.
Remember! Terminate *ILIST/BAS* with option 4!

The Adventure is Waiting for You...

How would you like to go back in time to 19th century London to match wits with Jack the Ripper? Out into space to brave the swirling vortex of a black hole? Into the depths of the ocean, or on a quest to rescue a beautiful princess from the clutches of evil monsters?

You never know where the **SoftSide Adventure Series** might take you. But you can be sure that with each adventure, you will experience new delights and new challenges. Imagine yourself tracking down the deadly **Dalton Gang** in the middle of the Arizona desert or stalking killer mice who are devouring the city of Chicago.

Starting with issue 38 of **SoftSide,** our adventures will be included on each issue's CV and DV. But, you can still order individual copies of our adventures from the past on cassette or disk. We also have **Adventure SuperDisks** available with three adventures on each disk.

Individual copies of past adventures cost $7 on cassette or $10 on disk. SuperDisks are priced at only $26 each. Place your order today. Supplies are limited and some of the more popular adventures may sell out quickly. Use the handy, postage-free bind-in card at the right to order your adventure experience. Warn your friends and family — they may not see you again after you've set out on your quest!

Adventure #20 — Danger Is My Business

So, you think you're tough and clever, eh? We'll find out soon enough. The American ambassador's daughter has been kidnapped and is somewhere in the jungles of India. You have to get her out. It may not sound too bad, but it's going to be the hardest thing you ever tried.

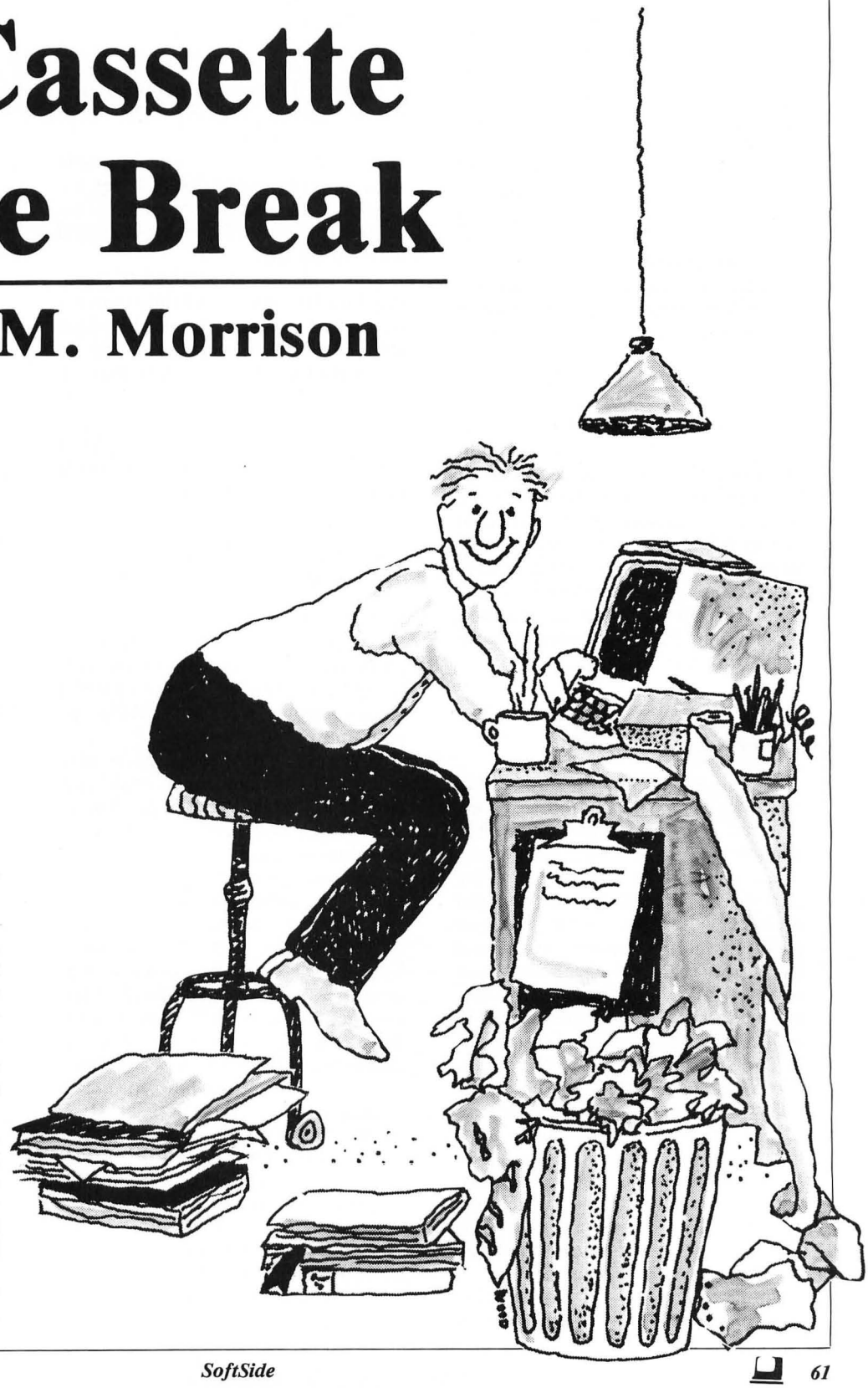
See the bind-in card facing this page to order your adventures.

Adventures Available:

#1 #2 #3	Arabian Adventure Alien Adventure Treasure Island	SuperDisk #1
#4 #5 #6	Jack the Ripper Crime Adventure Around the World In Eighty Days	SuperDisk #2
#7 #8 #9	Black Hole Adventure Windsloe Mansion Klondike Adventure	SuperDisk #3
#10 #11 #12	James Brand Adventure Witches' Brew Titanic Adventure	SuperDisk #4
#13 #14 #15	Arrow One Robin Hood The Mouse That Ate Chicago	SuperDisk #5
#16 #17 #18	Menagerie The Deadly Game The Dalton Gang	SuperDisk #6
#19	Alaskan Adventure	

The Cassette Coffee Break

by Charles M. Morrison



"I might as well get a cup of coffee. It's going to take at least five minutes to read in the data." How often have we cassette-equipped TRS-80® owners been frustrated by the slow rate of data transfer during input/output operations?

The most widely touted solution is to update the system to a disk configuration. However, the number of dollars involved in upgrading a system from tape to disk amounts to a significant figure that we may be unable or unwilling to spend.

Several techniques are available to improve effectiveness of cassette data transfer using hardware or assembler/Machine Language routines. This article is for those who neither have, nor care to develop, assembler language ability. It discusses programming methods you can apply using the inherent capabilities of BASIC, and tries to satisfy the needs of the small computer operator who wants more effective and efficient operation within his current cassette configuration.

With these criteria in mind, I will address several programming capabilities applicable to enhanced cassette operations. Various procedures for cassette input/output will be reviewed and analyzed, and possible software methods for attaining faster and more efficient data transfer to and from tape are

analyzed. I use a TRS-80 Model I, Level II, 16K system without disk at home. All routines and procedures discussed in this article were tested and verified on this system.

Cassette input and output is governed by the size of the input/output record and baud rate (the number of bits transferred per second). The Model I transfers data at a rate of 500 baud, which roughly equates to a theoretical maximum of 62 characters (or bytes) per second. The 1500 baud capability of the Model III equates to about 187 characters per second. For the remainder of this article I will refer to character transfer rather than bit or byte manipulation.

In addition to the rate of transfer, we are concerned with the quantity of meaningful data transferred when we read from or write to tape. We know that up to 248 characters may be transferred each time we execute a PRINT# or INPUT# statement. I will refer to this as an input/output record. Regardless of the number of data characters we send to the cassette, each execution causes a 256 character header to be written to the tape immediately before our data. Therefore it is very important to minimize the number of different PRINT# statements we use within a program.

A Serious Problem?

I analyzed my home budget accounting program, a modified version of a commercially available TRS-80 program. I input data semimonthly. During the data update, two write and read operations build and verify the data file. Sixty unique account numbers require sixty data item transfers for each read and write operation, and that means sixty times the 256 character leader is written to, or read from, cassette.

The program, as written, executed each of these transfers individually. In other words, before the program was modified, it took approximately four minutes and fifteen seconds to transfer transaction data between cassette and memory each time I used the input or output function — enough time for a coffee break!

The following short routine demonstrates this point. Sixty 3-digit numeric fields are transferred to cassette one field at a time:

```
100 X=150
110 FOR N=1 TO 60
120 X=X+2
130 PRINT#-1,X
140 NEXT N
```

Having defined the problem (too many coffee breaks), the next logical step is to resolve it. The obvious solution is to transfer more data (and correspondingly fewer unproductive leaders) each time we execute the program. You have several ways to do this.

One widely used method of packing data to reduce I/O time consists of specifying a series of data elements, each separated by a delimiting character (a comma), in each PRINT# and corresponding INPUT# statement. For example, you could transfer ten defined numeric values with the following record-packing statement:

```
830 PRINT#-1,A1,A2,A3,
A4,A5,A6,A7,A8,A9,A0
```

On execution, this statement will place ten data elements in the I/O record and transfer them to tape in a single output operation taking approximately four seconds.

Although this procedure is efficient and practical when packing a small number of items, it has limitations when you need to move large amounts of data.

The amount of data you can pack with this procedure reduces as a function of the number of fields defined in the PRINT# statement. Each of the delimiting commas that separate the individual fields takes up one character in the data record. Thus, the data transfer maximum of 248 reduces by the number of fields minus 1. For example, to define twenty fields in the PRINT# statement requires nineteen (20-1) characters for the delimiters, resulting in a maximum of 229 characters available for data transfer.

In the timing example, a single multiple PRINT# statement could not accomplish the transfer of the 60 data fields, as each delimiter would require one character space. To move all the data, we would need two PRINT# statements, each transferring part of the data.

PRINT# and INPUT# Relationships

Another characteristic associated with this procedure is the relationship between PRINT# statements and corresponding INPUT# statements. Radio Shack documentation states that: "The input list must be identical to the list that created the taped data-block (same number and type of variables in the same sequence)." This is not quite true.

If you have too few data items on the tape record generated by the PRINT# statement to satisfy the INPUT# statement, an "out of data" error will occur. If you have more data items on the tape record than are defined in the INPUT# statement, the program will read as many items as specified in the INPUT# statement, print an "extra ignored" warning, and continue processing. Be aware that the remaining data items on that tape record will be lost, as the next input command will read the next sequential record on the tape.

If the INPUT# statement requests a numeric data element which was output during the PRINT# operation as string data, an error will occur. The reverse, however, is not true. If the INPUT# statement requests a string variable from a data field output as numeric data, the system will accept the data as a string variable, even though it was output to tape as numeric data. The following short verification routine demonstrates and verifies this feature:

```
100 X=14250
110 PRINT#-1,X
120 PRINT"REWIND THE
TAPE, THEN TYPE
'CONT'"
130 STOP
140 INPUT#-1,P$
150 PRINT P$
160 Y=VAL(P$)
170 PRINT Y
```

During execution, the variable "X" was output with the PRINT#-1,X statement as a

numeric variable, and was subsequently read back into memory as a string variable with the INPUT#-1, P$ command. Remember that to be used in processing as a numeric variable, P$ must be converted back to numeric data using the VAL function.

One may also pack numeric data using the STR$ function. Using this system capability, the numeric fields are converted to string form using the STR$ function, concatenated, and output with a PRINT# statement. When the packed string is read back into memory with an INPUT# statement, the individual fields are broken out using the MID$ function and converted back into numeric form using the VAL function. The following code demonstrates this technique:

```
100 X=150
110 P$=""
120 FOR N=1 TO 10
130       X=X+50
140       S$=STR$(X)
150       P$=P$+S$
160 NEXT N
170 PRINT#-1,P$
180 PRINT"REWIND THE
TAPE, THEN TYPE
'CONT'"
190 STOP
200 P$=""
210 INPUT#-1,P$
220 FOR N=0 TO 9
230       K=N*4+1
240       S$=MID$(P$,K,4)
250       PRINT S$,
260       B1=VAL(S$)
270       PRINT B1
280 NEXT N
```

We have addressed speeding up data transfer by packing as much meaningful data as possible into each 248 character I/O record, thus reducing the number of leaders written to tape. To demonstrate the seriousness of the problem, we showed that it took more than four minutes to transfer sixty 3-digit variables to tape one item at a time. Applying the techniques we have discussed to pack a string for the transfer of the data, we can develop the following routine to accomplish the same task:

```
100 X=150:P$=""
110 FOR N=1 TO 60
120       X=X+2
130       S$=STR$(X)
140       P$=P$+S$
150 NEXT N
160 PRINT#-1,P$
```

With this short routine, the time required to transfer the data from the computer's memory to the cassette tape is reduced to slightly more than four seconds. This is a significant saving in time, and in tape.

When Field Length Varies

In our examples so far the multiple fields were of equal length so that, in the unpacking operation, the field length argument for the MID$ function was a fixed value. If the field length varied, but was known in advance, similar pack and unpack techniques could be applied. In this final discussion, we will address the application of these pack and unpack routines when length varies from field to field and is not known in advance.

We will define and build a small demonstration program. Our data fields will be defined as:

FILE # 7 CHAR	FA$
NAME 40 CHAR	NA$
CITY 10 CHAR	CI$

The sample data we will input at the keyboard for transfer to tape is:

FILE #	181246A
NAME	MORRISON
CITY	WEST ROXBURY

The file number exactly fits the defined field. The name is shorter than the defined field size and will have to be filled. The data for the city entry exceeds the size of the defined field and will be truncated before execution of the PRINT#

continued on page 66

AMDEK...your guide to

COLOR-II MONITOR
High resolution 560(H) X 240(V) • RGB video input • 80 X 24 character display capability • 16 color intensity modulation for IBM

COLOR-I MONITOR
260(H) X 300(V) line resolution • Built-in speaker & audio amplifier • front mounted controls

MODEL 310A MONITOR
Easy-view amber phosphor CRT • Composite video signal • 18 MH_z bandwidth • 900 lines (center) resolution IBM-PC compatible

MODEL DXY PLOTTER
Economical X-Y coordinate plotter • 10″ X 14″ plotting range • Centronics interface • ROM expandable • 4 pens, holders and chart hold-downs included

YOUR COMPATIBILITY CHART

MORE COMPATIBILITY INTERFACES DUE SOON . . . CHECK FACTORY!

COMPUTER	MONITORS							PLOTTER	3″ MICRO-DISC DRIVE	NOTES
	VIDEO-300	VIDEO-310	COLOR-I	COLOR-II	COLOR-IIA	COLOR-III	COLOR-IV			
IBM-PC	•	•	•	•	*	•	*	•	•	* Special Cabling Required
APPLE III	•			*	*	*	*	•		* Special Cabling or Converter Required
APPLE II	•		•	*	*	*	*	•	•	* DVM Board Required
ATARI 800	*		*							* Opt. Atari Cable Required
VIC-20			*							* Opt. VIC Cable Required
TRS-80	*									* Opt. TRS Cable Required
Osborne	*									* Opt. Interface Required
TI-99			*							* Opt. TI Cable Required
Commodore-64	*		*							* Opt. Commodore Cable Req.

innovative computing!

COLOR-III MONITOR

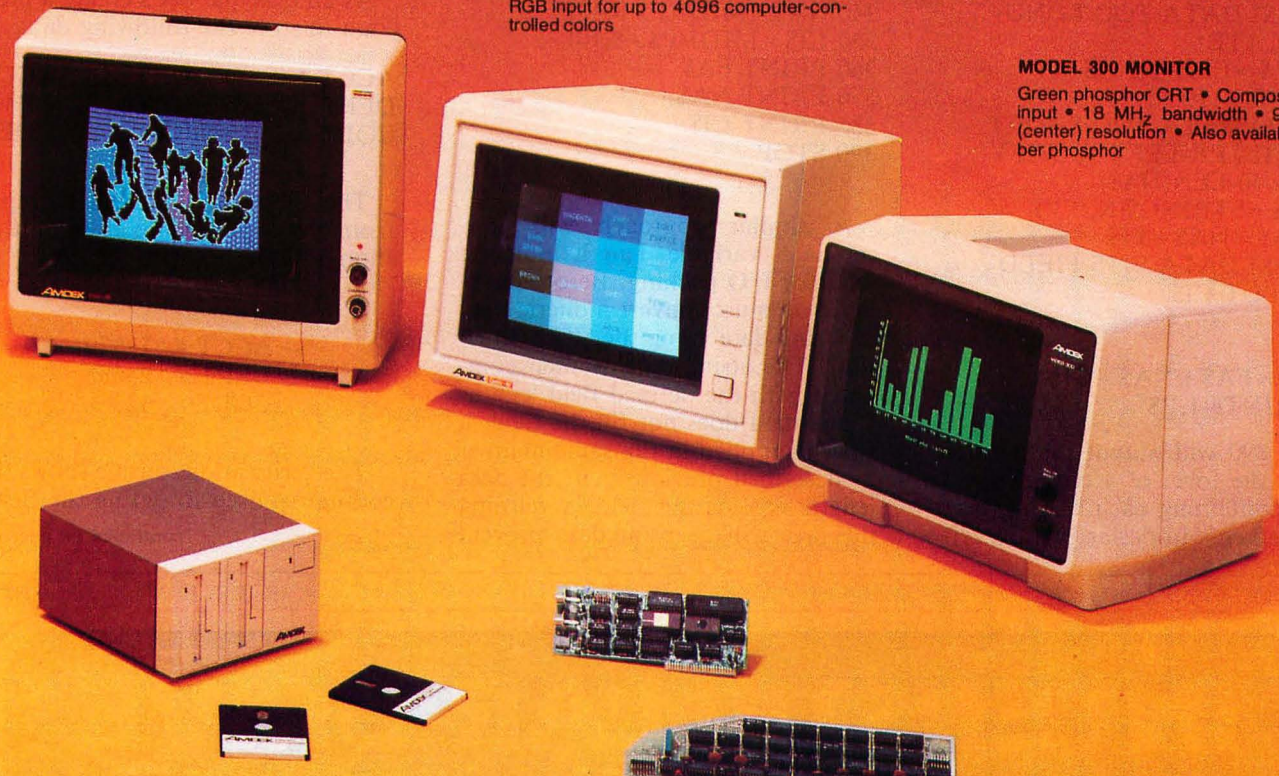
Economical RGB input monitor • 260(H) X 300(V) line resolution • 80 X 24 character display capability • Commercial grade CRT

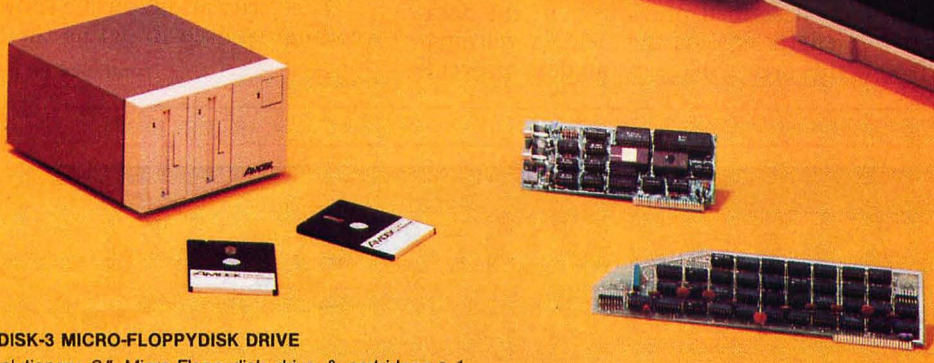
COLOR-IV MONITOR

Superior 720(H) X 420(V) line resolution • 80 X 24 character display • analog RGB input for up to 4096 computer-controlled colors

MODEL 300 MONITOR

Green phosphor CRT • Composite video input • 18 MH_z bandwidth • 900 lines (center) resolution • Also available in amber phosphor





AMDISK-3 MICRO-FLOPPYDISK DRIVE

Revolutionary 3″ Micro-Floppydisk drive & cartridges • 1 Megabyte (unformatted) storage • Track-to-track compatible with 5¼″ floppydisk drives • Built-in power supply • Hard plastic, protective floppydisk cartridge

COMBINATION DIGITAL VIDEO MULTIPLEXER

Makes Apple-II compatible with RGB monitors • Modes include Apple 40 character-line text, Apple high/low resolution color graphics, 80 character-line text from vendor board (included)

Amdek Corp. is dedicated to marketing quality computer peripheral equipment to enhance the use of popular personal computers. Our research & development staff keeps abreast of progress in computer techology and equipment and strives to offer you state-of-the-art advances in peripheral equipment.

Amdek products are distributed nationwide and in Canada through major distributors. And, we have factory-trained manufacturer's representatives ready to serve you in every major marketing area. Amdek offices are located in Chicago, Los Angeles & Dallas.

Just circle the reader service number, or contact us to receive complete technical specifications on these Amdek products.

2201 Lively Blvd. • Elk Grove Village, IL 60007
(312) 364-1180 TLX: 25-4786

AMDEK CORP.

The Cassette Coffee Break *continued*

command. The following extract shows the coding required to define these fields for output to the cassette:

```
100 INPUT NA$
110 IF LEN(NA$) > 40 THEN 140
120 IF LEN(NA$) < 40 THEN 160
130 GOTO 1190
139 REM DATA TOO LONG-
TRUNCATE
140 NA$ = LEFT$(NA$,40)
150 GOTO 1190
159 REM DATA TOO SHORT-
ADD FILLER-LEFT JUSTIFY
160 N = 40-LEN(NA$)
170 FI$ = STRING$(N," ")
180 NA$ = NA$ + FI$
189 REM DATA FITS FIELD
190 INPUT CI$
.
.
280 P$ = FA$ + NA$ + CI$
290 PRINT#-1,P$
```

Line 290 will output the data to tape in the form of a concatenated string containing each of the three data elements in defined fields, left justified. An INPUT# statement can be used to read the data back from cassette. Using the MID$ function, the user can then strip off the three fields of interest. The following routine shows how to accomplish this task:

```
320 INPUT#-1,P$
330 FA$ = MID$(P$,1,7)
340 NA$ = MID$(P$,8,40)
350 CI$ = MID$(P$,48,10)
360 PRINT FA$
370 PRINT NA$
380 PRINT CI$
390 STOP
```

A similar approach will concatenate variable length numeric data for I/O packing. Before packing, you must convert numeric data to string data using the STR$ function. The string data is then compared with the defined field length. If the data does not fill the field, a filler is applied in the same manner as described previously. If the data length exceeds the field, a warning message will be printed to prevent the processing of possibly erroneous data. When the concatenated string is read back into memory with the appropriate INPUT# statement, the fields are stripped out using the MID$ function. You must then convert the string data back to numeric data using the VAL function before attempting further processing of the data.

As an example, let's create a program to store the following data:

ACCT #	5 DIGITS	ACCT
DEPT	4 DIGITS	DPT
CODE	3 DIGITS	CDE

The following data will be input from the keyboard to exercise this short routine:

ACCT #	13579
DEPT	2468
CODE	123

The verification routine shows the coding required to define the fields

continued on page 69

INTERSTELLAR DRIVE™

A SOLID STATE DISK EMULATOR

Save valuable time!

5 to 50 times faster performance than floppy disks and Winchester drives

PION'S INTERSTELLAR DRIVE is designed for use with a family of interfaces and software packages. Currently available are interfaces for IBM, S100, TRS80, Apple, SS50, and most Z80 uP, and software for most popular operating systems. Additional interfaces are continually being developed for the most popular computers.

SAVE MONEY!
Increase your computer's productivity

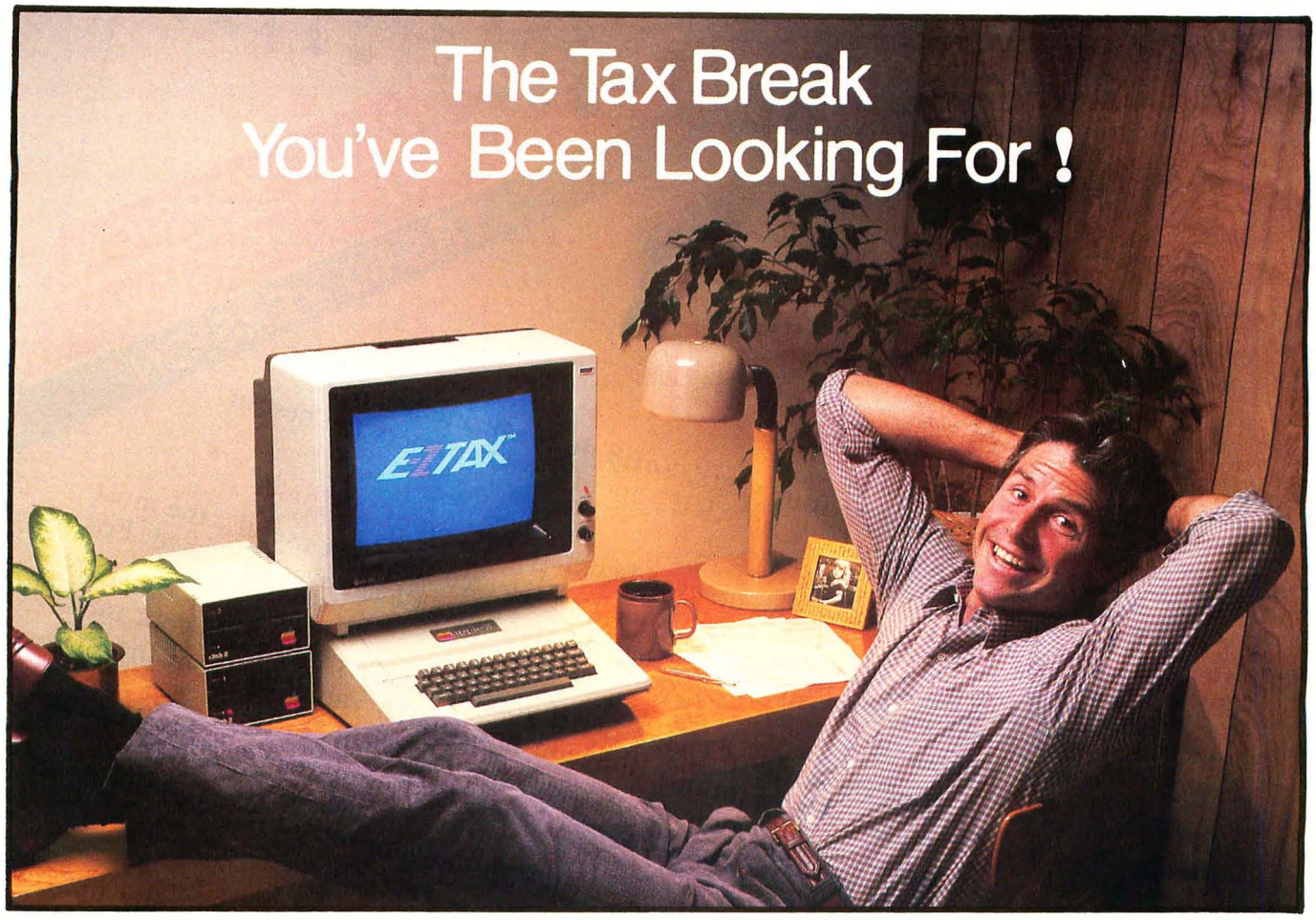
The INTERSTELLAR DRIVE is a high performance data storage subsystem with independent power supply, battery backup, and error detection. It has 256KB to 1 Megabyte of solid state memory integrated to perform with your operating system.

Basic Price for 256KB unit [includes interface and software]
$1095. *plus tax (where applicable) and shipping*
Visa and Master Card accepted.

PION, INC.
101R Walnut St., Watertown, MA 02172

TRS80 trademark of Tandy Corp. Apple trademark of Apple Computers
Interstellar Drive trademark of PION, Inc.

The Tax Break You've Been Looking For !



You Just Found It!

E-Z Tax. The simplest tax preparation software ever developed was designed for your Apple II personal computer.

Now you can prepare your own tax return without **any** knowledge of taxes or computer programming. From the moment you insert the E-Z Tax floppy disk, you'll be in full control. Every question is self-prompting and nothing is overlooked.

If you make a mistake, the program lets you know about it immediately. If you need tax help, just press a button and you'll get the answer. *Its simply the most amazing tax preparation software ever.*

COUPON

Please send me the following # of kits requested:

_____ APPLE II _____ IBM PC

_____ ATARI 400 & 800 _____ CP/M

_____ **TOTAL REQUESTED**

x $69.95 each

_____ Total

_____ Plus Postage & Handling ($4/kit)

_____ Plus C.O.D. Charges ($3/kit)

_____ **TOTAL ORDER**
(Enclose payment for this amount.)

ACT NOW!

Send: ☐ Check ☐ Money Order ☐ C.O.D.
Charge my credit card: ☐ Visa ☐ Mastercard

Card #_____ Exp. Date _____

Signature_____

Name_____

Address_____

City_____ State _____ Zip _____

Mail this coupon to: **TAX HELP, INC.**

Prints on Federal Forms

When you're finished, E-Z Tax will print out your tax return on official federal forms. If you don't have a printer, just fill in the forms from the data on the screen.

If you need help, you can call E-Z Tax's toll free customer service phone number.

E-Z Tax prepares the following IRS forms and schedules:

1040A	2106
1040 EZ	2119
1040 page 1 & 2	2210
Schedule A	2440
Schedule B	2441
Schedule C	3468
Schedule D	3903
Schedule E	4137
Schedule F	4684
Schedule G	4972
Schedule R/RP	5695
Schedule W	6251
1040 ES	6252
1040 SE	

ACT NOW!

You just found the tax preparation program you've been looking for. Now here's how you can get your hands on it . . .

- Fill in the coupon, or
- Call toll-free to order over the phone. Just give the operator your credit card number or request a C.O.D. shipment.

Only **$69.95**
TAX DEDUCTIBLE

E-Z TAX

Your E-Z Tax Kit Includes. . .

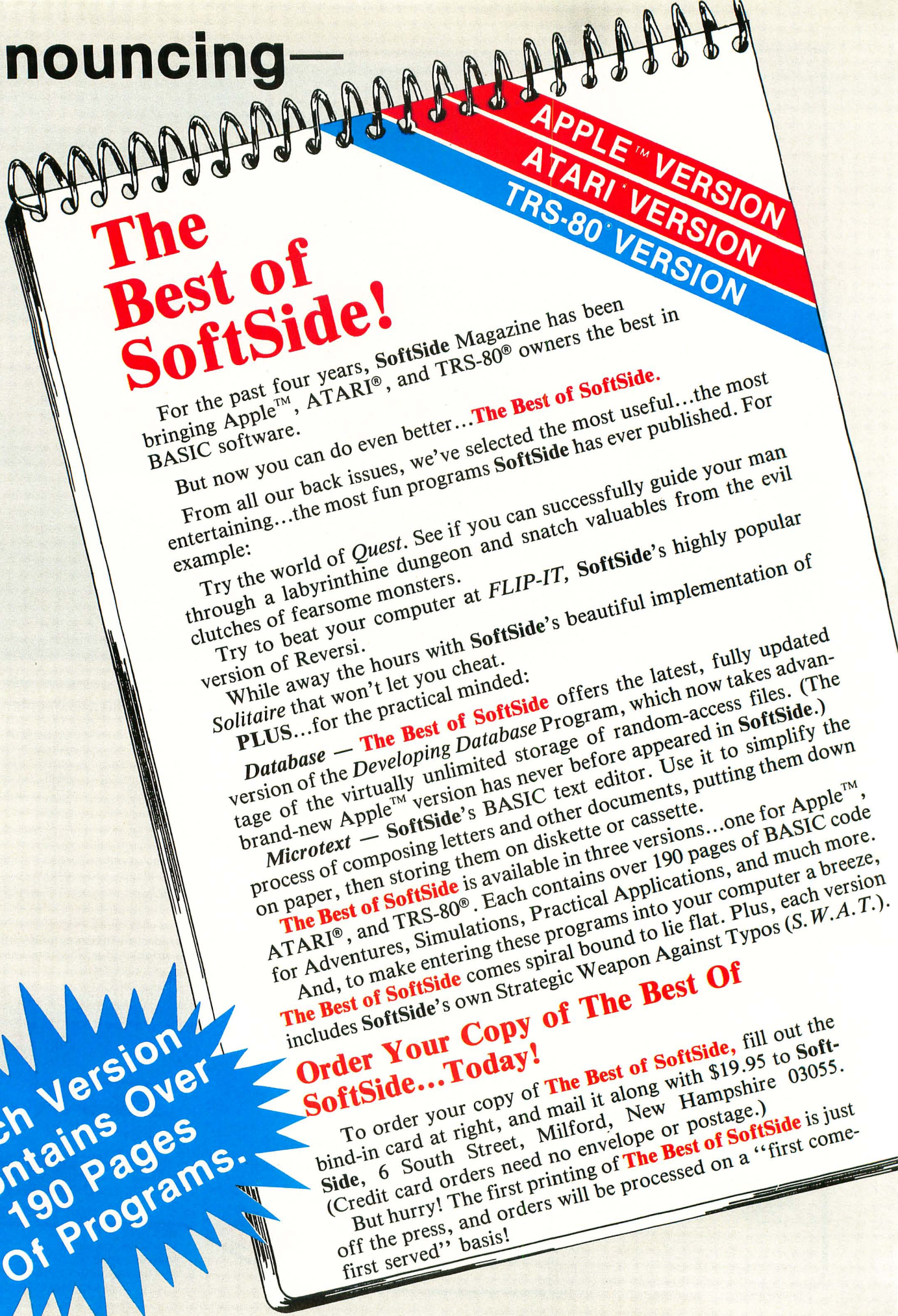
- E-Z Tax Software Program (2 Disks)
- E-Z Tax Guide Book
- Over 35 Official Federal Tax Forms for 1982 Tax Returns
- Tax Organizer Envelopes
- Instruction Guide
- Warranty Card

Distributed By

TAX HELP INCORPORATED

BOX 7676
SAN JOSE, CA 95150
(408) 998-1040
WATS LINE: (800) 331-1040 - USA
(800) 344-1040 - CA

Announcing—



The Best of SoftSide!

For the past four years, **SoftSide** Magazine has been bringing Apple™, ATARI®, and TRS-80® owners the best in BASIC software.

But now you can do even better…**The Best of SoftSide.**

From all our back issues, we've selected the most useful…the most entertaining…the most fun programs **SoftSide** has ever published. For example:

Try the world of *Quest*. See if you can successfully guide your man through a labyrinthine dungeon and snatch valuables from the evil clutches of fearsome monsters.

Try to beat your computer at *FLIP-IT*, **SoftSide**'s highly popular version of Reversi.

While away the hours with **SoftSide**'s beautiful implementation of *Solitaire* that won't let you cheat.

PLUS…for the practical minded:

Database **— The Best of SoftSide** offers the latest, fully updated version of the *Developing Database* Program, which now takes advantage of the virtually unlimited storage of random-access files. (The brand-new Apple™ version has never before appeared in **SoftSide**.)

Microtext **— SoftSide**'s BASIC text editor. Use it to simplify the process of composing letters and other documents, putting them down on paper, then storing them on diskette or cassette.

The Best of SoftSide is available in three versions…one for Apple™, ATARI®, and TRS-80®. Each contains over 190 pages of BASIC code for Adventures, Simulations, Practical Applications, and much more.

And, to make entering these programs into your computer a breeze, **The Best of SoftSide** comes spiral bound to lie flat. Plus, each version includes **SoftSide**'s own Strategic Weapon Against Typos (*S.W.A.T.*).

Order Your Copy of The Best Of SoftSide…Today!

To order your copy of **The Best of SoftSide,** fill out the bind-in card at right, and mail it along with $19.95 to **SoftSide**, 6 South Street, Milford, New Hampshire 03055. (Credit card orders need no envelope or postage.)

But hurry! The first printing of **The Best of SoftSide** is just off the press, and orders will be processed on a "first come-first served" basis!

Each Version Contains Over 190 Pages Of Programs.

The Cassette Coffee Break ***continued***

for output to cassette, and the code used to read the data back from cassette:

```
100 INPUT ACCT
110 AC$ = STR$(ACCT)
120 IF LEN(AC$)#6 THEN GOTO
9000:'WARNING ROUTINE
130 IF LEN(AC$) = 6 THEN 170
140 N = 6-LEN(AC$)
150 FI$ = STRING$(N," ")
160 AC$ = AC$ + FI$
170 INPUT DPT
.
.

300 INPUT#-1,P$
310 T$ = MID$(P$,1,6)
320 ACCT = VAL(T$)
.
.
```

The statement at line 120 branches to statement 9000, an error handling routine that determines the action to take when a numeric entry exceeds the predefined field size.

The comparator at line 120 for the comparison of the LEN value of AC$ is set to 6, one greater than the predefined length of the ACCOUNT field. This is necessary since the STR$ function used at line 110 will pick up a character for the leading blank of a positive numeric value.

Other Applications

A number of commercially produced educational programs include their data internally in DATA statements. During execution, READ statements pick up the necessary data for the problem being worked.

The disadvantage to this approach is that once the child becomes familiar with the data, you must change the program to change the data, and only the new data can be used. I have converted several of these programs to use an INPUT# statement to read packed data from a cassette generated with a short utility routine. This permits us to offer the student a choice of problem sets each time the program is used, significantly increasing its flexibility and learning value.

In summary, cassette I/O is linked to a data transfer rate dependent upon computer system characteristics over which we exert no control. For the TRS-80, this consists of a 256-character leader written or read with each PRINT# and INPUT# statement at either 500 or 1500 bits per second. Certain programming techniques can provide more efficient cassette data transfer. Several methods to accomplish this task have been described. No single procedure is recommended. Rather, you should use the approach that best suits the application at hand. To achieve maximum operation efficiency, you must apply programming techniques to insure that the maximum amount of meaningful data is transferred with each I/O operation, reducing input and output processing time. This should produce significantly fewer cassette coffee breaks!

SOFTSIDE ORDERING INFORMATION

FORM OF PAYMENT

USA

VISA, MasterCard, certified checks, money orders and personal checks are accepted.

Canada/Mexico

The preferred method of payment is by VISA or MasterCard. A bank check is acceptable if it has been preprinted for payment in U.S. dollars. No personal or company checks accepted.

Other Foreign Orders

Payment must either be by a bank check drawn on a U.S. bank payable in U.S. dollars or by affiliated bank credit cards of VISA or MasterCard.

GUARANTEE

All software is guaranteed to load and run. If you experience difficulties with the product within 30 days, it may be returned for replacement. Send your properly protected tape or disk to the attention of the Customer Service Representative and include your name, address, and the reason it is being returned.

LIABILITY

All software is sold on an as-is basis. **SoftSide** assumes no liability for loss or damage caused or alleged to be caused directly or indirectly by products sold or exchanged by them or their distributors, including, but not limited to, any interruption in service, loss of business or anticipatory profits or consequential damages resulting from use or operation of such software.

PRICES

Prices are subject to change without notice. We are not responsible for typographical errors.

Unless otherwise noted in a published advertisement, the following prices are in effect as of this issue:

	USA/Canada APO/FPO	USA/Canada FIRST CLASS Mexico	Other Foreign
SoftSide Magazine (yr)	$30	$40	$62
SoftSide Magazine (6 mo.)	**$15**	**$20**	**$31**

	USA APO/FPO	Mexico Canada	Other Foreign
CV (year) & magazine	$75	$95	$125
(6 mo.)	$40	$50	$70
DV (year) & magazine	$125	$145	$175
(6 mo.)	$70	$80	$90

BACK ISSUES

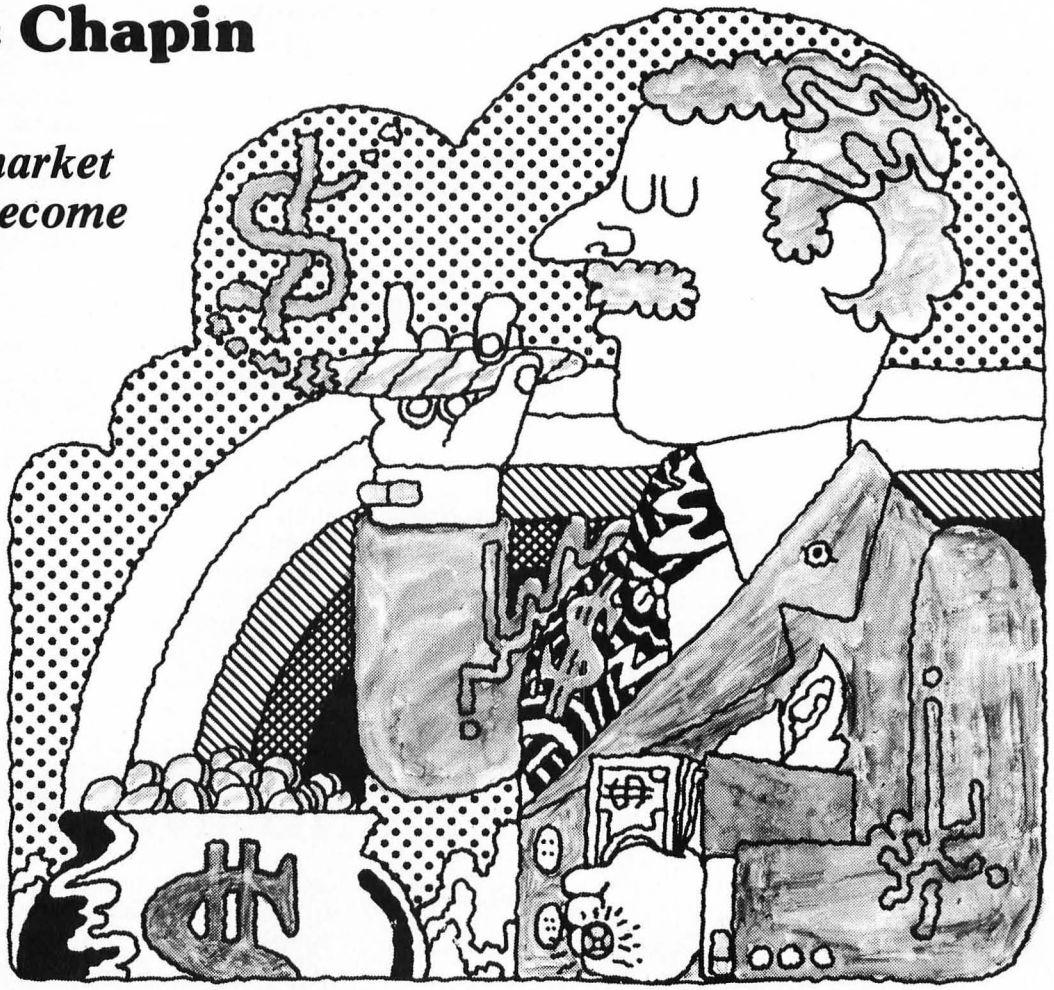
Minimum order — $10. Price includes shipping to the 48 states only. Alaska, Hawaii, Puerto Rico, APO/FPO, and ALL foreign orders — postage is additional.

ALL Foreign orders and all magazine/media combination orders — Order directly from **SoftSide, 6 South St., Milford, NH 03055.**

GARAGE SALE RECORDS

by Ernie Chapin

Corner the market on junk — become a garage sale magnate.



Garage Sale Records **is a home finance program for a TRS-80® Model I/III with 32K RAM and disk. Changes are provided to make the program work with a 16K tape system.**

The garage sale is a popular way to sell unwanted items. At many sales, several people are selling items, and keeping track of each person's sales can be confusing. With this program, ten sellers can keep track of up to 75 items each. You can adjust this limit upward by changing the DIMension statements in lines 30 and 40.

When the program starts and the menu appears, enter "1" to create the list of sellers. Enter a unique name for each seller. The computer uses the names to compare later sales so be sure no two names are the same when entering.

After you enter the names, you are ready to start the sale. Type "2", and when prompted, enter enough of the seller's name to identify him or her. You only need one letter if no other names start with the same letter. However, if "TOM" and "TONY" are both sellers, then "TOM" or "TON" would be needed to identify the name. Other entries are self-explanatory.

The program is currently set up to use drive 1 to hold the garage sale data. You may change this by modifying lines 1100 and 1300.

Variables

A$: Program name.
B$: Author's name.
C(N): Array to count items sold by each seller.
GT: Grand total for the sale.
P(N,C(N)): Price of item sold.
NS: Number of sellers.
N$: Input for name of seller at time of sale.
S$(X): Sellers' names.
T$(N,C(N)): Name of items sold.
TT: Running total for each buyer.
T(N): Each seller's total for the sale.
TD & DD: Time delay loop variables.
XX,YY: Screen positions.
X,Q,A: Loops.
Z,M,N: User input replies.

```
SS SS SS SS SS SS SS SS SS SS SS SS
SS                                SS
SS       TRS-80 DISK BASIC        SS
SS     'GARAGE SALE RECORDS'      SS
SS     AUTHOR: ERNIE CHAPIN       SS
SS      COPYRIGHT (C) 1982        SS
SS  SOFTSIDE PUBLICATIONS, INC.   SS
SS                                SS
SS SS SS SS SS SS SS SS SS SS SS SS
```

If you don't wish to type this program, it is also included in this month's SoftSide CV and DV.

Clear string space and initialize variables.

```
10 CLS
20 CLEAR 7500: GT=0:NS=0
30 DIM S$(10),T$(10,75)
40 DIM T(10),P(10,75)
```

Introduce the program.

```
50 A$="GARAGE SALE RECORDS"
60 B$="CREATED BY ERNIE CHAPIN WOODBURN, ORE.  1982"
70 PRINT@33-(LEN(A$)/2),A$
80 PRINT:PRINT@33-(LEN(B$)/2)+64,B$
```

Set-up the menu.

```
90 PRINT:PRINT"TYPE 1 TO ENTER THE LIST OF SELLERS"
100 PRINT"TYPE 2 TO ENTER A SALE"
110 PRINT"TYPE 3 TO REVIEW AN INDIVIDUALS SALE RECORD"
120 PRINT"TYPE 4 TO SEE TOTALS FOR THE SALE"
130 PRINT"TYPE 5 TO CHANGE OR ADD A NEW NAME"
140 PRINT"TYPE 6 TO SAVE THE DATA ON DISK"
150 PRINT"TYPE 7 TO RETRIEVE DATA FROM DISK"
160 PRINT"TYPE 8 TO CHECK SELLERS NUMBERS"
170 PRINT"TYPE 9 TO PRINT INDIVIDUAL REPORTS ON PAPER"
180 INPUT M: IF M>9 OR M<1 THEN CLS: GOTO 90
190 ON M GOTO 200,300,500,700,800,1100,1300,1500,1600
```

Input names of sellers.

```
200 CLS
210 NS=NS+1
220 PRINT"TYPE 'END' WHEN LIST IS COMPLETE"
230 PRINT
240 PRINT"ENTER THE NAME OF SELLER #"NS
250 INPUT S$(NS)
260 IF S$(NS)="END" THEN NS=NS-1:CLS:GOTO 90
270 GOTO 200
```

Enter a sale.

```
300 CLS:TT=0
310 XX=211:YY=236
320 CLS:PRINT"WHEN ENTERING SELLER, USE ENOUGH OF NAME TO IDENTI
FY FROM OTHER SELLERS. ENTER 'TOTAL' FOR NAME WHEN CURRENT BUYER
 IS FINISHED"
330 PRINT"SELLER"TAB(20)"ITEM"TAB(45)"PRICE"
```

Get name of seller.

```
340 PRINT@XX-19,"";:INPUT N$:FOR X=1 TO NS:IF N$=LEFT$(S$(X),LEN
(N$))THEN PRINT@XX-19,STRING$(15,32):PRINT@XX-19,S$(X) ELSE NEXT
 X
```

If name input is "TOTAL" then wait for input, otherwise get name and price of the item sold.

```
350 IF N$="TOTAL" THEN PRINT@XX-19,STRING$(20,32):PRINT TAB(25)"
TOTAL"TAB(44)USING"$###.##";TT:GOSUB 1000 ELSE 370
360 GOTO 460
370 PRINT@XX,STRING$(40,32):FOR X=1 TO NS+1
380 IF N$=LEFT$(S$(X),LEN(N$)) THEN N=X ELSE NEXT X:IF X>NS THEN
 450
390 C(N)=C(N)+1
400 PRINT@XX,"";:INPUT T$(N,C(N)):PRINT@YY,"";:INPUT P(N,C(N)):P
RINT@YY,USING"$###.##";P(N,C(N))
410 XX=XX+64:YY=YY+64
420 J=P(N,C(N)): T(N)=T(N)+J: GT=GT+J: TT=TT+J
430 PRINT@ XX+64,"TOTAL";:PRINT@YY+64,USING"$###.##";TT
```

Go back and get another sale.

```
440 GOTO 340
```

If the name is not in the list of sellers, issue a warning message and try again.

```
450 FOR TD=1 TO 5: PRINT@978,"INVALID SELLER--TRY AGAIN";:FOR DD
=1 TO 80:NEXT DD:PRINT@960,STRING$(60,32);:FOR DD=1 TO 50:NEXT D
D:NEXT TD:GOTO 340
```

Check for another sale.

```
460 CLS: INPUT"DO YOU HAVE ANOTHER SALE TO ENTER";Z$
470 IF LEFT$(Z$,1)="Y" THEN 300 ELSE CLS: GOTO 90
```

List sellers' names on screen.

```
500 CLS:FOR X=1 TO NS
510 PRINT X"---"S$(X):NEXT X
520 PRINT@832,"";:INPUT "WHAT IS THE NUMBER OF THE SELLER";N
```

Check for incorrect input.

```
530 IF N<1 THEN GOTO 520
540 IF N>NS THEN PRINT"WE DON'T HAVE THAT MANY SELLERS": GOTO 52
0
```

Print the list of items and prices for one seller onto the screen.

```
550 CLS:PRINT"HIT 'SHIFT' AND '@' KEYS TO STOP LIST AND ANY KEY
TO CONTINUE
560 PRINT:FOR X=1 TO C(N)
570 PRINT T$(N,X) TAB(24)USING"$###.##";P(N,X)
580 NEXT X
590 PRINT:PRINT"TOTAL" TAB(23)USING"$####.##";T(N)
```

Check for another seller.

```
600 PRINT@896,"": INPUT"DO YOU WANT TO CHECK ANOTHER SELLER";Z$
610 IF LEFT$(Z$,1)="Y" THEN 500 ELSE CLS:GOTO 90
```

Print seller's totals and sale total.

```
700 CLS
710 PRINT" NAME" TAB(30) "SOLD"
720 FOR X=1 TO NS
730 PRINTS$(X);: PRINT TAB(28)USING"$$###.##";T(X)
740 NEXT X
750 PRINT: PRINT"GRAND TOTAL" TAB(28)USING"$####.##";GT
760 GOSUB 1000
770 GOTO 90
```

Routine to add or change names.

```
800 CLS
810 PRINT"TYPE 1 TO ADD A NEW NAME"
820 PRINT"TYPE 2 TO CHANGE A NAME"
```

```
830 INPUT M: ON M GOTO 840,890
840 NS=NS+1:CLS
850 INPUT "WHAT IS THE NEW SELLERS NAME";S$(NS)
860 INPUT "DO YOU HAVE ANOTHER NAME";Z$
870 IF LEFT$(Z$,1)="Y" THEN 840 ELSE CLS
880 GOTO 90
890 CLS:INPUT "WHAT IS THE NUMBER OF THE NAME TO CHANGE";M
900 INPUT"WHAT IS THE CORRECT NAME";S$(M)
910 PRINT:INPUT"DO YOU WISH TO CHANGE ANY MORE";Z$
920 IF LEFT$(Z$,1)="Y" THEN 890 ELSE CLS:GOTO 90
```

Subroutine to wait for input.

```
1000 PRINT@980,"";:INPUT"PRESS -ENTER- TO CONTINUE";M
1010 CLS:RETURN
```

Save data on disk.

```
1100 CLS:OPEN "O",1,"SALES:1"
1110 PRINT "COPYING--"
1120 PRINT #1,NS
1130 FOR X=1 TO NS:PRINT#1,S$(X);",";T(X);",";C(X)
1140 FOR Q=1 TO C(X):PRINT#1,T$(X,Q);",";P(X,Q)
1150 NEXT Q
1160 NEXT X
1170 PRINT #1,GT
1180 CLOSE 1
1190 PRINT"TRANSFER IS COMPLETE---"
1200 GOSUB 1000
1210 GOTO 90
```

Routine to transfer data from disk to memory.

```
1300 CLS:OPEN"I",1,"SALES:1"
1310 PRINT"INPUTING---"
1320 INPUT #1, NS
1330 FOR X=1 TO NS:INPUT#1,S$(X),T(X),C(X)
1340 FOR Q=1 TO C(X):INPUT#1,T$(X,Q),P(X,Q)
1350 NEXT Q
1360 NEXT X
1370 INPUT#1,GT
1380 CLOSE 1
1390 PRINT"TRANSFER IS COMPLETE---":GOSUB 1000
1400 CLS:GOTO 90
```

List all names on screen.

```
1500 CLS:FOR X=1 TO NS
1510 PRINT"#"X"---";S$(X)
1520 NEXT X
1530 GOSUB 1000
1540 CLS:GOTO 90
```

Prints hardcopy of each seller's record.

```
1600 CLS: PRINT"PREPARE PRINTER FOR INDIVIDUAL REPORTS ON SALES"
1610 PRINT
1620 FOR X=1 TO NS
1630 INPUT"HIT ENTER WHEN READY FOR NEXT PAGE";Z
1640 LPRINT TAB(25)S$(X)
1650 LPRINT TAB(15)STRING$(30,45):LPRINT " ":FOR Q=1 TO C(X)
1660 LPRINT TAB(15)T$(X,Q) TAB(38)USING"$###.##"; P(X,Q)
1670 NEXT Q
1680 LPRINT:LPRINT TAB(37)STRING$(9,45)
1690 LPRINT TAB(25)"TOTALS"TAB(38)USING"$###.##";T(X)
1710 NEXT X
1720 CLS:GOTO 90
```

TRS-80® SWAT TABLE FOR:
GARAGE SALE RECORDS — DISK VERSION

LINES	SWAT CODE	LENGTH
10 - 120	PQ	371
130 - 240	HN	393
250 - 380	XP	522
390 - 520	WV	498
530 - 720	II	379
730 - 860	JL	291
870 - 1130	BH	356
1140 - 1330	DW	217
1340 - 1540	ZB	188
1600 - 1720	IH	302

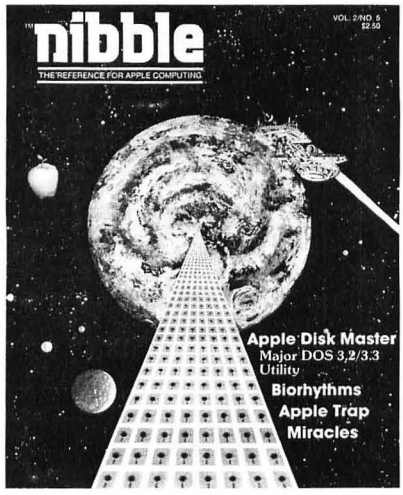
```
SS SS SS SS SS SS SS SS SS SS SS SS
SS                                SS
SS       TRS-80 TAPE BASIC        SS
SS 'GARAGE SALE -- TAPE CHANGES' SS
SS     AUTHOR: ERNIE CHAPIN       SS
SS      COPYRIGHT (C) 1982        SS
SS  SOFTSIDE PUBLICATIONS, INC.   SS
SS                                SS
SS SS SS SS SS SS SS SS SS SS SS SS
```

```
140 PRINT"TYPE 6 TO SAVE THE DATA ON CASSETTE TAPE"
150 PRINT"TYPE 7 TO RETRIEVE DATA FROM CASSETTE TAPE"
1100 CLS:PRINT "INSERT A NEW TAPE AND PRESS 'RECORD' & 'PLAY' ON
 RECORDER.  NOTETAPE COUNTER NUMBER.":GOSUB1000
1120 PRINT #-1,NS
1130 FOR X=1 TO NS:PRINT#-1,S$(X);",";T(X);",";C(X)
1140 FOR Q=1 TO C(X):PRINT#-1,T$(X,Q);",";P(X,Q)
1170 PRINT #-1,GT
1180 PRINT:PRINT
1300 CLS:PRINT"INSERT DATA CASSETTE AND REWIND/ADVANCE TO PROPER
 TAPE COUNTER  POSITION (BEGINNING OF DATA FILE).  PRESS 'PLAY'
ON RECORDER":GOSUB1000
1320 INPUT #-1, NS
1330 FOR X=1 TO NS:INPUT#-1,S$(X),T(X),C(X)
1340 FOR Q=1 TO C(X):INPUT#-1,T$(X,Q),P(X,Q)
1370 INPUT#-1,GT
1380 PRINT:PRINT
```

TRS-80® SWAT TABLE FOR:
GARAGE SALE RECORDS — TAPE VERSION

LINES	SWAT CODE	LENGTH
10 - 120	PQ	371
130 - 240	EJ	411
250 - 380	XP	522
390 - 520	WV	498
530 - 720	II	379
730 - 860	JL	291
870 - 1130	ID	435
1140 - 1330	KT	338
1340 - 1540	VI	190
1600 - 1720	IH	302

"NIBBLE® IS TERRIFIC" (For Your Apple)



NIBBLE IS: *The Reference for Apple computing!*

NIBBLE IS: One of the Fastest Growing new Magazines in the Personal Computing Field.

NIBBLE IS: Providing Comprehensive, Useful and Instructive Programs for the Home, Small Business, and Entertainment.

NIBBLE IS: A Reference to Graphics, Games, Systems Programming Tips, Product News and Reviews, Hardware Construction Projects, and a host of other features.

NIBBLE IS: A magazine suitable for both the Beginner and the Advanced Programmer.

Each issue of NIBBLE features significant new Programs of Commercial Quality. Here's what some of our Readers say:

- *"Certainly the best magazine on the Apple II"*
- *"Programs remarkably easy to enter"*
- *"Stimulating and Informative; So much so that this is the first computer magazine I've subscribed to!"*
- *"Impressed with the quality and content."*
- *"NIBBLE IS TERRIFIC!"*

In coming issues, look for:

☐ Stocks and Commodities Charting ☐ Assembly Language Programming Column
☐ Pascal Programming Column ☐ Data Base Programs for Home and Business
☐ Personal Investment Analysis ☐ Electronic Secretary for Time Management
☐ The GIZMO Business Simulation Game

And many many more!

NIBBLE is focused completely on the Apple Computer systems.

Buy NIBBLE through your local Apple Dealer or subscribe now with the coupon below.

Try a NIBBLE!

nibble

We accept Master Charge & Visa

Box 325, Lincoln, MA. 01773 (617) 259-9710
I'll try nibble!
Enclosed is my $19.95 (for 8 issues)
(Outside U.S., see special note on this page.)
☐ **check** ☐ **money order**

Your subscription will begin with the next issue published after receipt of your check/money order.

Card # _____ Expires _____
Signature _____
Name _____
Address _____
City _____
State _____ Zip _____

NOTE:

- Domestic U.S. First Class subscription rate is $36.50
- Canada Air Mail subscription rate is $42.50
- Outside the U.S. and Canada Air mail subscription rate is $47.50

All payments must be in U.S. funds drawn on a U.S. bank.

©1980 by MICRO-SPARC., INC. Lincoln, Mass. 01773. All rights reserved.
*Apple is a registered trademark of Apple Computer Company.

Scriptr and Crayon

Reviewed by Tim Knight

by Jerry Goodwin (Pioneer Software, 1746 N.W. 55th Avenue, Apt. 204, Lauderhill, FL 33313). *Scriptr* system requirements — TRS-80® Model I or III, 32K RAM. Suggested retail price $40.00. *Crayon* system requirements — TRS-80 Model I or III, 32K RAM, disk; 16K RAM, tape. Suggested retail price disk, $45.00; cassette, $35.00.

I have always liked the *Scripsit* word processor from Radio Shack. However, it lacks some features I use with my Epson printer, such as different printing modes, underlining, italicizing, and so on. Several *Scripsit* modification programs allow you to do those things, but too often they are poorly documented, poorly written, or both. Fortunately, you now have a *Scripsit* modification program from Pioneer Software which enhances *Scripsit* considerably.

Scriptr

I was attracted to *Scriptr* by its long list of features. I have used it for some time, but I continue to discover new features which increase the power of my word processor still more.

The *Scriptr* package includes a disk full of programs — some demonstration programs and some utilities in addition to a registration card and 60 pages of documentation. The program may also be purchased to modify tape *Scripsit.*

One of *Scriptr's* most important features is the format line which allows you to change the print *font* (or type). For example, you might want to underline a certain word in a sentence, put another word in

"One of Scriptr's most important features is the format line which allows you to change the print font (or type)."

italics, make the last two sentences of a paragraph extra bold, and then make the first word of every paragraph extra large. Using only *Scripsit*, this would be nearly impossible. With *Scriptr*, all of these things are easy. *Scriptr* can do it *all* and still justify text.

Other *Scriptr* features which make *Scripsit* easier are available in the display mode, the pause mode, and the edit mode of the program. Some of the commands from these three modes are:

1. Slow down the rate at which the print-out is displayed on the screen to improve readability.
2. Halt execution of the print-out (for changing paper or ribbons).
3. Load and print a graphics picture from disk.
4. Enter graphics from the keyboard to be printed onto paper.

Of those four, I especially appreciate the power to construct graphics directly within the word processor, and the ability to print them out on paper later.

Disk users have commands which allow you to go to and from the DOS. For instance, if you type something and for some reason want to see a disk directory, you can exit *Scripsit*, get a directory, and return to *Scripsit* with the text still intact. Once, while typing in a long article, I hit the reset key by mistake. I thought my text was long gone. *Scriptr*, however, had kept the entire text within memory, and when I loaded in *Scripsit*, the text was automatically loaded in also. *Scriptr* can be a *life-saver* at times.

Macros are also available with *Scriptr*. A macro is a series of statements you are likely to use frequently. They are compressed into a single command and may be called up at any time. Macros are convenient, especially when writing something which frequently calls for a certain type of printing.

Jerry Goodwin, the author of *Scriptr*, has made additional commands for Graphtrax owners. Because I have a Graphtrax in my Epson printer, I can print out *italicized* letters along with several other special type fonts. Slashed characters and underlining are also possible with *Scriptr* and Graphtrax, in addition to the ability to reset the printer.

Epson printer owners are not the only ones who can use this program, though. Programs are provided for the Microline printer as well. If you don't own either of these, but would still like to use *Scriptr*, Jerry Goodwin will create *custom programs* for $25.00.

Scriptr is an invaluable tool, and certainly worth the modest price. It is a program I will use for some time, and I recommend it to anyone who wants more power from *Scripsit*.

"I especially appreciate the power to construct graphics directly within the word processor, and the ability to print them out on paper later."

Crayon

Crayon is a counterpart to *Scriptr*, but still an independent program in itself. As the name suggests, this program allows a TRS-80 owner to draw pictures directly on the screen using simple commands. This multi-faceted program has many other features, though. The three modes are graphics mode, letter mode, and command mode.

The graphics mode allows you to draw anything you wish on the video screen, using a cursor to move about, and pressing letters and numbers to light up any of the 6,144 pixels (graphic dots). A total of sixteen commands light up different pixels, which makes creating a picture efficient and easy.

The cursor itself may be turned on or off, and may even change. Many special graphics functions, available in the graphics mode, allow you to center graphic lines automatically, exchange the screen with other screens (called *buffers*), and manipulate the lines in other ways. Graphics manipulation includes deletion, insertion, justification (just like *Scripsit*), exchanging, inversion (turning all the on-pixels to off, and vice versa), and other powerful and useful controls.

The letter command mode is similar to the graphics mode, and contains many of the same commands. However, the letter mode enters text rather than graphics. This is useful for things such as banners, which need to look impressive, but still require text.

The command mode is the real heart of this program. It is like a graphic DOS, since it allows you to manipulate file data with the graphics files. Pictures may be saved to disk (in just about any format), printed, killed, appended, and so on. You can alter these graphic files as easily as any other file. You can even view the disk buffer (storage area) and number the screens. This filing system is so efficient that you can create cartoons with it. A sample cartoon is included in the *Crayon* package, running about 35 frames every second.

Crayon also works with the *Scriptr* program, because picture files may be inserted directly into *Scriptr*. You can create a picture using *Crayon*, save it to disk, then retrieve it and print it later with *Scriptr*. This is particularly nice if you want more graphics in your word processing.

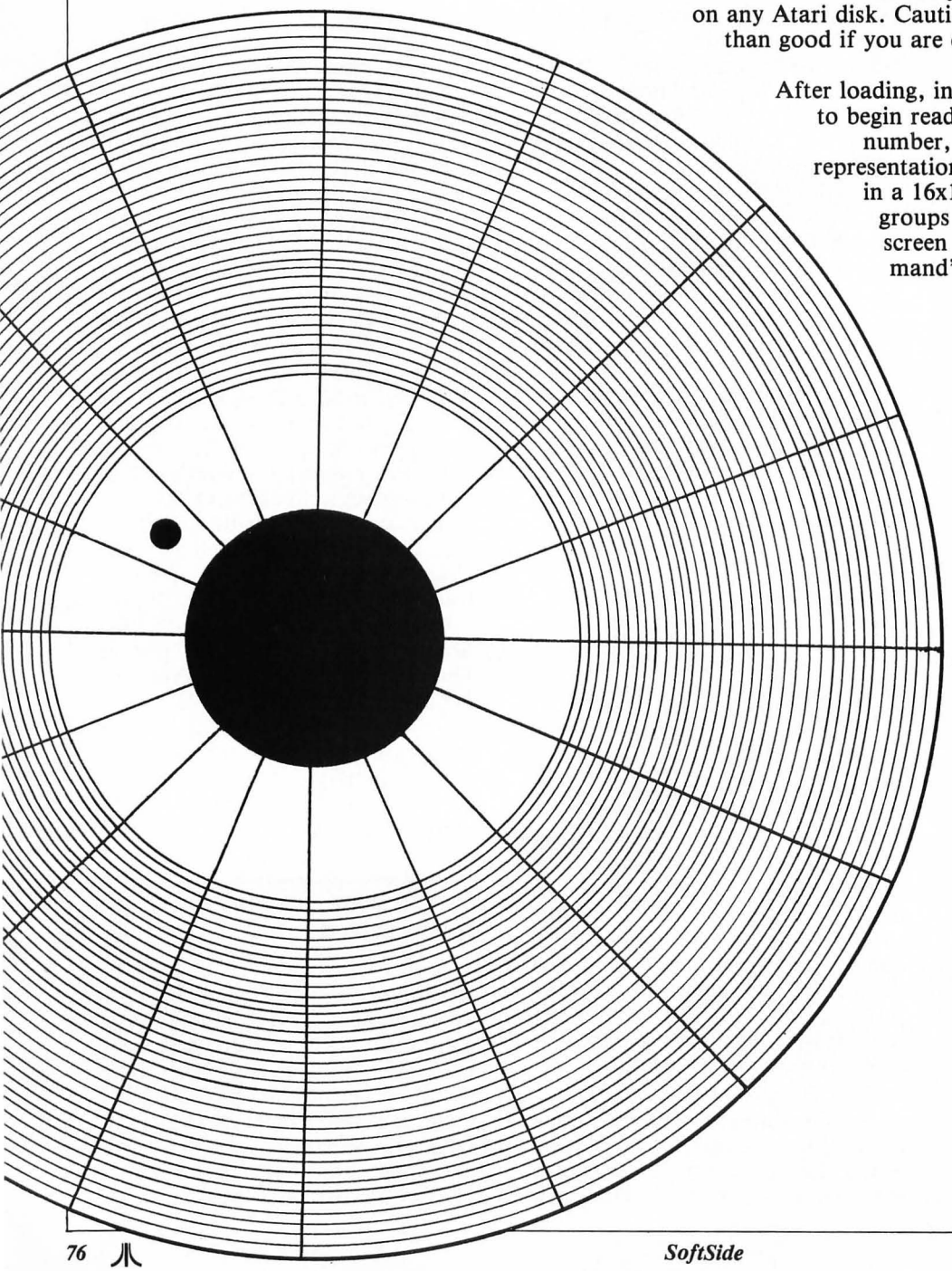
Scriptr and *Crayon* Summarized

Both of these programs are professionally constructed and well-documented. The documentation is convenient, easy to read, and well-written. I recommend both programs, especially *Scriptr*. Both of them are well-supported by Jerry Goodwin, who is willing to answer questions about either. *Scriptr* and *Crayon* are two great programs.

Disk Peeker-Poker

by Mike Westerfield. Atari® translation by Brad Sagarin

***Peeker-Poker* is a disk editing utility for an Atari with at least 16K. It is included as the bonus program on issue 37 DV. See the bind-in card elsewhere in the magazine to order this issue's disk.**



Disk Peeker-Poker allows you to examine and alter sector data on any Atari disk. Caution: this program can do more harm than good if you are careless. Altering certain sectors can destroy programs on the disk.

After loading, input the drive number and the sector to begin reading. The screen will show the sector number, the ATASCII, and the hexadecimal representation of the sector data. The hex will be in a 16x18 byte grid, with a space separating groups of four bytes. At the bottom of the screen is the command area. At the "command" prompt, you may enter one of the following commands:

Read

To read a sector, type CTRL-R. Don't hit RETURN. The computer will prompt for a sector number from 1 to 720. To exit this command without reading, type 0.

Write

Type CTRL-W (again, no RETURN) to write the data on the screen onto the disk. Type the sector number (1 to 720) onto which you wish to write. Typing 0 aborts this operation. You need not write data onto the same sector from which you read it.

Edit

To edit sector data, use the arrow keys to move the cursor to the nybble (half of a byte) you wish to change. Type any valid hexadecimal digit (0 to 9, A to F) to change that nybble. The ATASCII display changes automatically.

End

To end the program, type CTRL-Q and "Y" to confirm your action.

CAR RACE

by Jonathan D. Youngwood

***Car Race* is an arcade style game for an Atari® with a joystick and 32K RAM (24K disk).**

Instructions

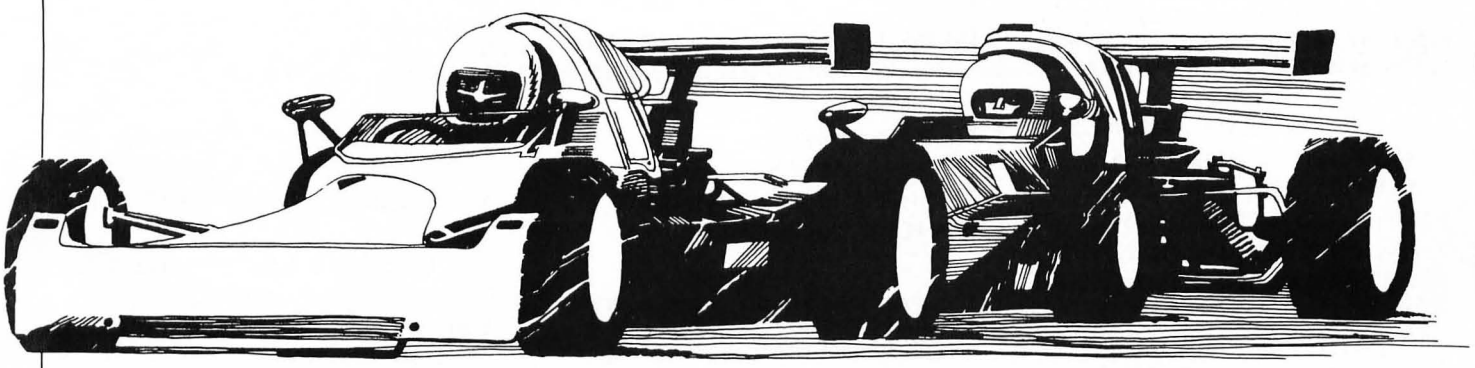
It's Saturday night, time to get out your old dragster and see what it can do. Just RUN the program, and you're all set. Of course, you have several options available, such as speed and level, which you can preset with the console keys while viewing the cover page:

OPTION — Level
SELECT — Speed
START — Begin
All three simultaneously — Preview the track

Because the track is different every time, we suggest you take a look at the track before playing. When you have all the options set, push START or your joystick button to begin. To change speed in the middle of the race, push forward or backward on the joystick; to pause in the middle of one of those grueling races, push the joystick button. The race continues until you have destroyed enough of your car that the tow truck has to come to the rescue and repair it. This occurs after you have crashed twice, although the computer does not register every collision as a crash. If you can manage to get past the first lap of your race, the second one starts, and your car repairs itself as you complete additional laps, giving you extra opportunities to crash. If you get off to a bad start in a game, push the START key and the cover graphic will be displayed.

Variables

A$: Screen currently drawing (D.V.) and/or data for line currently plotting.
A: Last screen drawn; also used to help free the required memory to store the track, to help update the clock, and to determine which of the special option keys has been pressed.
B,C: Used to help update the clock.
C$: Cursor control string.
D$: Cursor-down character.
CHS: Used to help update information on the screen (time, distance, etc.).
DL: Address of the display list.
DL4,DL5: Fourth and fifth numbers in the display list.
DIS: Holds distance covered.
HIT: Number of hits the car has taken.
I,J: Dummy variable used throughout the program.
LEVEL: Current level.
N,NN: Help set new colors for each level.
NOW: Holds the position of the top of the track.
NUMH: High byte of screen memory.
NUML: Low byte of screen memory.
PMBASE: Starting position of PM graphics.
Q0, Q1, Q2, etc.: Variables used to help conserve memory.
RT: Used to help draw track.
SP: Used in a FOR-NEXT loop to move the screen the proper amount of times.
SPEED: Current speed.
TIME: Current time.
U$: Cursor-up character.
X: Used several times in the program to represent the horizontal position of the players.
Y: Used to represent the vertical position of the players as they are being drawn.




```
SS SS SS SS SS SS SS SS SS SS SS
SS                           SS
SS        ATARI  BASIC       SS
SS         'CAR RACE'        SS
SS BY: JONATHAN D. YOUNGWOOD SS
SS     COPYRIGHT (c) 1982    SS
SS SOFTSIDE PUBLICATIONS, INC SS
SS                           SS
SS SS SS SS SS SS SS SS SS SS SS
```

If you don't wish to type this program, it is also included in this month's SoftSide CV and DV.

These lines define many variables which are used throughout the program to represent frequently used integers. This saves quite a bit of memory, for it requires much less memory to define a variable and then use it to represent an integer than to simply use the integer over and over. Also, they define cursor control strings.

```
2 DIM C$(9),U$(1),D$(1):U$=CHR$(28):D$
=CHR$(29):C$(1)=U$:C$(2)=U$:C$(3)=CHR$
(30):C$(9)=" ":C$(4)=C$(3)
3 Q0=0:Q1=1:Q2=2:Q4=3:Q5=4:Q6=5:Q7=280
00:Q8=14000:Q10=6:Q11=8:Q12=20000:Q13=
1000:Q14=9:Q15=10:Q16=11:Q17=12:Q18=13
:Q19=19
4 Q20=14:Q21=16:Q23=18:Q25=21:Q28=3000
:Q29=10000:Q31=1.5:Q34=20:Q35=7:Q36=30
30:Q37=26:Q39=39:Q40=40:Q41=4040:Q46=4
6:Q49=12000
5 Q50=50:Q60=60:Q64=64:Q69=3069:Q88=88
:Q89=89:Q106=106:Q128=128:Q176=176:Q18
8=188:Q200=200:Q232=232:Q248=248:Q249=
53248
6 Q250=53250:Q251=250:Q252=53252:Q253=
53249:Q255=255:Q256=256:Q260=53260:Q27
8=53278:Q279=53279:Q300=9300:Q400=9400
:Q512=512
7 Q536=65536:Q559=559:Q560=560:Q561=56
1:Q752=752
```

These lines simply put the initialization message on the screen and define a few more variables.

```
100 GRAPHICS Q0:SETCOLOR Q2,Q0,Q0:POKE
 Q752,Q1:? "INITIALIZATION...":PRINT "
SCREEN WILL BLANK FOR ABOUT 30 SECONDS
"
150 POSITION 15,9:PRINT "CAR RACE":? "
   Version 1.0(November 13, 1982)"
160 POSITION 3,11:? "(c) 1982 by Jonat
han D. Youngwood"
200 LEVEL=Q1:SPEED=Q1:GOSUB 7000:DIM A
$(Q40):POKE 82,Q0:TIME=Q1:NOW=Q0
1000 REM
```

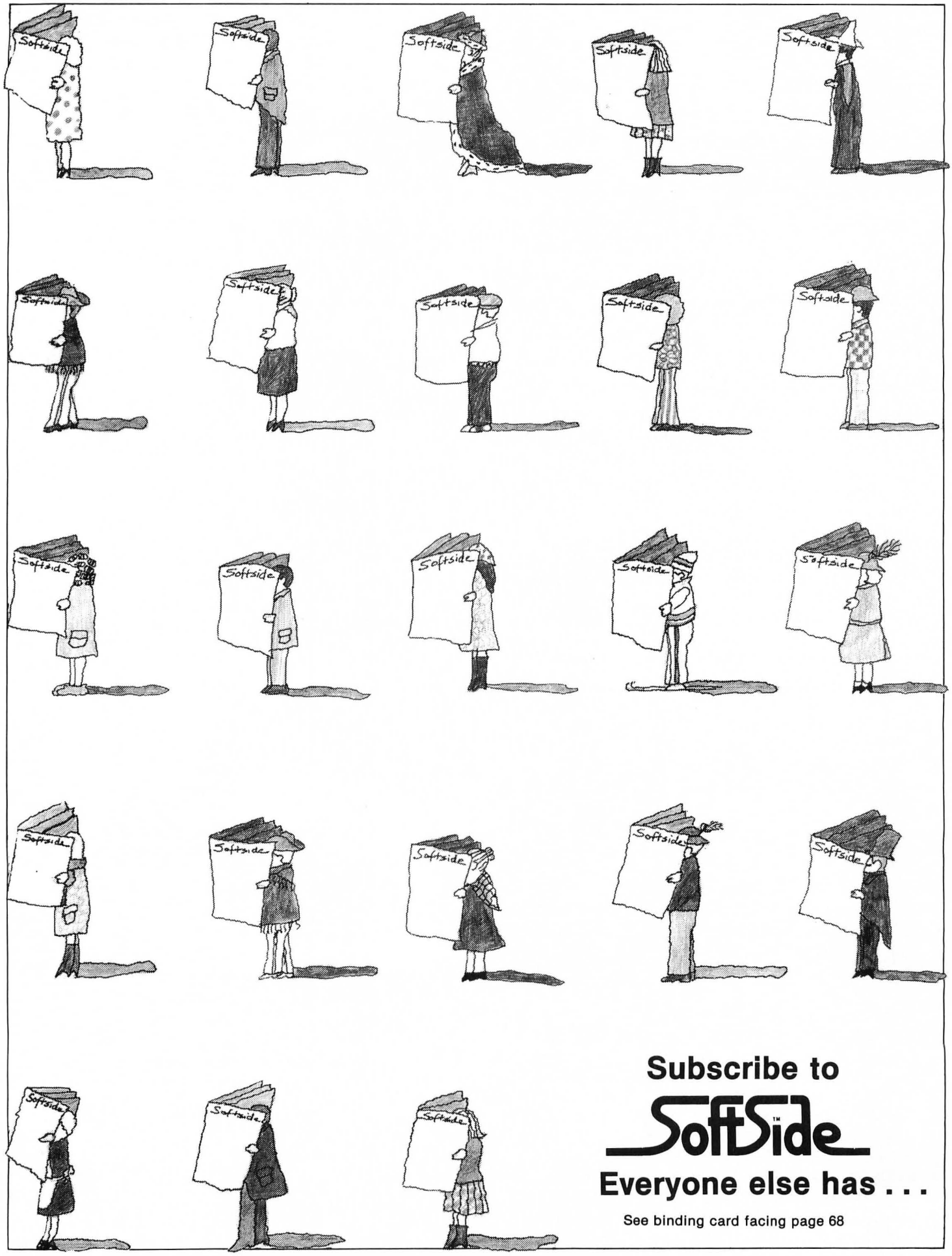
These are the lines that store the track in memory. Those who wish to read up on this subject and that of scrolling should take a look at the article by David Plotkin in Compute's Second Book of Atari.

```
1010 RT=PEEK(Q106):RT=RT-24:POKE Q106,
RT:GRAPHICS Q4:POKE Q752,Q1:POKE Q559,
Q0
1026 DL=PEEK(Q560)+Q256*PEEK(Q561):DL4
=DL+Q5:DL5=DL+Q6:POKE Q89,RT+Q2:POKE Q
88,Q0:RESTORE Q7:GOSUB Q8
1040 POKE Q89,RT+Q2:POKE Q88,Q200:REST
ORE 21000:GOSUB Q8:POKE Q89,RT+Q4:POKE
 Q88,144:GOSUB Q41:GOSUB Q8
1060 POKE Q89,RT+Q5:POKE Q88,Q88:GOSUB
 Q41:GOSUB Q8:POKE Q89,RT+Q6:POKE Q88,
32:GOSUB Q41:GOSUB Q8
1080 POKE Q89,RT+Q6:POKE Q88,Q232:GOSU
B Q41:GOSUB Q8:POKE Q89,RT+Q10:POKE Q8
8,Q176:GOSUB Q41:GOSUB Q8:PROB=A
1110 POKE Q89,RT+Q11:POKE Q88,Q64:REST
ORE Q12+(A*Q13):GOSUB Q8:POKE Q89,RT+Q
14:POKE Q88,Q11:GOSUB Q41:GOSUB Q8
1130 POKE Q89,RT+Q14:POKE Q88,208:GOSU
B Q41:GOSUB Q8:POKE Q89,RT+Q15:POKE Q8
8,152:GOSUB Q41:GOSUB Q8
1150 POKE Q89,RT+Q16:POKE Q88,96:GOSUB
 Q41:GOSUB Q8:POKE Q89,RT+Q17:POKE Q88
,Q40:GOSUB Q41:GOSUB Q8
1170 POKE Q89,RT+Q17:POKE Q88,240:GOSU
B Q41:GOSUB Q8:POKE Q89,RT+Q18:POKE Q8
8,184:GOSUB Q41:GOSUB Q8
1190 POKE Q89,RT+Q20:POKE Q88,Q128:GOS
UB Q41:GOSUB Q8:POKE Q89,RT+Q21:POKE Q
88,Q21:RESTORE Q12+(A*Q13):GOSUB Q8
1220 POKE Q89,RT+Q21:POKE Q88,216:GOSU
B Q41:GOSUB Q8:POKE Q89,RT+17:POKE Q88
,160:GOSUB Q41:GOSUB Q8
1240 POKE Q89,RT+Q23:POKE Q88,104:GOSU
B Q41:GOSUB Q8:POKE Q89,RT+Q19:POKE Q8
8,48:GOSUB Q41:GOSUB Q8
1260 POKE Q89,RT+Q19:POKE Q88,Q248:RES
TORE Q7:GOSUB Q8:POKE Q89,RT+Q25:POKE
Q88,Q188:RESTORE 29000:GOSUB Q8
1999 GOSUB 9000:POKE Q559,Q46:GOSUB Q4
00:GOTO Q28
```

This is the main loop of the program. After initialization has taken place, most of the time is spent here.

```
3000 GOSUB Q29:HIT=Q0:X=123:DL=PEEK(Q5
60)+Q256*PEEK(Q561):DL4=DL+Q5:DL5=DL+Q
6:POKE Q250,Q0:GOSUB Q300:GOSUB 6000:V
=Q2
3001 DIS=Q0:POKE DL5,RT+Q19:POKE DL4,Q
248:NUML=PEEK(DL4):NUMH=PEEK(DL5):GOSU
B Q400
```

continued on page 80



Subscribe to

SoftSide™

Everyone else has . . .

See binding card facing page 68

Car Race *continued*

```
3025 ? U$;U$;U$;:? "SPEED:";SPEED;D$;D
$:SOUND Q0,38,Q0,(SPEED*Q31)-Q1:SOUND
Q1,67,Q10,SPEED*Q31:POKE 77,Q0
3030 ST=STICK(Q0):IF ST<>Q18 AND ST<>Q
20 THEN CHS=Q0
3032 IF ST=Q18 AND CHS=Q0 THEN SPEED=S
PEED-Q1:CHS=Q1:IF SPEED=Q0 THEN SPEED=
Q1
3033 IF ST=Q20 AND CHS=Q0 THEN SPEED=S
PEED+Q1:CHS=Q1:IF SPEED=Q6 THEN SPEED=
Q5
3034 IF CHS=Q1 THEN ? "}";:GOTO 3025
3035 IF STRIG(Q0)=Q0 THEN GOTO 9200
3036 C=PEEK(Q23):B=PEEK(Q19):A=PEEK(Q3
4):? U$;U$;"TIME:";INT((A+B*Q256+C*Q53
6)/Q60):? "LEVEL:";LEVEL,
3037 PRINT "DISTANCE:";:DIS=DIS+SPEED/
Q15:PRINT DIS
3038 X=X+(Q6-LEVEL)*(ST=Q35)-(Q6-LEVEL
)*(ST=Q16):FOR SP=Q1 TO SPEED:NUML=NUM
L-Q15:NUMH=NUMH+(NUML>Q255)-(NUML<Q0)
3050 NUML=NUML+Q256*(NUML<Q0)-Q256*(NU
ML>Q255)
3051 IF NUML<>NOW OR NUMH<>RT+Q2 THEN
POKE 77,0:GOTO 3055
3052 NUML=Q248:NUMH=RT+Q19:LEVEL=LEVEL
+Q1:GOSUB Q400:IF LEVEL=Q6 THEN LEVEL=
Q5
3053 HIT=HIT-INT(RND(Q0)*Q2)
3055 IF NUML=Q64 AND NUMH=RT+Q11 THEN
NUML=Q176:NUMH=RT+Q10
3056 IF NUML=Q21 AND NUMH=RT+Q21 THEN
NUML=Q128:NUMH=RT+Q20
3057 IF ST=Q35 OR ST=Q16 THEN SOUND Q4
,Q251,Q11,Q20
3058 IF PEEK(Q279)=Q10 THEN GOSUB Q300
:GOTO Q28
3059 POKE DL4,NUML:POKE DL5,NUMH:POKE
Q249,X:SOUND Q4,Q0,Q0,Q0:POKE Q278,Q1
3061 IF PEEK(Q260)<>Q0 OR PEEK(Q252)<>
Q0 THEN GOTO 3065
3062 NEXT SP:GOTO Q36
3065 POKE Q278,Q1:IF PEEK(Q260)<>Q0 OR
 PEEK(Q252)<>Q0 THEN GOTO Q69
3066 NEXT SP:GOTO Q36
3067 POKE Q278,Q1:IF PEEK(Q260)<>Q0 OR
 PEEK(Q252)<>Q0 THEN GOTO Q69
3068 NEXT SP:GOTO Q36
3069 HIT=HIT+Q1:SOUND Q2,Q50,Q15,Q20:F
OR I=Q1 TO Q50:NEXT I:SOUND Q2,Q0,Q0,Q
0:SOUND Q2,Q50,Q15,Q20:FOR I=Q1 TO Q50
:NEXT I:SOUND Q2,Q0,Q0,Q0
3070 IF HIT=Q2 THEN GOTO 9100
3080 GOTO Q36
4040 A=INT(RND(Q0)*Q11):RESTORE Q12+(A
*Q13):IF A=Z AND Z=Q1 THEN GOTO Q41
4050 Z=A:RETURN
```

Move the car up to the starting line.

```
6000 POKE Q249,Q0:POKE Q253,Q0:POKE DL
5,RT+Q34:POKE DL4,Q232:FOR I=Q0 TO X:P
OKE Q253,I:NEXT I:POKE Q249,X:POKE Q25
3,Q0
6020 FOR I=Q232 TO Q2 STEP -Q15:POKE D
L4,I:FOR J=Q1 TO Q6:NEXT J:NEXT I:POKE
 Q34,Q0:POKE Q19,Q0:POKE Q23,Q0:RETURN
```

Load the cars into PM graphics.

```
7000 X=Q0:Y=87:A=PEEK(Q106)-Q11:POKE 5
4279,A:PMBASE=Q256*A:POKE Q106,A:POKE
Q559,Q46:POKE 53277,Q4
7030 POKE Q249,X:POKE Q253,X:POKE Q250
,X:FOR I=PMBASE+Q512 TO PMBASE+896:POK
E I,Q0:NEXT I
7050 POKE 704,Q37:POKE 705,Q37:POKE 70
6,98:POKE 53256,Q0:POKE 53257,Q0:POKE
53258,Q1:POKE 623,Q1
7060 RESTORE 8000:FOR I=PMBASE+Q512+Y
TO PMBASE+519+Y:READ A:POKE I,A:NEXT I
7070 Y=Q89:RESTORE 8010:FOR I=PMBASE+6
40+Y TO PMBASE+646+Y:READ A:POKE I,A:N
EXT I
7080 Y=90:RESTORE 8020:FOR I=PMBASE+76
8+Y TO PMBASE+772+Y:READ A:POKE I,A:NE
XT I
7090 RETURN
8000 DATA 16,186,254,170,40,170,254,18
6
8010 DATA 238,68,254,199,254,68,238
8020 DATA 18,31,255,31,18
```

Finish drawing the track.

```
9000 RESTORE Q12:POKE Q89,RT+Q34:POKE
Q88,192:GOSUB Q8
9010 POKE Q88,Q232:COLOR Q0:FOR I=Q21
TO Q19:PLOT Q0,I:DRAWTO Q39,I:NEXT I:R
ETURN
```

The truck routine.

```
9100 SOUND Q1,Q0,Q0,Q0:SOUND Q0,Q0,Q0,
Q0:FOR I=Q0 TO X+15 STEP Q2:POKE Q250,
I:IF I/Q5=INT(I/Q5) THEN SOUND Q2,Q50,
Q5,Q11
9112 IF I/Q5<>INT(I/Q5) THEN SOUND Q2,
Q0,Q0,Q0
9113 NEXT I:POKE Q249,Q0:POKE Q253,X:F
OR I=X TO 220 STEP Q2:POKE Q253,I:POKE
 Q250,I+Q35
9130 IF (I-X)/Q5=INT((I-X)/Q5) THEN SO
UND Q2,Q50,Q5,Q11
9140 IF (I-X)/Q5<>INT((I-X)/Q5) THEN S
OUND Q2,Q0,Q0,Q0
9150 NEXT I:SOUND Q2,Q0,Q0,Q0:GOTO Q28
```

The pause routine.

```
9200 A=PEEK(Q34):B=PEEK(Q19):C=PEEK(Q2
3):FOR I=Q1 TO Q200:NEXT I
9210 IF STRIG(Q0)=Q1 THEN GOTO 9210
9220 IF STRIG(Q0)=Q1 THEN POKE Q34,A:P
OKE Q19,B:POKE Q23,C:GOTO 3038
9230 GOTO 9220
```

Set the clock to zero.

```
9300 POKE Q23,Q0:POKE Q19,Q0:POKE Q34,
Q0:? :? :? :RETURN
9400 N=Q2*(LEVEL-Q1):NN=LEVEL-Q1:SETCO
LOR Q5,Q0+NN,Q0+N:SETCOLOR Q0,Q4+NN,Q5
+N:SETCOLOR Q1,Q11+NN,Q10*N:SETCOLOR Q
2,Q17+NN,N+Q11:RETURN
```

The cover subroutine.

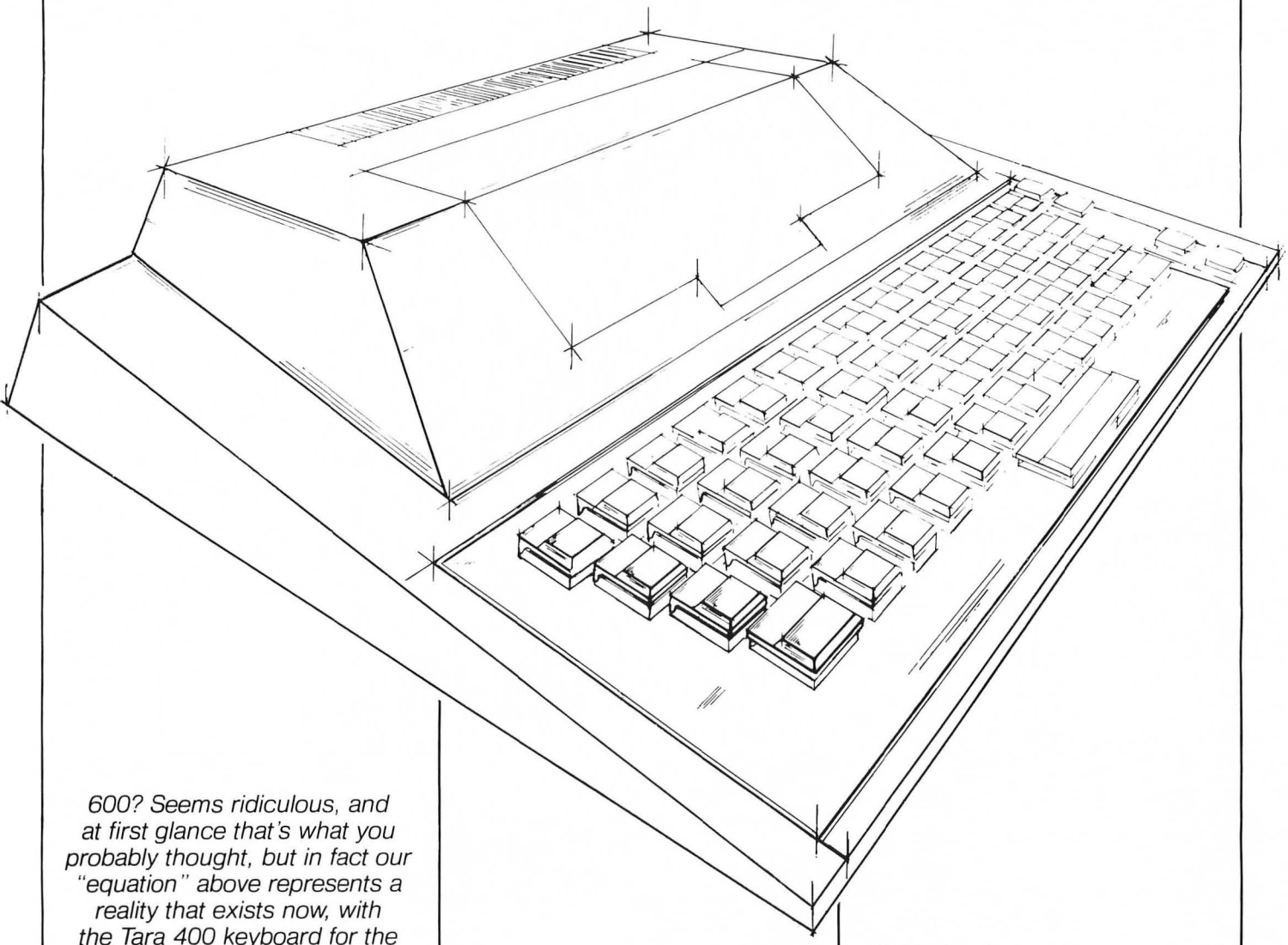
```
10000 SOUND Q0,Q0,Q0,Q0:SOUND Q1,Q0,Q0
,Q0
10010 ? "}":POKE Q249,Q0:POKE Q253,Q0:
POKE Q250,Q0:POKE DL5,RT+Q25:POKE DL4,
Q188:FOR I=Q1 TO 100:NEXT I
10011 TIME=INT((A+B*Q256+C*Q536)/Q60):
IF TIME=Q0 THEN TIME=Q1
10015 PRINT "PRESS SELECT FOR SPEED, O
PTION FOR LEVEL";
10020 PRINT "SPEED:";SPEED,"DISTANCE:"
;DIS:SOUND Q0,RND(Q0)*Q255,Q15,Q2
10030 PRINT "TIME:";TIME,:PRINT "SCORE
:";INT(Q40*DIS/TIME)+INT(TIME*DIS):PRI
NT "LEVEL:";LEVEL;:PRINT C$;
10060 A=PEEK(Q279)
10070 IF A=Q10 OR STRIG(Q0)=Q0 THEN ?
"}";:GOSUB Q49:SOUND Q0,Q0,Q0,Q0:RETUR
N
10080 IF A=Q6 THEN SPEED=SPEED+Q1:IF S
PEED=Q6 THEN SPEED=Q1
10090 IF A=Q4 THEN LEVEL=LEVEL+Q1:IF L
EVEL=Q6 THEN LEVEL=Q1
10095 IF A=Q0 THEN GOTO 10200
10100 GOSUB Q400:GOTO 10020
```

This subroutine shows the user the entire track.

```
10200 SOUND Q0,Q0,Q0,Q0:DL=PEEK(Q560)+
Q256*PEEK(Q561):DL4=DL+Q5:DL5=DL+Q6:GO
SUB Q49:POKE DL5,RT+Q19:POKE DL4,Q248
```

continued on page 82

400 + 400 = 600?



600? Seems ridiculous, and at first glance that's what you probably thought, but in fact our "equation" above represents a reality that exists now, with the Tara 400 keyboard for the Atari 400. Designed to provide the Atari 400 user with the hardware of tomorrow, today. Designed with an understanding of the essential superiority of a keyboard as a man–machine interface. Designed with the user in mind. For example, our keyboard does not attach to the 400 with a ribbon cable, but fits neatly into the original housing in 5 minutes, directly replacing the old membrane panel, and is styled to complement the lines of the computer itself. Sure, other keyboards have been sold, but who wants one that hangs off the computer, or whose keys fall off when you type on it? Our keys are actually gold-contact switches, offering increased reliability and performance, second to none. Coupled with the Tara 48K RAM expansion board, you can easily see how 400 + 400 = 600, providing the user today with the hardware of tomorrow.*

Why wait? This and many of the quality Tara products are waiting for you at your favorite dealer. Or call us for the Tara dealer nearest you. He'll be happy to show you how rudimentary it can all be with Tara.

Tara
Computer Products Inc.

Statler Building, 107 Delaware Ave.,
Suite 1410, Buffalo, N.Y. 14202 (716) 855-0133

2 Robert Speck Parkway, Suite 1540,
Mississauga, Ontario L42 1H8
(416) 273-6820

TARA PRODUCT LINE:
- Atari 400 Keyboard
- Atari 48K RAM
- Atari 32K RAM
- Apple 16K RAM

COMING SOON FROM TARA
- Apple 64K/128K RAM
- IBM 256K RAM
- Atari 64K RAM

**600 – The Atari redesigned full keyboard version of the Atari 400.*

Atari and Atari 400 are registered trademarks of Warner Commmunications.

Car Race *continued*

```
10202 NUML=PEEK(DL4):NUMH=PEEK(DL5):PR
INT "}":PRINT "The track..."
10209 NUML=NUML-Q15:NUMH=NUMH+(NUML=Q2
55)-(NUML<Q0):NUML=NUML+Q256*(NUML<Q0)
-Q256*(NUML>Q255)
10220 IF NUML=NOW AND NUMH=RT+Q2 THEN
GOTO Q29
10230 IF NUML=Q64 AND NUMH=RT+Q11 THEN
 NUML=Q176:NUMH=RT+Q10
10240 IF NUML=Q21 AND NUMH=RT+Q21 THEN
 NUML=Q128:NUMH=RT+Q20
10250 POKE DL4,NUML:POKE DL5,NUMH:GOTO
 10209
```

Find the beginning of the subroutine.

```
12000 POKE Q89,RT+Q2
12010 POKE Q88,NOW
12015 LOCATE Q0,Q0,Z:IF Z<>Q0 THEN RET
URN
12020 IF NOW=Q251 THEN RUN
12030 NOW=NOW+Q15:GOTO 12010
```

This line draws each screen.

```
14000 POSITION Q0,Q0:FOR I=Q1 TO Q34:R
EAD A$:PRINT #Q10;A$;:NEXT I:RETURN
```

The data for the screens.

```
20000 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
20010 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
20020 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
20030 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
20040 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
20050 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
20060 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
20070 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
20080 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
20090 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
20100 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
20110 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
20120 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
20130 DATA ABBBBBBBBBBBBBCCDDDDDDDDDDDC
BBBBBBBBBBBBA
20140 DATA ABBBBBBBBBBBBCCDDDDDDDDDDDDC
CBBBBBBBBBBBA
20150 DATA ABBBBBBBBBBBCCDDDDDDDDDDDDDD
CCBBBBBBBBBBA
20160 DATA ABBBBBBBBBCCCDDDDDDDDDDDDDDD
DCCCBBBBBBBBA
20170 DATA ABBBBBBBBBCDDDDDDDDDDDDDDDDD
DDDCBBBBBBBBA
20180 DATA ABBBBBBBBBCDDDDDDDDDDDDDDDDD
DDDCBBBBBBBBA
20190 DATA ABBBBBBBBBCDDDDDDDDDDDDDDDDD
DDDCBBBBBBBBA
21000 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21010 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21020 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21030 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21040 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21050 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21060 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21070 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21080 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21090 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21100 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21110 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21120 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21130 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21140 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21150 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21160 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21170 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21180 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
21190 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDCB
BBBBBBBBBBBBA
22000 DATA ABBBBBBBBBBBCDDDDDDDDDDDDDDD
DCBBBBBBBBBBA
22010 DATA ABBBBBBBBBBBCDDDDDDDDDDDDDDD
DCBBBBBBBBBBA
22020 DATA ABBBBBBBBBBBCDDDDDDDDDDDDDDD
DCBBBBBBBBBBA
22030 DATA ABBBBBBBBBBBCDDDDDDDDDDDDDDD
DCBBBBBBBBBBA
22040 DATA ABBBBBBBBBBBCDDDDDDCCCCDDDDD
DCBBBBBBBBBBA
22050 DATA ABBBBBBBBBBBCDDDDDCCCCCCDDDD
DCBBBBBBBBBBA
22060 DATA ABBBBBBBBBBBCDDDDDCAAAACDDDD
DCBBBBBBBBBBA
22070 DATA ABBBBBBBBBBBCDDDDDCCCCCCDDDD
DCBBBBBBBBBBA
22080 DATA ABBBBBBBBBBBCDDDDDCAAAACDDDD
DCBBBBBBBBBBA
22090 DATA ABBBBBBBBBBBCDDDDDCCCCCCDDDD
DCBBBBBBBBBBA
22100 DATA ABBBBBBBBBBBCDDDDDCAAAACDDDD
DCBBBBBBBBBBA
22110 DATA ABBBBBBBBBBBCDDDDDCCCCCCDDDD
DCBBBBBBBBBBA
22120 DATA ABBBBBBBBBBBCDDDDDCAAAACDDDD
DCBBBBBBBBBBA
22130 DATA ABBBBBBBBBBBCDDDDDCCCCCCDDDD
DCBBBBBBBBBBA
22140 DATA ABBBBBBBBBBBCDDDDDCAAAACDDDD
DCBBBBBBBBBBA
22150 DATA ABBBBBBBBBBBCDDDDDCCCCCCDDDD
DCBBBBBBBBBBA
22160 DATA ABBBBBBBBBBBCDDDDDDCCCCDDDDD
DCBBBBBBBBBBA
22170 DATA ABBBBBBBBBBBCDDDDDDDDDDDDDDD
DCBBBBBBBBBBA
22180 DATA ABBBBBBBBBBBCDDDDDDDDDDDDDDD
DCBBBBBBBBBBA
22190 DATA ABBBBBBBBBBBCDDDDDDDDDDDDDDD
DCBBBBBBBBBBA
23000 DATA ABBBBBBBBBBBBBBBBCDDDDDDDDD
CBBBBBBBBBBBA
23010 DATA ABBBBBBBBBBBBBBBBBCDDDDDDDDC
CBBBBBBBBBBBA
23020 DATA ABBBBBBBBBBBBBBBBBCDDDDDDDDC
BBBBBBBBBBBBA
23030 DATA ABBBBBBBBBBBBBBBBCCCDDDDDDDCC
BBBBBBBBBBBBA
23040 DATA ABBBBBBBBBBBBBBBBCDDDDDDDDCCB
BBBBBBBBBBBBA
23050 DATA ABBBBBBBBBBBBBBCCCDDDDDDDDCBB
BBBBBBBBBBBBA
23060 DATA ABBBBBBBBBBBBBBCDDDDDDDDDDCBB
BBBBBBBBBBBBA
23070 DATA ABBBBBBBBBBBBCCCDDDDDDDDDCCBB
BBBBBBBBBBBBA
23080 DATA ABBBBBBBBBBBBCDDDDDDDDDDCCBBB
BBBBBBBBBBBBA
23090 DATA ABBBBBBBBBBBBCDDDDDDDDDCCBBBB
BBBBBBBBBBBBA
23100 DATA ABBBBBBBBBBBBCDDDDDDDDDCCBBBB
BBBBBBBBBBBBA
```

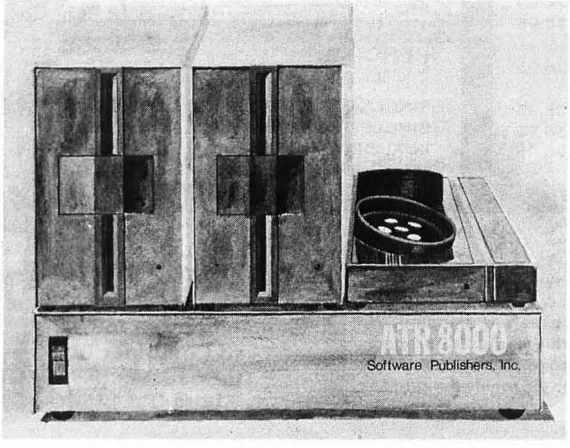
```
23110 DATA ABBBBBBBBBBCDDDDDDDDDDCBBBB
BBBBBBBBBBBA
23120 DATA ABBBBBBBBBBCDDDDDDDDDDCCBBB
BBBBBBBBBBBA
23130 DATA ABBBBBBBBBBCCCDDDDDDDDDCBBB
BBBBBBBBBBBA
23140 DATA ABBBBBBBBBBBCDDDDDDDDDCCBB
BBBBBBBBBBBA
23150 DATA ABBBBBBBBBBBCCDDDDDDDDDCBB
BBBBBBBBBBBA
23160 DATA ABBBBBBBBBBBBCCDDDDDDDDCCB
BBBBBBBBBBBA
23170 DATA ABBBBBBBBBBBBBCCCDDDDDDDCC
BBBBBBBBBBBA
23180 DATA ABBBBBBBBBBBBBBBCDDDDDDDDC
CBBBBBBBBBBA
23190 DATA ABBBBBBBBBBBBBBBCCDDDDDDDD
CBBBBBBBBBBA
24000 DATA ABBBBBBBBBBBBBBBCDDDDDDDDD
CBBBBBBBBBBA
24010 DATA ABBBBBBBBBBBBBBBCDDDDDDDDD
CBBBBBBBBBBA
24020 DATA ABBBBBBBBBBBBBBBCDDDDDDDDD
CBBBBBBBBBBA
24030 DATA ABBBBBBBBBBBBBBBCCCDDDDDDDD
CCCBBBBBBBBBA
24040 DATA ABBBBBBBBBBBBBBBCDDDDDDDDDD
DDCBBBBBBBBBA
24050 DATA ABBBBBBBBBBBBBCCCDDDDDDDDCDD
DDCCCBBBBBBBA
24060 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDDD
DDDDCBBBBBBBA
24070 DATA ABBBBBBBBBBBCCCDDDDDDDDDDDDC
DDDDCCCBBBBBA
24080 DATA ABBBBBBBBBBBCDDDDDDDCDDDDDDD
DDDDDDCBBBBBA
24090 DATA ABBBBBBBBBBBCDDDDDDDDDDDDDDD
DDCDDDCBBBBBA
24100 DATA ABBBBBBBBBBBCDDDDDDDDDDCDDDD
DDDDDDCBBBBBA
24110 DATA ABBBBBBBBBBBCDDDDDDDDDDDDDDD
DDDDDDCBBBBBA
24120 DATA ABBBBBBBBBBBCDDDDDDDDDDDDDDD
DDDDDDCBBBBBA
24130 DATA ABBBBBBBBBBBCCCDDDCDDDDDDDDD
DDDDCCCBBBBBA
24140 DATA ABBBBBBBBBBBBBCDDDDDDDDDDDDC
DDDDCBBBBBBBA
24150 DATA ABBBBBBBBBBBBBCCCDDDDDDDDDDD
DDCCCBBBBBBBA
24160 DATA ABBBBBBBBBBBBBBCDDDDDDDDDDD
DDCBBBBBBBBBA
24170 DATA ABBBBBBBBBBBBBBCCCDDDDDDDDD
DCCBBBBBBBBBA
24180 DATA ABBBBBBBBBBBBBBBBCDDDDDDDDD
DCBBBBBBBBBBA
24190 DATA ABBBBBBBBBBBBBBBBCCDDDDDDDD
CCBBBBBBBBBBA
25000 DATA ABBBBBBBBCCCCCDDDDDDDDDCCC
CCCCCBBBBBBBA
25010 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDDCBBBBBBBA
25020 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDDCBBBBBBBA
25030 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDDCBBBBBBBA
25040 DATA ABBBBBBBBCDDDDDDDDCDDDDDDD
DDDDCBBBBBBBA
25050 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDDCBBBBBBBA
25060 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDDCBBBBBBBA
25070 DATA ABBBBBBBBCDDDDDAAADDDDAAADD
DDDDCBBBBBBBA
25080 DATA ABBBBBBBBCDDDDDAAADDDDAAADD
DDDDCBBBBBBBA
25090 DATA ABBBBBBBBCDDDDDAAAADDDAAADD
DDDDCBBBBBBBA
25100 DATA ABBBBBBBBCDDDDDAAADDDDAAADD
DDDDCBBBBBBBA
```

continued on page 85

NOW YOU *CAN* MIX BUSINESS AND PLEASURE . . . THE ATR8000 AND THE ATARI® 800/400 GIVE YOU GREAT GAMES *AND* A VIABLE DOUBLE DENSITY BUSINESS COMPUTER.

The ATR8000 converts the ATARI 800/400 into an expandable 4MHz Z80 double density business machine. Add the 64K upgrade (includes CP/M) and step into a further dimension as you enter the vast CP/M market place to select programs tailored to your specific needs.

The ATR8000 is versatile, so it can grow as you need it to. Its handling of disk drives is revolutionary . . . it runs four 5¼" or 8" drives, single or double or quad density, single or double-sided *AND* allows you to mix them!



SPECS: 4MHz Z80 processor • 16k RAM standard • Connects to expansion port of the ATARI 800/400 • 12½" x 11½" x 2½" gray and beige enclosure • Runs four drives of mixed definition • Centronics parallel and RS-232 serial port (and printer drivers) • Runs single density ATARI DOS and existing ATARI software • With OSA+, Ver. 4, is double density • 64K upgrade includes CP/M.

PRICING:	ATR8000	$499.95	OSA+, Ver. 4	$49.95
	5¼" Drive	$399.95	Printer Cable	$29.00
	64K Upgrade (with CP/M)	$250.00	5¼" Drive Cable	$35.00
			8" Drive Adapter	—CALL—

CONTACT: SOFTWARE PUBLISHERS, INC.
2500 E. Randol Mill Rd., Suite 125
Arlington, TX 76011 817-469-1181

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

Lyco Computer Marketing & Consultants

TO ORDER CALL US

TOLL FREE 800-233-8760

In PA 1-717-398-4079

FEBRUARY ATARI SPECIALS

810 Disk Drive ... $429.00

400 32K RAM ... $CALL$

NEW ATARI COMPUTER ... $ CALL $

800 48K... $499.00

ATARI®

A Warner Communications Company

ATARI HARDWARE

410 CASSETTE RECORDER	$ 75.00
825 PRINTER	$585.00
830 PHONE MODEM	$149.00
850 INTERFACE	$164.00

PACKAGES

CX481 ENTERTAINER	$ 69.00
CX482 EDUCATOR	$125.00
CX483 PROGRAMMER	$ 49.00
CX494 COMMUNICATOR	$325.00

SOFTWARE

CXL4012 MISSILE COMMAND	$28.75
CXL4013 ASTEROID	$28.75
CXL4020 CENTIPEDE	$32.75
CXL4022 PACMAN	$32.75
CXL4011 STAR RAIDER	$34.75
CXL4004 BASKETBALL	$26.75
CXL4006 SUPER BREAKOUT	$28.75
CXL4008 SPACE INVADER	$28.75
CX8130 CAVERNS OF MARS	$31.75
CX4108 HANGMAN	$12.75
CX4102 KINGDOM	$12.75
CX4112 STATES & CAPITALS	$12.75
CX4114 EUROPEAN COUNTRIES	$12.75
CX4109 GRAPHIT	$16.75
CX4121 ENERGY CZAR	$12.75
CX4123 SCRAM	$19.75
CX4101 PROGRAMMING I	$19.75
CX4106 PROGRAMMING II	$22.75
CX4117 PROGRAMMING III	$22.75
CXL4015 TELELINK	$21.75
CX4119 FRENCH	$39.75
CX4118 GERMAN	$39.75
CX4120 SPANISH	$39.75
CXL4007 MUSIC COMPOSER	$33.75
CXL4002 ATARI BASIC	$45.75
CX8126 MICROSOFT BASIC	$65.75
CXL4003 ASSEMBLER EDITOR	$45.75
CX8126 MACRO ASSEMBLER	$69.75
CXL4018 PILOT HOME	$65.75
CX405 PILOT EDUCATOR	$99.75
CX415 HOME FILING MANAGER	$41.75
CX414 BOOKKEEPER	$119.75

NEW RELEASES

CHOP LIFTER	$27.75
APPLE PANIC	$23.75
PREPPIE	$19.95

THIRD PARTY SOFTWARE

EASTERN FRONT 1941	$25.50
OUTLAW/HOWITZER	$15.50
WIZARD of WAR	$31.00
GORF	$31.00
FROGGER	$26.00
CHOP LIFTER	$27.75
APPLE PANIC	$23.75
PREPPIE	$19.95
STAR WARRIOR	$28.00
CRUSH,CRUMBLE,&CHOMP	$23.00
SHOOTING GALLERY	$19.95
VIDEO MATH FLASH	$12.00
MY FIRST ALPHABET	$25.50
BAHA BUGGIES	$24.95
TEMPLE of ASPHAI	$27.95
UPPER REACHES of ASPHAI	$15.00
TRACK ATTACK	$23.00
STAR BLAZER	$25.00
LABYRINTH	$23.00
SEA FOX	$23.00
POOL 1.5	$26.95
SPEEDWAY BLAST (ROM)	$29.95
JAWBREAKER	$22.95
THRESHOLD	$29.95
MOONBASE IO	$23.95
PROTECTOR	$24.95
NAUTILUS	$24.95
SLIME	$24.95
SUBMARINE COMMANDER (ROM)	$36.95
JUMBO JET PILOT (ROM)	$36.95
SOCCER (ROM)	$36.95
KICKBACK (football ROM)	$36.95

PRINTERS

Okidata 82A	$479.00
Okidata 83A	$719.00
Okidata 84	$1089.00
Citoh	CALL
Prowriter I	$499.00
Prowriter II	CALL
SMITH CORONA TP-1	$625.00
NEC	CALL

(Interfacing Available)

BUSINESS SOFTWARE

ATARI WORD PROCESSING	$109.00
LETTER PERFECT (ROM)	$149.00
LETTER PERFECT (disc)	$129.00
TEXT WIZZARD	$ 89.00
DATA PERFECT	$ 75.00
VISICALC	$169.00
DATASAM/65	$125.00

JOYSTICKS

ATARI CX-40	$18.00
LESTICK	$34.00
WICO COMMAND CONTROL	$23.75
WICO RED BALL	$26.75
WICO TRACK BALL	$54.75
STICK STAND	$ 6.75

COMPUTER COVERS

800	$6.99
400	$6.99
410	$6.99
810	$6.99

PERCOM

SINGLE DRIVE (SD)	$399.00
SINGLE DRIVE (DD)	$549.00
DUAL DRIVE (DD)	$869.00
DUAL HEAD (DD)	$669.00

POLICY

In-Stock items shipped within 24 hours of order. Personal checks require four weeks clearance before shipping. No deposit for COD orders. PA residents add sales tax. All products subject to availability and price change. Advertised prices show 4% discount offered for cash. Add 4% for Mastercard and Visa.

TO ORDER CALL TOLL FREE

800-233-8760

In PA 1-717-398-4079 or send order to

Lyco Computer
P.O. Box 5088
Jersey Shore, PA 17740

Car Race *continued*

```
25110 DATA ABBBBBBBBCDDDDDAADDDDDAAADD
DDDDCBBBBBBBA
25120 DATA ABBBBBBBBCDDDDDADDDDDAAAADD
DDDDCBBBBBBBA
25130 DATA ABBBBBBBBCDDDDDADDDDDAADDD
DDDDCBBBBBBBA
25140 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDDCBBBBBBBA
25150 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDDCBBBBBBBA
25160 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDDCBBBBBBBA
25170 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDDCBBBBBBBA
25180 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDDCBBBBBBBA
25190 DATA ABBBBBBBBCCCCCCDDDDDDDDDCCC
CCCCCBBBBBBBA
26000 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDCBBBBBBBBA
26010 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDCBBBBBBBBA
26020 DATA ABBBBBBBBCCDDDDDDDDDDDDDDD
DDCCBBBBBBBBA
26030 DATA ABBBBBBBBBCDDDDDDDDDDDDDDD
DDCBBBBBBBBBA
DDCBBBBBBBBBA
26040 DATA ABBBBBBBBBCCDDDDDDDDDDDDDD
DCCBBBBBBBBBA
26050 DATA ABBBBBBBBBBCCCCAAAAADDDDDCC
CBBBBBBBBBBA
26060 DATA ABBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBA
26070 DATA ABBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBA
26080 DATA ABBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBA
26090 DATA ABBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBA
26100 DATA ABBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBA
26110 DATA ABBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBA
26120 DATA ABBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBA
26130 DATA ABBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBA
26140 DATA ABBBBBBBBBBBBCCDDDDDDDDDDCB
BBBBBBBBBBBA
26150 DATA ABBBBBBBBBBBCCDDDDDDAAAAACC
CBBBBBBBBBBBA
26160 DATA ABBBBBBBBBBCCDDDDDDDDDDDDDD
CCBBBBBBBBBBA
26170 DATA ABBBBBBBBCCCDDDDDDDDDDDDDDD
DCCCBBBBBBBBA
26180 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDCBBBBBBBBA
26190 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDCBBBBBBBBA
27000 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDCBBBBBBBBA
27010 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDCBBBBBBBBA
27020 DATA ABBBBBBBBCCDDDDDDDDDDDDDDD
DDCCBBBBBBBBA
27030 DATA ABBBBBBBBBCDDDDDDDDDDDDDDD
DDCBBBBBBBBBA
27040 DATA ABBBBBBBBBCCDDDDDDDDDDDDDD
DCCBBBBBBBBBA
27050 DATA ABBBBBBBBBBCCDDDDDDDDDDDDD
CCBBBBBBBBBBA
27060 DATA ABBBBBBBBBBBCDDDDDDCCCDDDDD
CBBBBBBBBBBBA
27070 DATA ABBBBBBBBBBBCDDDDDDDCDDDDDD
CBBBBBBBBBBBA
27080 DATA ABBBBBBBBBBBCDDDDDDDCDDDDDD
CBBBBBBBBBBBA
27090 DATA ABBBBBBBBBBBCDDDDDDDCDDDDDD
CBBBBBBBBBBBA
```

continued on page 86

THIS TAX PROGRAM WILL NOT ONLY SAVE YOU TIME,

IT WILL SAVE YOU MONEY.

For your Apple™ II, Apple II+, Apple IIe, Apple ///, IBM PC™, TRS-80™, *and your Visicalc*™.

With The Tax-Templates™ you don't have to spend an arm and a leg to hopefully save a couple of bucks. For just $89.95 you get instructions and templates for your 1040, Schedules A, B, C, D, E, G, SE, ES, W, and Investment Tax Credits, Energy Credits, Child Care Credits and much, much more.

It's current and it's good. The Tax-Templates™ author Barry D. Bayer is also the noted author of the monthly column *Visulating* in *Desktop Computing* and has written many articles for *The Apple Orchard, InfoWorld, Creative Computing* and *Microcomputing.*

It will organize, categorize and calculate your taxes the way many CPA's do.

And just in case you want to nitpick a fine point in an IRS ruling, that low $89.95 *includes* the 1982 edition of J. K. Lasser's 328-page book, Your Income Taxes.

The Tax ★★★★ Templates™

All you need is your computer, a VisiCalc™ program and the simple desire to keep your taxes and tax preparation time to a minimum.

Please don't wait until April 14 to order.

OMEGA MICROWARE, INC.

222 SO. RIVERSIDE PLAZA
CHICAGO, IL 60606
(312) 648-4844

MasterCard and Visa holders order
toll-free 1-800-835-2246

Dealer inquiries invited. Purchase of this program may be considered a tax preparation related expense. Apple is a registered trademark of Apple Computer, Inc. IBM PC is a registered trademark of IBM Corp. TRS-80 is a registered trademark of Radio Shack. VisiCalc is a registered trademark of VisiCorp, Inc. The Tax-Templates and Omega MicroWare are trademarks of Omega Microware, Inc.
© 1982 Omega MicroWare, Inc

Car Race *continued*

```
27100 DATA ABBBBBBBBBBBCDDDDDDDCDDDDDD
CBBBBBBBBBBBA
27110 DATA ABBBBBBBBBBBCDDDDDDDCDDDDDD
CBBBBBBBBBBBA
27120 DATA ABBBBBBBBBBBCDDDDDDDCDDDDDD
CBBBBBBBBBBBA
27130 DATA ABBBBBBBBBBBCDDDDDDDCDDDDDD
CBBBBBBBBBBBA
27140 DATA ABBBBBBBBBBBCDDDDDDDCDDDDDD
CBBBBBBBBBBBA
27150 DATA ABBBBBBBBBBBCDDDDDDCCCDDDDD
CBBBBBBBBBBBA
27160 DATA ABBBBBBBBBBCCDDDDDDDDDDDDDD
CCBBBBBBBBBBA
27170 DATA ABBBBBBBBCCCDDDDDDDDDDDDDDD
DCCCBBBBBBBBA
27180 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDCBBBBBBBBA
27190 DATA ABBBBBBBBCDDDDDDDDDDDDDDDD
DDDCBBBBBBBBA
28000 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
28010 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
28020 DATA ABCCCBCCCBCCCBCDDDDDDDDDDCB
CCCCBCCCBCCBA
28030 DATA ABCBBBBCBBCBCBCDDDDDDDDDDCB
CBBCBBCBBCCBA
28040 DATA ABCCCBBCBBCCCBCDDDDDDDDDDCB
CCCCBBCBBCCBA
28050 DATA ABBBCBBCBBCBCBCDDDDDDDDDDCB
CBCBBBCBBBBBA
28060 DATA ABCCCBBCBBCBCBCDDDDDDDDDDCB
CBBCBBCBBCCBA
28070 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
28080 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
28090 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
28100 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
28110 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
28120 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
28130 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
28140 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
28150 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
28160 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
28170 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
28180 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
28190 DATA ABBBBBBBBBBBBBCDDDDDDDDDDCB
BBBBBBBBBBBBA
29000 DATA AAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAA
29010 DATA ABBBABBBABBBBAAAAAAAAABBBBA
BBBABBBABBBBA
29020 DATA ABAAABABABAABADAAAAAAABAABA
BABABAAABAAAA
29030 DATA ABAAABBBABBBBADAAAAAAABBBBA
BBBABAAABBBAA
29040 DATA ABAAABABABABAADDDADADABABAA
BABABAAABAAAA
29050 DATA ABBBABABABAABADADADADABAABA
BABABBBABBBBA
29060 DATA AAAAAAAAAAAAAADDDADDDAAAAAA
AAAAAAAAAAAAA
29070 DATA AAAAAAAAAAAAAAAAAAAADAAAAAA
AAAAAAAAAAAAA
29080 DATA AAAAAAAAAAAAAAAAAAADDAAAAAA
AAAAAAAAAAAAA
29090 DATA AAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAA
29100 DATA ACCCACCCACCCACCCACACACC
CACCCAADDAAAA
29110 DATA AACAACACACACACAACAACACACA
CACACAADADAAA
29120 DATA AACAACACACACACCCAACAACCCACC
CACACAADADAAA
29130 DATA ACCAACCCACACACACAACAACACACA
CACACAADDAADA
29140 DATA AAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAABA
29150 DATA AABABABBBABABABBBABBBABAAAB
ABBBABBBAAABA
29160 DATA AABABABABABABABABABABABABAB
ABABABABABBBA
29170 DATA AABBBABABABABABABABBBABABAB
ABABABABABABA
29180 DATA AAAABABBBABBBABABAAABABBBBB
ABBBABBBABBBA
29190 DATA AAABBAAAAAAAAAAAAABBBAAAAAA
AAAAAAAAAAAAA
```

ATARI® SWAT TABLE FOR:
CAR RACE
(Modified Parameters: NU = 6 B = 300)

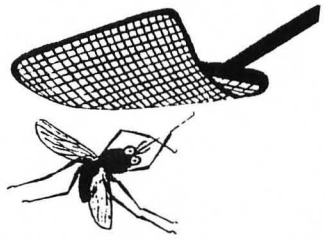
LINES	SWAT CODE	LENGTH	LINES	SWAT CODE	LENGTH
2 - 3	UU	382	22040 - 22090	BC	276
4 - 5	ZK	366	22100 - 22150	OO	276
6 - 150	DU	368	22160 - 23010	GD	276
160 - 1040	LX	285	23020 - 23070	KF	276
1060 - 1170	YR	330	23080 - 23130	EW	276
1190 - 3000	AL	340	23140 - 23190	SF	276
3001 - 3034	VQ	258	24000 - 24050	PB	276
3035 - 3051	QO	264	24060 - 24110	JX	276
3052 - 3058	YM	161	24120 - 24170	MF	276
3059 - 3067	YV	146	24180 - 25030	FP	276
3068 - 4050	OA	165	25040 - 25090	MH	276
6000 - 7050	OH	334	25100 - 25150	DD	276
7060 - 8010	FX	249	25160 - 26010	VG	276
8020 - 9113	FI	237	26020 - 26070	LM	276
9130 - 9220	UX	193	26080 - 26130	EG	276
9230 - 10011	JE	238	26140 - 26190	HB	276
10015 - 10080	QF	237	27000 - 27050	EO	276
10090 - 10209	NN	223	27060 - 27110	QE	276
10220 - 12010	ID	122	27120 - 27170	RG	276
12015 - 20010	OV	181	27180 - 28030	RL	276
20020 - 20070	EW	276	28040 - 28090	GQ	276
20080 - 20130	SU	276	28100 - 28150	TG	276
20140 - 20190	KC	276	28160 - 29010	ZG	276
21000 - 21050	VQ	276	29020 - 29070	MR	276
21060 - 21110	JM	276	29080 - 29130	CV	276
21120 - 21170	XI	276	29140 - 29190	DU	276
21180 - 22030	YU	276			

NEW FOR ATARI FROM MMG MICRO SOFTWARE

*NOW — THE **TWO** MOST POWERFUL AIDS FOR ATARI BASIC!!*

BASIC COMMANDER

- **Single key entry file commands**
  - — ENTER "D:
  - — SAVE "D:
  - — LIST "D:
  - — RUN "D:
  - — LOAD "D:
- **Single Key DOS functions from BASIC**
  - — FORMAT a disk
  - — LOCK a file
  - — RENAME a file
  - — UNLOCK a file
  - — DELETE a file
  - — DISK DIRECTORY
- **THREE PROGRAMMABLE KEYS!!**
  - — single keys programmed for your own use, even whole subroutines
- **AUTONUMBER**
  - — automatically generates line numbers for you — speeds program entry 25-75%
- **BLOCK DELETE**
  - — deletes any range of lines instantaneously!
- **RENUMBER**
  - — renumbers lines and all references
  - — extensive error trapping
  - — 3 seconds to renumber 500 lines



MMG BASIC DEBUGGER

- **TRACE through your Basic program**
  - — Single step
  - — TRACE while
  - — TRACE UNTIL
  - — change variables
  - — LIST line numbers executed
  - — examine variables' values
- **Full screen BASIC editing**
  - — scroll up or down by cursor
  - — edit your whole program easily
  - — no more LIST line number ranges
- **Split screen mode**
  - — view two parts of your BASIC program at once, and edit both!
  - — scroll each window independently
- **CROSS REFERENCE**
  - — provides a list of variables and the line numbers in which they are used in your program
- **SEARCH FOR PHRASE**
  - — search your BASIC program for any phrase, command or string of characters; let your computer do the searching for you!

EACH PRODUCT ALONE REQUIRES 16K, AND IS AVAILABLE ON DISK FOR *ONLY $34.95*

Now, the convenience of both powerful utilities together in your Atari *at once*

The Combined Basic Commander and MMG Basic Debugger requires 24K. $74.95

ADDITIONAL PRODUCTS

NECESSITIES

RAM TEST II - The fastest and most thorough memory test available for the ATARI has now been further improved! Tests not only all locations, but also tests the memory addressing system. This all machine language program takes 4 minutes to test 48K. It's the only program that tests the cartridge area of RAM. Good for new 400/800 computer owners, for testing new RAM boards and for use in computer stores to test and pinpoint bad memory locations. Bad memory locations are pinpointed so repair is as simple as replacing a chip!

Requires 8K, Disk or Cassette **$29.95**

DISK COMMANDER II - Just save this program on your BASIC disks and it will autoboat and automatically list all programs for the disk into your screen. Simply run any program by typing a single number.

Requires 16K, Disk Only **$29.95**

TUTORIALS

ASTEROID MINERS - This 50 page book and program provides for a unique intermediate to advanced tutorial. A 32K BASIC game utilizing over 25 players in player-missle graphics, machine language subroutines, a redefined character set, multiprocessing utilizing the vertical blank interrupt interval, and much more! The 50 page book included with the program documents each part of the entire program and contains the fully documented source code for both the BASIC and assembly language parts of the program. Use these routines in your own programs. These examples make it easy!

Requires 32K, Disk or Cassette **$34.95**

GAMES

CHOMPER - An all machine language arcade style game with intelligent monsters. Requires 16K Ram, 1 Joystick and nerves of steel.

Available on Disk or Cassette **$29.95**

BUSINESS/HOME

MAILING LIST - Extremely fast BASIC and machine language program. Each data disk holds over 500 files. Sort on any of 6 fields at machine langugage speed or search on any fragment of a field! Use any size labels or envelopes.

Require 40K, Disk Only **$39.95**

NEW

MMG DATA MANAGER - If you frequently find yourself looking for something, only to find it eventually right under your nose, then MMG DATA MANAGER is for you. Organize virtually anything into a computer-searchable format, and let your ATARI do the hunting for you. MMG DATA MANAGER is the first of a series of business applications from MMG MICRO SOFTWARE, all of which will share the ability to access files created by any of them. This flexible database manager will allow many fields, with machine language sorting, on any field. In addition, you have total control of the structure of your data, allowing you to design a database which you feel most comfortable. A special feature of MMG DATA MANAGER is its ability to select for a given value of any single field, or any combination of values from many fields. You could, for instance, determine who lived in Las Vegas, Nevada, and bought item #3145 from you, and whose last name began with SM, and whose telephone number began with (702) 873-4. You'll never lose track of information again! Multiple print options add to the versatility of MMG DATA MANAGER.

Requires 40K, Disk Only **$49.95**

Available At Your Favorite Computer Store

OR Send a Check or Money Order to:

MMG MICRO SOFTWARE

P.O. Box 131 • Marlboro, NJ 07746

OR CALL

(201) 431-3472

For MasterCard, Visa or COD Deliveries

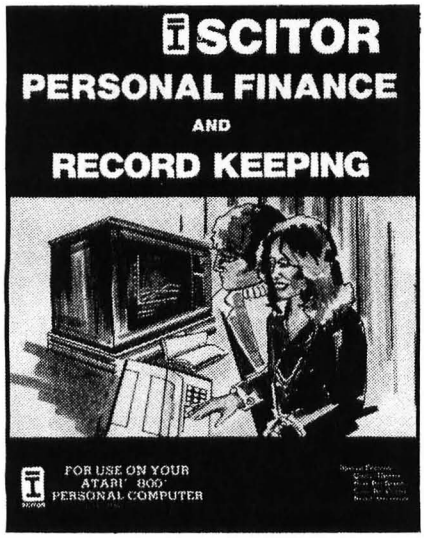
(Please Add $3.00 For Postage & Handling)

NJ RESIDENTS ADD 5% FOR SALES TAX

ATARI is a registered trademark of ATARI, Inc.

PERSONAL FINANCE & RECORD KEEPING

Reviewed by Edward & Sharon Middlebrook



"Now, a new finance system promises to increase the Atari's appeal to the homeowner and small businessman even more."

from Scitor Corporation, 710 Lakeway, Suite 290, Sunnyvale, CA 94086. System requirements: Atari® 400/800 with 40K RAM, Atari BASIC language cartridge, Atari 810 and a printer. Retail price: $79.95.

The Atari 800 has long served as a fine game machine. Its potential for personal and small business record keeping, however, has often been overlooked, even maligned. Recently, with the release of double-density disk drives and 80-column boards, Atari's flexibility has become obvious. Software houses are producing text editors, electronic spreadsheet programs, utility packages, and database systems. Now, a new finance system promises to increase the Atari's appeal to the homeowner and small businessman even more.

The Scitor Personal Finance and Record Keeping system (hereafter *PFRK*) is a comprehensive package with features to please everyone. The standard check input, search, update, and balance functions are only the beginning. Included are many report utilities (both 40 and 80 column formats), bar charts and high-resolution line graphs, monthly and year-to-date totals and summaries. Also built-in are electronic scratch pads, non-financial

"The PFRK is a well-packaged, boxed set containing two program disks and pre-punched documentation ready for three-ring binder. The documentation alone is impressive — 80 pages of instructions..."

record keepers, tables with automatic percent calculations, and (hooray) a two-way interface with VisiCalc®, that famous "what-if" utility from VisiCorp.

The *PFRK* is a well-packaged, boxed set containing two program disks and pre-punched documentation ready for a three-ring binder. The documentation alone is impressive — 80 pages of instructions and three substantial appendices. It definitely rates an "A" for effort. The examples are generally easy to understand. Several pages of instructions and an appendix explaining how to power up the Atari, boot, initialize, copy and back up disks might have been used better to illustrate setting up a finance system, but this is still a major improvement over a few xeroxed pages of instructions from a programmer.

Running the *PFRK* is easy. Boot either of the program disks with the BASIC cartridge inserted, and run the main menu program. When the main menu selection panel appears, select a number to load and execute the appropriate program.

Setting up your finance system involves the creation of two files. The first menu option sets up the category file and changes existing files. You can select any of 60 categories and assign it a label. For example, item 1 may be your house payment, 2 gas, 3 electricity, 60 income from paychecks, and so forth. This process takes only a few minutes.

Next, using the utility run by the second menu selection, you create the control file for the database and define combinations of those categories for up to ten summary fields for the Totals reports. Now, choose which month starts your fiscal year and your starting checking account numbers and balances. This takes a few minutes more, but concludes with the initializing and formatting of the actual data file. Fortunately, this is a procedure you only go through when preparing a data disk. When finished, your data disk is ready and all report and budget formats are saved for later use.

Entering data (usually financial transactions such as checks, deposits, and withdrawals) is a function of another menu selection. You input the usual check information such as date, amount, and name, and assign the item to one of the previously defined categories. The transaction is then given one of nine possible status types (deposit outstanding, item returned, and so on). Finish the input, verify the data, and the transaction is recorded. If you cannot remember which category number you assigned to the car insurance, just enter a command which lists all of the categories defined and then returns you to the point you left. Between transactions, you may request an accounting of the balances by status type. This, along with the starting balance input at initialization time, gives an immediate verification of your checkbook balance.

The input utility has many other practical features. Extra categories may be defined as the minimum or maximum value from one of the other categories, or may be the total number of items in that category. You may therefore set up a category for Atari purchases, and have fields defined which automatically tell you the smallest and largest purchases you made for the month, as well as how many Atari purchases were made.

The input screen has a built-in scratch pad for numeric calculations, and allows up to ten categories to be paid with the same check. As an extra aid, a constant, such as a state tax rate, may be kept in memory for occasional calculations.

Edit/Update

The same menu selection can put you into Edit/Update mode. Here you can change errors in all existing records, such as amount, date, name, category, and status. Records may also be deleted. The search functions allow selection by any of the fields, with Next (record), Front (of file), Backwards (record), and Last (of file) being some of the available commands. Printer options are also supported, such as SP (search and print) and PA (print all). You also use this utility to clear transactions when you receive your monthly statement. Simply change the records status from Outstanding to Received and check your balance against the statement.

The fourth choice on the main selection menu is the Totals and Monthly Reports utility. This program generates monthly totals for each category and summary totals as previously defined in the initialization process. After the totals are calculated, they are stored on a totals disk for later reports and as the basis for input into VisiCalc. Reports are generated by month or year-to-date, with itemization of categories and grand summaries. A nice touch is the choice of row or column formats. Users with 40-column printers may get reports, while those with 80-column printers may produce full-page reports or use the 40-column format.

The final option of the main menu invokes the Interactive Graphics programs. The programs

"One of the most exciting features in the PFRK is the two-way interface to VisiCalc. A set of menu options converts the Totals file to VisiCalc format, or converts a VisiCalc file to PFRK format."

allow either Graphics mode 7 line plots of up to ten items at once, or a bar chart of two items. Both use the files from the Totals disk. You can enter formulas that combine the categories and plot the output. Forecasting uses a built-in least-square fit function. The plots are automatically labeled with the name of the category label stored in the Totals file. Also, a table may be displayed listing the monthly totals of any two selected categories. This table will compute yearly totals, average values, and percent comparisons automatically. The graphing functions are fairly quick and easy to understand. The only fault we find is that you cannot save the plotting formulas to disk. You must input these every time you use this utility.

One of the most exciting features in the *PFRK* is the two-way interface to VisiCalc. A set of menu options converts the Totals file to VisiCalc format, or converts a VisiCalc file to *PFRK* format. We found these processes very easy to use. It was fascinating to take our real data and play "what-if" with VisiCalc. There are a few compatibility considerations to remember when designing the VisiCalc file to convert to *PFRK* format. These considerations, along with specific instructions for the conversion processes, are spelled out in an appendix. Anyone at all familiar with VisiCalc can appreciate the flexibility this gives the *PFRK*. We are still discovering new uses for this feature.

Non-financial Record Keeping

One appendix is devoted to creating and maintaining a jogging log. This log would track the

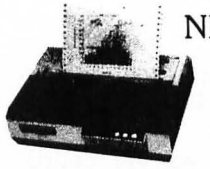
APPLE ATARI **COMPARE OUR PRICES!!** IBM

Rana Systems

State of the art design brings you the most advanced Floppy Drive available for your Apple II! 100% Compatible with Apple II Disks, Controller, Software, CP/M, Pascal.

	W/Controller	W/O Contoller
Elite 1 (163K)	**$409.00**	**$339.00**
Elite 2 (326K)	**$599.00**	**$509.00**
Elite 3 (652K)	**$729.00**	**$669.00**

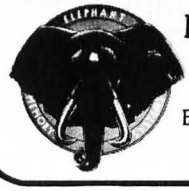
Controller (1-4 Drives, 3.2 or 3.3) **99.00**



NEC PC-8023A **$459.00**

100CPS, 8 Types, Graphics, Tractor/Friction

Call for Interface Info.!!



REMEMBER

100% Certified Error Free, Hub Rings

ELEPHANT DISKETTES: **23.95**/10 **219.50**/100

KRELL LOGO — Complete package, with 2 copies of KRELL Logo, Utilities disk, Alice in Logoland, 4 manuals, wall chart. **$129.95** APPLE II

ENTERTAINMENT SOFTWARE

	Retail	Our Price		Retail	Our Price
Apple Panic Ap, At, IBM	29.95	**20.95**	**Gorf** At	39.95	**29.95**
Arcade Machine Ap	54.95	**38.95**	**Knight of Diamonds** Ap	34.95	**24.95**
Canyon Climber Ap, At	29.95	**20.95**	**Pig Pen** Ap, At, IBM	29.95	**20.95**
Choplifter Ap, At	34.95	**24.95**	**Preppie** At	29.95	**20.95**
Crystal Caverns Ap	34.95	**24.95**	**Starcross** Ap, At, IBM, TRS	39.95	**28.95**
Dark Crystal Ap	39.95	**28.95**	**Temple of Apshai** Ap, At, IBM, TRS	39.95	**28.95**
Deadline Ap, At, IBM, TRS	49.95	**35.95**	**Trivia Trek** At	29.95	**20.95**
Deadly Secrets Ap, At	34.95	**24.95**	**Ultima II** Ap, At	59.95	**41.95**
Dogstar Raiders Ap	29.95	**20.95**	**Ulysses & Golden Fleece** A, A, IBM	34.95	**24.95**
Flip Flop Ap	34.95	**24.95**	**Wizard of Wor** At	39.95	**29.95**
Frogger Ap, At, IBM	34.95	**24.95**	**Wizardry** Ap	49.95	**35.95**
Gold Rush Ap, At	34.95	**24.95**	**Zork I, II or III** Ap, At, IBM	39.95	**28.95 ea.**

BUSINESS, UTILITIES, LANGUAGES

Ampersoft Ap	49.95	**35.95**	**GraForth II** Ap	75.00	**52.95**
Atari Basic At Cart	59.95	**44.95**	**Lisp Interpreter** Ap, At	124.95	**87.95**
Atari Pilot (Home Package)	79.95	**59.95**	**Microsoft Basic** At	89.95	**66.95**
Basic Compiler Ap, At	99.95	**69.95**	**QS Forth** At	79.95	**56.95**
Forth II Ap	69.95	**49.95**	**TransForth II** Ap	125.00	**87.95**

EDUCATIONAL SOFTWARE

Bumble Games (4-10) Ap	60.00	**42.95**	**My First Alphabet** (3-6) At	34.95	**24.95**
Counting Bee (5-10) Ap	29.95	**20.95**	**Sammy the Sea Serpent** (4-9)	23.95	**17.95**
Face Maker (4-8) Ap, IBM	34.95	**24.95**	**Snooper Troops I or II** Ap, IBM	44.95	**31.95 ea.**
Gertrude's Puzzles (6-A) Ap	75.00	**52.95**	**Spelling Bee/Primer** (5-10) Ap	39.95	**27.95**
Juggle's Rainbow (3-6) Ap	45.00	**31.95**	**Story Machine** (5-9) Ap, IBM	34.95	**24.95**
Master Type Ap, At	39.95	**27.95**	**Word Race** (9-A) Ap	24.95	**17.95**

20-30% DISCOUNTS Call for Prices on new releases and items not listed. We have it!

Most Orders shipped same day. Add $2.00 Shipping (Software) or 3% (Hardware). Add $2.00 for C.O.D.'s (Max $100.00). CA Residents add 6% tax. Personal Checks take 10-15 days to clear bank. Foreign orders add 10%, Min. $10.00. Prices subject to change without notice.
Call or write for Free Catalog. Over 1,000 Items at 20-30% Discount. Please Specify Computer.

COLLINS COMPUTING

Box 6424 • San Bernardino, CA 92412
SOURCE ST9386
CALL: 714-783-3155

VISA MasterCard

"The Scitor Personal Finance and Record Keeping system is a worthwhile addition to anyone's software library, with features and utilities for everyone."

number of miles run, total number of runs, and minimum and maximum distances. These totals could then be plotted against preset goals.

Drawbacks

We were very impressed with the *PFRK*. It does, however, have some drawbacks. First and foremost, the entire package is in BASIC. Being assembly language snobs who bought our database system primarily because it was coded in assembler, this was a disappointment. However, excluding the formatting and search functions, BASIC speed is adequate. Also, the number of records per disk appears preset to 340. The database may be spanned to multiple disks, but between two program disks and a couple of data disks you will find yourself coveting a second disk drive. Paradoxically, what was the primary sore spot was also one of the best features. Since the system is written in BASIC and is totally unprotected, it is easy to modify. In fact, the license that comes with the *PFRK* gives the purchaser the right to make backup copies of the program disks and any modifications he wishes. The customer is *not* given the right to photocopy the documentation. This is a reasonable position on the much-contested question of software protection. Flowers to Scitor for their user-friendliness and sense of responsibility to the customer.

The *Scitor Personal Finance and Record Keeping* system is a worthwhile addition to anyone's software library, with features and utilities for everyone. It is easy to use, well-documented, and may give you the answer to that oft-asked question "Your computer is nice but what good is it?"

CLEARANCE! CLEARANCE! CLEARANCE!

Was ~~$19.95~~

Unlock the hidden power of your TRS-80® Model I for fast and easy programming! All you need to know is in-Pathways Through The ROM

Includes:

- **SUPERMAP** from Fuller Software ($18.95)
- **TRS-80 Disassembled Handbook** by Robert Richardson ($10.00)
- **HEXMEM** by John Phillip
- **Z-80 Dissassembler** by George Blank
- **Guide to Level II BASIC and DOS Source Code**

Limited quantities available. Place your order now!

YES! Send me _____ copies of Pathways Through The ROM. I'm enclosing $9.95 plus $1 shipping and handling for each copy ordered.

☐ Check ☐ M.O. ☐ VISA ☐ MasterCard

Name _____

Address _____

City _____ State____ Zip____

Name of Cardholder _____

MC# and Interbank#/VISA# _____

Exp. Date_____

Signature _____

Send Orders to: SoftSide Publications, Inc., 6 South Street, Milford, NH 03055

Price subject to change without notice. TRS-80 is a registered trademark of The Tandy Corporation.

Your ATARI® Computer

Reviewed by James V. Trunzo

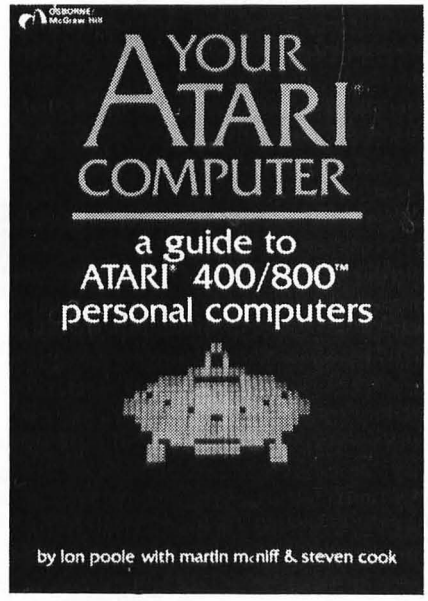
by Poole, McNiff, and Cook (OSBORNE/McGraw-Hill, 630 Bancroft Way, Berkeley, CA 94710) Retail price: $15.00.

To state that *Your Atari Computer* is a "must read" for novice programmers who own either the Atari 400 or 800 is a gross understatement. If you have worked your way through Atari BASIC and the Atari Operators' Manual only to discover that you don't even know mundane fundamentals such as how to ring the Atari internal buzzer, that "?" is an abbreviation for the command PRINT, and that Sound 0,0,0,0, shuts off voice one, *Your Atari Computer* is the proverbial life preserver thrown to a drowning man.

In 458 pages, *Your Atari Computer* covers everything found in the Atari computer handbooks, and does so in a clearer and more complete way. One small example of this is the way *Your Atari Computer* handles tabs. The Atari Operators' Manual gives this subject a total of seven written lines and two 1-line examples. Atari BASIC fails to address it directly at all. *Your Atari Computer* allows thirty-nine lines of explanation, plus several short programs and an illustration demonstrating the use of the TAB key. Furthermore, if you were to look for TAB in the index of either Atari BASIC or the Atari Operators' Manual, you wouldn't find it. (After some searching, you would discover TAB in the Operators' Manual under the heading KEYS.) No such time-consuming search is necessary when using the excellent index found in the Osborne publication.

To continue to compare *Your Atari Computer* with the other Atari handbooks would be to belittle the efforts of those who put together Atari BASIC and the Atari Operators' Manual, both of which are well-done, but rather limited. Instead, let's take an overall look at what *Your Atari Computer* has to offer. The book can be divided into six distinct parts (though it is not set up this way in the Table of Contents), with subdivisions within those parts.

Part One is aimed at the true novice, taking him through the actual hook-up procedures of the computer and its peripheral devices, including the Atari 825 printer and 810 disk drive, all of which are illustrated. Also included in this section are in-depth explanations of



fundamental operating procedures such as loading from a cassette, booting DOS, and basic editing, to name a few. A thorough examination of the keyboard is given during the course of Part One, too, pointing out the function of the many special Atari keys.

Part Two consists of two very well-done sections on programming skills. The beginning of this section deals with everything to which a novice should be exposed in a concise, well-documented manner. Major topics such as the use of arrays and strings, subroutines and variables, and input-output instructions are well covered. Minor (but important) areas such as interrupting and continuing a listing, abbreviating the PRINT statement, chaining programs, and advanced editing techniques are given the attention they deserve. How many times have you listed and re-listed a program from beginning to end, wishing that you could interrupt and then continue the listing? If that information is available in the Operators' Manual, it's well-hidden! *Your Atari Computer* provides it within the first 50 pages.

The second section of Part Two focuses on advanced programming techniques and can benefit the more experienced programmer as well as the beginner. Subject matter making up the advanced portion of the book covers such diverse material as string concatenation, variable storage, columnar output, resetting margins, input masks, etc. Especially interesting and of value is the work done on creating a sophisticated error-handling routine. An entire routine is developed in a step-by-step approach, with complete explanations as to what lines are doing what within the utility program. Another valuable area covered in detail is the use of joysticks to restrict input responses, select from a menu, and provide numeric input. The section wraps itself up neatly by giving a brief introduction to the USR function and also discussing ways to debug a program, increase the speed of a program and prevent inadvertent crashing by hitting the BREAK key, for example.

Part Three of *Your Atari Computer* presents, in great detail, the operating procedures and characteristics of the Atari 410 recorder, the Atari 810 disk drive, and the Atari printers. This section is also nicely done. One can derive much benefit from reading the section on something as basic as the 410 program recorder. When dealing with this peripheral, *Your Atari Computer* focuses on how to use the cassette medium to function as a

data filing system, something especially useful if you cannot presently afford a disk drive. The material on the various Atari printers and the Atari disk drive is copious. While I have never had access to the Atari handbooks dealing with the hardware in question, they couldn't be any more detailed or informative than the material presented on the pages of Chapters Six and Seven, dealing with the printer or disk drive, respectively. Even if you don't, as yet, own a printer or disk system, the reading is provocative and beneficial. It explains the abilities, advantages and limitations of those items, providing you with the knowledge you will eventually need if you decide to expand your computer system.

All of this is fine and dandy, you say. But what about the good stuff! Does *Your Atari Computer* touch on the much heralded Atari graphics and sound capabilities? Does it say anything about the infamous and mysterious player missile graphics? Does it deal with mixing display lists? Of course! *Your Atari Computer* devotes 53 pages to graphics alone, and deals with sound as a separate chapter altogether. The graphics unit is presented in two segments: the first part dealing with introductory graphics and the second with advanced techniques. For novice programmers, *Your Atari Computer* takes you on a step-by-step adventure through the uses of SETCOLOR, the COLOR statement, color registers in Modes 1 and 2 and, of course, the use of the PLOT and DRAWTO commands. Also covered in more than adequate detail are such specialties as using the XIO fill command, inserting text on the graphics screen, and various graphic applications. As in all sections of *Your Atari Computer*, a variety of useful demonstration programs accompany the text. Advanced graphics discuss such complexities as display list creation, character set animation, and player missile graphics. Each topic contains enough information to give the reader a general idea of what is involved when dealing with a particular area and usually provides a program listing demonstrating the topic. For example, in the section on display lists, a complete utility (Display List Loader) is included for the reader's use. The section on player missile graphics contains a short program for defining a player, another program for showing simple player movement, and instructions for laying out the player missile graphics table. While the various topics covered in the advanced graphic sections could use more explanation, you must realize that each of these areas could be the subject of its own book. With this in mind, *Your Atari Computer* handles the topics quite adequately.

Chapter 10, "Sound," delves into all the basic areas of sound that a new programmer should understand. Distortion is very nicely handled, complete with a handy reference chart that clearly shows the results of each distortion value in silences and secondary tones. Pitch, duration, and voice also receive their due. As a bonus, sixteen short sound effects programs are provided that simulate everything from a phone ringing, to a rocket ship takeoff, to an explosion of epic proportions.

A very handy alphabetized list of statements and functions makes up Part Five. Each entry is accompanied by a format and example, plus one to three paragraphs explaining the use of the particular entry being addressed. Where needed, the entries are supplemented by tables and charts which aid in the explanation of the entry itself.

Finally, Part Six of the text consists of several appendices, each of which, if not immediately useful, will become so as your programming knowledge increases. The nine appendices cover such diverse areas as error messages and explanations, useful PEEK and POKE locations, STATUS statement codes, a bibliography, and the comprehensive index mentioned earlier in this review.

The subtitle of *Your Atari Computer* is "A Guide to Atari 400/800 Personal Computers." Priced at $15.00, you couldn't purchase a better "guide" to lead you out of the computer wilderness that all of us become lost in from time to time. Thorough, concise, and easy to read, *Your Atari Computer* is guaranteed to save you time, effort, and frustration while awakening you to the true potential of your computer.

★ FREE SHIPPING ★
Within Continental 48 States

MICRO IMAGES

SOFTWARE FOR THE TRS80
MAXI CRAS · Mod I/III $84.95
LAZYWRITER Mod I/III $159.95
NEW SCRIPT 7.0 Mod I or III $114.95
LDOS -Ver. 5.1 Mod I or III $114.95
DOSPLUS - 3.4S/3.4D/3.4III $119.95
MULTIDOS - Improved Version ... $84.95
GEAP - $42.95 — W/Dot Writer 1.5 $84.95
SUPERUTILITY + W/Backup $59.95
DATA-WRITER 2.0 New Vers. I/III $129.95

VOICE BOX by Alien Group
ATARI - Tape or Disk (Specify) ... $149.95
APPLE - with Firmware, Rom $189.95
APPLE -without Firmware $119.95

MICROSOFT - For The Apple
SOFTCARD $349.95
RAMCARD $174.95
SOFTCARD PREMIUM SYSTEM . $674.95
Includes Softcard, Ramcard, & Videx Videoterm 80 Col.
FORTRAN 80 - Req. Softcard $174.95
A.L.D.S. - Req. Softcard $114.95

MODEMS
SIGNALMAN for Atari $94.95
LYNX - For TRS80 Mdl I/III $239.95
HAYES MICRO MODEM II $334.95

C. ITOH PROWRITER - $499.00
Parallel Printer 120 CPS 10" carriage

LNW-Doubler 5/8
Includes Dosplus 3.4D $205.95
LNW EXPANSION II - $375.00

RIBBONS

ZIP BOX RELOADS	½ Dz.	Dz.
Epson MX 70/80-20 Yds	24.00	42.00
Epson MX 100-30 Yds	30.00	52.00
NEC/Prowriter	21.00	36.00
Centronics 730/737/739/779 or LP-I/II/IV-16Yds	18.00	32.00

All ZIP BOXES are individually sealed black nylon and require no rewinding. Epson Reloads also available in red, blue, brown, green & purple. Any mix allowed.

CARTRIDGES	Each	Dozen
Epson MX 70/80	8.95	90.00
LP-III/V	6.50	70.00
Centronic 702/03/04/53	11.00	120.00
RS DSY WHL II Multi Strike	6.50	70.00
Diablo Hytype II Multi Strike	6.50	70.00
Qume-300,000 chr Multi Strike	6.50	70.00
Nec SPIN H-Yield Multi Strike	7.00	75.00
MCRLNE 80/82A/83A Spl	N/A	24.00

Minimum order 3 cartridges - any mix. For smaller quantities add $1.50 per order. All our reloads and cartridges are manufactured by one of the oldest and most reputable ribbon Mfg's. in the country.
★ ★ ★ ★ ★ ★ QUALITY GUARANTEED ★ ★ ★ ★ ★ ★

ORDERING INFORMATION
No credit cards at these low prices. Add $2.00 on all COD orders. Certified Ck/MO/COD shipped immediately. Please allow 2 weeks for personal checks. For extra fast service phone in your COD order. Free shipping within Continental 48 states via UPS ground. For Canada, Hawaii, Alaska, applicable shipping and insurance charges apply. Prices subject to change without notice. New York State residents please add appropriate sales tax.
The items listed above are a cross-section of our product line. We carry the full line of most companies listed in the ad, plus much more. SEND FOR YOUR FREE CATALOG.

146-03 25th Road, Dept. S
Flushing, New York 11354
(212) 445-7124
Mon-Fri 10 AM-9PM
Sat. 10AM-5PM

APPLESOFT® EXTENSIONS 2

by Kerry Shetline

***Applesoft Extensions 2* is a programming utility for the Apple II® with at least 32K RAM and DOS 3.3. It is included as the bonus program on Issue #37 Apple DV. See the Bind-in Card elsewhere in this issue to order this month's disk.**

Applesoft is a trademark of Apple Computer, Inc.

Applesoft Extensions 2 is an expansion of *Applesoft Extensions,* which appeared in issue 31. I have decided that my routines are important enough to steal the ampersand (&) vector, rather than rely on CALL statements. This change provides a practical way to implement a greater number of new statements, especially the IF-THEN-ELSE-CONT feature. *Applesoft Extensions 2 (AE2)* occupies a bit more memory than its predecessor — 2048 bytes to be exact. Despite the increased size, *AE2* remains totally relocatable.

The procedure for loading *AE2* is simple. The first line of a program would be as follows:

```
100 PRINT CHR$(4)"BRUN
AE2.LDR"
```

Any line number can be used, as long as it is the first line. It is important to use CHR$(4) rather than a pre-defined D$, because the loader routine clears variables, and possibly lowers HIMEM. D$ would be null and would have to be defined again.

After the loader routine executes, the necessary code for *AE2* resides safely above HIMEM, and the & vector points at the start of the code. All of the *AE2* statements are then available for use.

Note: The syntax for the statements discussed in this article will be defined using metasymbols. If you are unfamiliar with metasymbols, please read the section "Syntactic Definitions and Abbreviations" on page 30 of your Applesoft manual.

The Alternate & Vector

Although *AE2* uses the & vector, an alternate vector is available. This is the &# vector. If you have other & utilities that you wish to use, replace the & with &#. The &# vector is set with the statement &=expr, where expr is the address of a Machine Language routine. Let's look at a typical example. Suppose you had a utility that would mangle any selected track on your disk. If the routine were located at $6000, you would normally do the following to destroy track 5:

```
100 POKE 1010,76: POKE 1011,0:
POKE 1012,96: &5
```

However, using AE2, you would enter:

```
100 &=24576: &#5
```

You only have to do the &= one time, of course, and it will remain

pointing at the same routine until you do another &=.

The &BEEP Statement

You can easily produce sound with the &BEEP statement. The syntax is:

&BEEP;expr1,expr2[,expr3]

Expr1 is the pitch. Permissible values are 0-255, 255 being the lowest pitch, and 0 the highest. Expr2 is the duration, which is also in the range 0-255. The shortest duration is 0, and the longest is 255. Expr3 is an optional repetition factor.

Unlike most sound routines, &BEEP makes it possible to break the program with a CTRL-C while the sound is still being created, or cut the sound short with an ESC.

&IF-THEN-&ELSE-&CONT

You probably have heard of IF-THEN-ELSE. Many Apple users wish this construct were available. Now, you may be wondering what CONT has to do with IF-THEN-ELSE. The &CONT statement is a new logical construct unique to *AE2*. While &ELSE is a point where execution goes if the &IF condition is false, &CONT is used in *AE2* to indicate a point where execution will go regardless of whether the condition is true or false. The syntax is as follows:

&IF condition THEN linnum | statements [:&ELSE: statements] [:&CONT: statements]

Suppose you wanted to simulate flipping a coin ten times. Using &IF, you could do it like this:

```
100  FOR X = 1 TO 10: &  IF  RND
     (1) < .5 THEN  PRINT "HEADS"
     : & ELSE: PRINT "TAILS": &  CONT
     : NEXT
```

No matter what happens, the program will reach the NEXT statement, allowing you to complete the operation on one line, and avoid the sloppy programming technique of having more than one NEXT statement for a single FOR.

When using &IF, neither &ELSE nor &CONT are required. In fact, you could use &IF by itself, and it would act as an ordinary IF statement.

One of the more interesting uses of &IF is nesting &IF's within other &IF's. It is a rather advanced technique, and may be a bit confusing to the novice programmer. The following rules govern nesting:

- An &ELSE matches with the most recent unassigned &IF.
- An &CONT matches with the most recent unassigned &IF, if there are no &ELSE's in between, or to the same &IF as the most recent &ELSE.
- Never use a "double" &ELSE. There should always be an &IF or an &CONT between two &ELSE's.

If you do enough nesting, you see that the relationships among &IF's, &ELSE's, and &CONT's could become quite complicated and obscure. To help you sort through the more elaborate nesting situations, the DV this month includes an update to my program, *List Formatter (SoftSide*, May 1982). The update provides proper logical indenting to show the relationships among &IF, &ELSE, and &CONT.

To update *List Formatter,* first make a backup copy. Then, use FID to copy the files LFORM.M and LFORM.B.UPDATE from this month's disk onto your disk. Once the files are transferred, EXEC the copy of LFORM.B.UPDATE on your disk. This completes the update.

I shall use *List Formatter* on the following examples:

The rule about double &ELSE's may be unclear. The program below demonstrates the use of a double &ELSE:

```
230 &IF DONT=EVER THEN
      &IF UWANT=THEPROG2WORK THEN
        PRINT "TRY THIS EXAMPLE":
      &ELSE:
        PRINT
        "RESULTS ARE UNPREDICTABLE":
      &ELSE:
        PRINT "THIS PROGRAM VOID":
        PRINT "WHERE PROHIBITED."
```

As you can see, the &ELSE's do not line up properly. This shows that the example does not work. Such a construct should never be used. Here is an example of the proper use of the &IF statement:

```
100 &IF DAY=FRI THEN
      PRINT "IT'S PAYDAY!":
    &ELSE:
      &IF DAY=THU THEN
        PRINT "IT'S ALMODT PAYDAY!":
      &CONT:
    &CONT:
    &IF DTE/2=INT(DTE/2) THEN
      PRINT "DON'T ";:
    &CONT: PRINT "WATER THE PLANTS"
```

This program assumes two things: that you get paid on Friday, and that your plants are on an odd-even rationing system. If it's Friday, you get the message "IT'S PAYDAY!". If it is Thursday, the message "IT'S ALMOST PAYDAY!" is displayed. If it's neither day, no reference to payday is made. In any event, execution continues to the matter of the date (DTE). On even numbered days, you are instructed "DON'T WATER YOUR PLANTS," and on odd numbered days, "WATER YOUR PLANTS."

You may also use ordinary IF statements with &IF's. Remember, however, that &ELSE and &CONT don't match with IF statements. If an IF statement is evaluated as false, execution will fall through to the next program line. Here is an example:

```
300 &IF MOOD=PHILOSOPHICAL THEN
      IF THINK=TRUE THEN
        PRINT
        "I THINK, THEREFORE I AM":
    &CONT: INPUT
    "WHAT'S FOR DINNER? ";MEAL$
```

If you're not in a philosophical mood, the program skips directly to the question, "WHAT'S FOR DINNER?". However, if you decide to contemplate existence, the computer considers your ability to think. If you are fortunate, you will receive the message "I THINK, THEREFORE I AM," and then be asked about dinner. If the computer does not care to recognize your higher mental processes, no great wisdom is imparted, and the question "WHAT'S FOR DINNER?" is skipped.

MID$ Assignment

The &MID$ statement allows you to place a string directly into another string. The syntax is:

&MID$(svar,expr) = sexpr

The string changed is svar. Expr indicates the position in svar where sexpr will be placed. Expr may have any value from 1 to the length of svar. Sexpr can be any string expression so long as placing sexpr into svar does not create a string longer than 255 characters. If you are confused by all of these metasymbols, here are two examples:

```
150 A$ = "AMPLE":
&MID$(A$,2) = "P"
160 B$ = "BACH":
&MID$(B$,3) = "SIE"
```

In line 150, "M" is replaced by "P," and the resulting value of A$ is "APPLE." In line 160, "CH" is replaced by "SI," and the remaining letter "E" is placed at the end of B$. The resulting value of B$ is "BASIE."

Trapping RESET in BASIC

The statement for trapping RESET will only work on Apples with the Auto-start ROM, because the old monitor ROM does not support the ability to trap RESET.

If you have an Auto-start ROM, this is how you intercept RESET within a BASIC program:

&ON RESET GOTO linnum

Once this statement has been executed, pressing RESET transfers control to the specified line number. The Apple does not go directly to the routine pointed to by the RESET vector. It does a number of things first: It selects a normal text window, sets the I/O hooks to IN#0 and PR#0 (which disconnects DOS), and then beeps. After all of this, the Apple is ready to look at the RESET vector. When you are using &ON RESET GOTO, *AE2* sets the RESET vector to a routine that will reconnect DOS and clear the BASIC stack before going to the line number you specify. When you are where you want to be, you will have DOS available, and will not be within any FOR-NEXT loops or subroutines.

If you wish to return to normal RESET handling, use this statement:

&ON RESET CLEAR

Keep in mind that RESET causes the Apple to stop whatever it is doing immediately. If you hit RESET at the wrong time, a variable that Applesoft was changing might end up with a bizarre value, or another equally critical task might go uncompleted. Above all, avoid hitting RESET while the Apple is writing to a disk, as this will almost certainly do something unspeakable to the unfortunate floppy.

RESTORE To A Specified Line

AE2 provides the ability to restore the data pointer to a line number. The syntax is as follows:

&RESTORE GOTO linnum

Rather than restoring to the beginning of all data, you may restore to a specific line. The line you restore to should start with a DATA statement, or you will get an OUT OF DATA error.

Setting The Screen Window

Setting the screen window on the Apple can be tedious. You have to do up to four POKEs to set the window, and if you accidentally POKE in a bad value, you can hang the computer, or have screen information go running off into your BASIC program doing all manner of nasty things. To provide an easier and safer way of setting the text window, *AE2* has the &SCRN statement, with the syntax as follows:

&SCRN([left margin],[right margin],[top margin],[bottom margin])

The left and right margins must be values from 1 to 40; The top and bottom margins should be values from 1 to 24. All parameters are optional. Any margin not specified will remain unchanged. When a window is set, the cursor moves to the left-most position on the top line.

Suppose you wanted to set a small window in the middle of the screen. You could do it like this:

&SCRN(11,29,6,19)

To set just the left and right margins:

&SCRN(5,20,,)

Note that when not all margins are specified, all of the commas must remain. If you were setting only the bottom margin, it would look like this:

&SCRN(,,,20)

The 16-bit POKE

AE2 has a statement, &POKE, for placing 16-bit values into memory. The syntax is:

&POKE expr1,expr2

Expr1 is the address at which the low-order byte of expr2 is placed. The high-order byte of expr2 is placed at expr1 + 1. A typical use of &POKE might be setting up a shape table. If you had a shape table at 16384, you could set up the shape table pointer with:

&POKE 232,16384

Swapping Variables

The &SWAP statement provides the ability to swap the values of any two variables of the same type. You can also swap more than one pair of variables in a single &SWAP statement. &SWAP has the advantage of swapping variables about 50% faster than a swap using a hold variable. The syntax is:

&SWAP;var1,var2[{;var1,var2}]

Here is an example:

```
A$ = "WHERE": B$ = "ELSE":
&SWAP;A$,B$: PRINT A$;B$
```

If you were to type this in, the output would be "ELSEWHERE".

Using &SWAP and &IF, you can make a very short and efficient sort routine. This example sorts ten high scores and the names of the people who made them:

```
210 FOR J=0 TO 8:
      PTR=J:
      FOR K=J+1 TO 9:
        &IF SCR(K)>SCR(PTR) THEN
          PTR=K:
        &CONT:
      NEXT K:
      &SWAP:SCR(J),SCR(PTR):NAME$(J),
      NAME$(PTR):
    NEXT J
```

The &WAIT Statement

This command does absolutely nothing. It is useful, however, because it does nothing for a specified duration of time. The syntax is:

&WAIT expr

Expr is the approximate number of seconds for the time delay. You have a resolution of .1 second, and a range of delays from 0 to 6553.5. The &WAIT statement is meant to replace the use of FOR-NEXT loops for time delays. You may break the execution of an &WAIT with CTRL-C, or end it early by pressing ESC.

&LINE INPUT

This statement allows you to input a string of up to 255 characters into a string variable. It differs from the normal INPUT statement because it allows you to enter the characters quote, comma, and semicolon, does not remove leading or trailing spaces, and can receive an input of up to 255 characters. (A normal INPUT statement will truncate any input beyond the 239th character.) When using &LINE INPUT, the Apple's normal ESCape sequences remain available.

This statement is very useful for reading text files. Even when characters would cause an error with a normal input statement, &LINE INPUT reads them in. This method is much faster than the use of GET statements, yet does not limit the type of data that may be read from a file. For &LINE INPUT to reliably read text files, however, there should be no more than 255 characters between carriage returns, or everything from the first character to the 256th is ignored.

The syntax for &LINE INPUT is:

&LINE INPUT svar

&TEXT INPUT

Occasionally, you need to read a file with large numbers of characters between carriage returns. This is often the case when reading a text file created by a word processor. You can use &TEXT INPUT to read such files. &TEXT INPUT will read 255 characters from the input source you have selected, most often a text file. The strings created by this form of input will actually contain imbedded carriage returns. The syntax is:

&TEXT INPUT svar

You may ask, "What happens to the last few characters if the number of characters in a file isn't a multiple of 255?" The DOS produces an end-of-file error, the normal result of attempting to read past the end of a file, and those last few characters are still in the input buffer. As long as you don't do anything to affect the input buffer, like a DOS command or another input operation, you may get those characters with this form of &TEXT INPUT:

&TEXT INPUT svar;

Presuming you have opened a text file and have DIMensioned the array T$ to a size capable of holding the file, you may read the entire text file into the array T$ like this:

```
30 N=0: ONERR GOTO 50
40 &TEXT INPUT T$(N): N=N+1: GOTO
   40
50 &TEXT INPUT T$(N);
```

APPLE II™ INCOME TAX PROGRAM

$29.00

...On Diskette For Apple II Plus, 48K

Simple-to-use tax estimator gives quick readout of Federal Income Tax for individuals. You enter income, deductions, credits etc., and the computer calculates your taxes. No knowledge of accounting or computer expertise is required. The program also saves your entries for later recall. It allows you to adjust individual entries to evaluate the effects of investments and changes in income. An optional printout routine generates hard copy summaries (the program does not print IRS forms). FREE update to 1983 tax rates becomes available after April 15, 1983.

— A $45.00 VALUE —
THIS SPECIAL OFFER EXPIRES MAR. 31, 1983

NAVIC SOFTWARE
BOX 14727
NORTH PALM BEACH, FL 33408
(305) 627-4132

Free Catalog Included

MOVING?

If you're planning to move, please let us know at least six weeks in advance. This will help us to change your address insuring you with prompt and accurate service on your subscription. Attach your current mailing label filling in your name and NEW address in the space provided.

Attach
old label
here

Name _____

New Address _____

City _____ **State** ____ **Zip** _____

Send old label with your name and NEW address to:

SoftSide
100 Pine Street
Holmes, PA 19043

The &INPUT Statement

This form of input creates a better user interface when data is requested from the user. It is for keyboard input only.

Input time is a very dangerous time for many programs. Your screen display can be destroyed by messages like "?EXTRA IGNORED" or "?REENTER." Many cancelled inputs could scroll the screen. There are also problems for the user, especially non-programmers. Characters that do not go away when you backspace make some people unsure whether the characters after the cursor will be entered or not. Input cancellation when too many characters are typed can be very frustrating.

&INPUT allows you to specify maximum input length, what characters can be input, what characters can't, what happens when you hit RETURN, whether or not you can break with CTRL-C, and many other options. It also displays the input line differently. The cursor is a blinking underscore, or a blinking dash to indicate that you are at the end of input. When you backspace, the last character disappears, but reappears if you hit the right arrow. Cancelling the input line with CTRL-X will not skip to the next line, but will erase what you have typed and put the cursor back at the input prompt. After a CTRL-X, the right arrow can still retrieve characters already typed.

The metasymbol syntax for &INPUT is rather complicated, but I shall clarify with a detailed explanation afterwards.

```
&INPUT [([expr] [,[B][C][E]
[H][K][N][S][, | ; sexpr]]);]
["prompt";] var
```

The first optional parameter, specified by expr, is the maximum input length, which may be in the range 1-255. If it is not specified, the default is 255. What occurs when you reach the maximum length will be explained with the "E" option.

A group of seven options may follow the length parameter. They perform the following functions:

B — Allow CTRL-C to cause a break.
C — Allow the input of control characters. Control characters are displayed in inverse when typed.
E — If specified, the input line is entered automatically when the maximum length is reached. The cursor becomes a dash instead of an underscore when waiting for the last character. If the option is not specified, the cursor becomes a dash after the last character is entered, and there is no response until there is a carriage return, a backspace, or a CTRL-X.
H — Do not generate a carriage return after the input. The cursor remains just after the last character entered.
K — Clear the keyboard strobe.
N — Ignore null input.
S — Read the shift key modification. To use this you will need the shift key modification and some method of displaying lower case.

Note: The order in which you place these options is insignificant.

The next parameter, sexpr, is an optional input mask. If the character preceding the mask is a comma, sexpr will indicate which characters are not allowed. If it is a semicolon, sexpr will specify which characters are to be permitted.

The optional prompt works just like the prompt in Applesoft. If not specified, however, there will be no question-mark prompt.

Var may be either a string variable or a real variable. A TYPE MISMATCH ERROR will result if you use an integer variable. When you input a real variable, the input can be a formula, not necessarily just a straight number. Keep in mind that BASIC can produce a SYNTAX ERROR or any of the math errors when evaluating the formula. You can avoid such errors by disallowing null input and using the input mask to screen out all characters except numbers.

Let's look at some examples now.

A typical YES/NO question:

```
&INPUT (1,EN;"YN");"WOULD
YOU LIKE A PRINTOUT?  ";P$
```

To enter a number representing one of the Apple's peripheral slots:

```
&INPUT (1,NE;"1234567");
"WHAT SLOT IS YOUR
PRINTER IN?";SL
```

To enter a file name without a drive specification:

```
&INPUT (30,N,",");FL$
```

To enter up to three characters, forbidding "Q". (Note the double comma.)

```
&INPUT (3,,"Q");A$
```

To enter any three characters (except control characters):

```
&INPUT (3);T$
```

To evaluate an input expression:

```
&INPUT X
```

Technical Information

AE2 is 2048 bytes long, and can be placed anywhere in memory. If you do not use the loader program, you may BLOAD the file AE2.OBJ to whatever address you wish, and place a jump instruction at $3F5 to the start address.

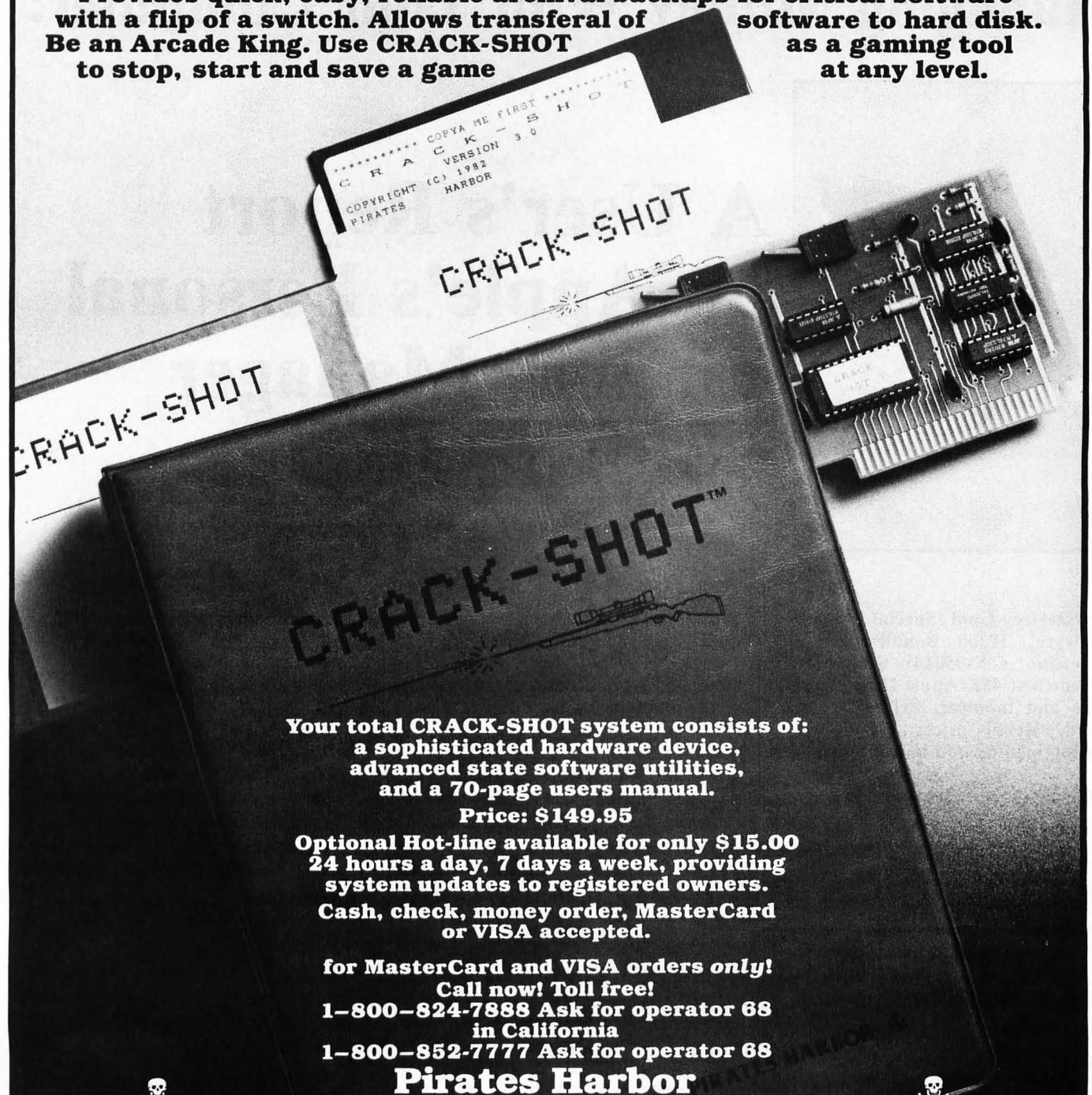
AE2 uses only four zero page locations: 3, 4, $CE and $CF. The locations $CE and $CF are temporaries. If you are not using &ON RESET GOTO, you may also consider locations 3 and 4 as temporaries. The length of the input buffer is stored in location $2FF by the three *AE2* input commands. The loader program loads into page 3 from $300 to $3CB. Once it is executed, page 3 is again free.

The loader program looks through memory for *AE2* by trying to find an area in memory with the proper checksum value. If it finds *AE2,* the loader checks if HIMEM is above the start of *AE2.* If it is, HIMEM is lowered to the start of *AE2.* Otherwise, HIMEM is unaffected. If the loader does not find *AE2,* HIMEM is lowered 2K, and *AE2* is loaded at HIMEM. Because of the checksum technique, if you wish to modify *AE2,* you must modify the loader program.

CRACK-SHOT™

Internationally acclaimed
The ultimate in copy de-protection

CRACK-SHOT is a total system comprised of hardware and software modules. CRACK-SHOT is easy to use, designed for users and programmers. Eliminates "owners paranoia" about disk failure. Provides quick, easy, reliable archival backups for critical software with a flip of a switch. Allows transferal of software to hard disk. Be an Arcade King. Use CRACK-SHOT as a gaming tool to stop, start and save a game at any level.



Your total CRACK-SHOT system consists of:
a sophisticated hardware device,
advanced state software utilities,
and a 70-page users manual.

Price: $149.95

Optional Hot-line available for only $15.00
24 hours a day, 7 days a week, providing system updates to registered owners.

Cash, check, money order, MasterCard or VISA accepted.

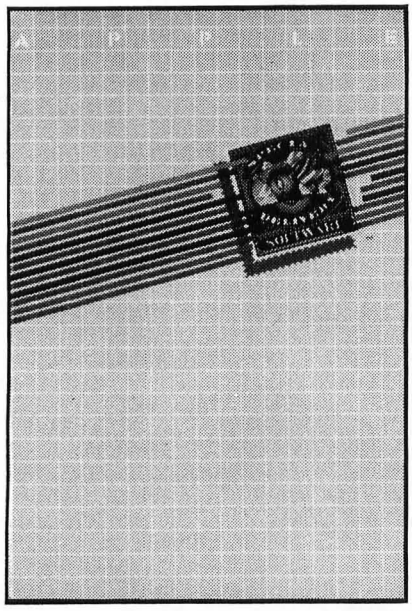
for MasterCard and VISA orders *only*!
Call now! Toll free!
1–800–824-7888 Ask for operator 68
in California
1–800–852-7777 Ask for operator 68

Pirates Harbor

P.O. Box 8928, Boston, MA 02114 617–720-3600 MODEM

Apple is a registered trademark of Apple Computer. CRACK-SHOT is the trademark of Pirates Harbor.
System requirements: 48K Apple II or Apple II Plus, 1 disk drive, Ramcard helpful but not required.
CRACK-SHOT is intended to be used as a programmers aid, gaming tool, and legal archival backup of your programs.
CRACK-SHOT should not be used for illegal purposes.

Managing Money With Your Computer:



A User's Report on Apple's Personal Finance Manager

Reviewed by Steve Birchall

by Jeffrey Gold (Special Delivery Software, 10260 Bandley Drive, Cupertino, CA 95014). System requirements: 48K Apple II with disk drive and monitor. Printer is optional. Retail price: $75.00, including manual and backup disk.

Balance your checkbook! Manage your family budget! Figure your taxes! Remember all those vague and wonderful promises you drooled over before you bought your computer? All of them were just rationalizations for buying that glittering technological toy, because what you really wanted it for was to play Quasar Gobblers. In actual practice, very few people use their computers for those mundane tasks. The promises are fulfillable, however. More important, the benefits of computer-managed bank accounts and spending tracking are substantial.

You can balance your checkbook with the cheapest calculator. You can write a program to do it on a microcomputer with only rudimentary knowledge of BASIC. More difficult problems are easy to resolve on a spreadsheet program. Why would anyone be foolish enough to buy a separate piece of software just to handle his banking?

The answer is to get the convenience and ease that will encourage you to begin and continue to use it. That's what computers are for: to make things easy and less tedious. With *PFM*, you can plan your financial affairs more wisely, and perhaps learn how to divert your money toward the things you really want, rather than wondering where it all went. The tax laws of this country are so complicated that

"With PFM, you can plan your financial affairs more wisely, and perhaps learn how to divert your money toward the things you really want, rather than wondering where it all went."

almost anyone could benefit from having a computer to generate the orderly listings of data needed for filling out the old 1040. Would you like to have a printout, sorted by date, with totals for each category, of every tax deductible purchase you made for the year, just by typing a couple of command characters? If the IRS is going to use computers to check you out, why not use your own computer as a convenient tool for getting all your tax breaks. Formerly this privilege belonged only to Big Business. Now you can have it, too.

"All sections have menus with clearly listed program choices, and the expected response obvious. If you make an inappropriate choice, PFM recycles to the beginning of the sequence, with no harm done."

Apple's Personal Finance Manager puts all these functions together (and maybe a couple you wouldn't have thought about until much later) into one convenient package. Putting data in and retrieving it are painless (though what *PFM* tells you about your finances may be painful). All sections have menus with clearly listed program choices, and the expected response obvious. If you make an inappropriate choice, *PFM* recycles to the beginning of the sequence, with no harm done. All the data for an entire year is stored on the program disk so you don't have to shuffle disks in and out as you use it. (A duplicate, backup disk is used for a safety copy.)

How *PFM* Works

Conceptualize the system as an elaborate checkbook, because you enter information the same way — current balance, check number, date, payee, and amount. *PFM* will show your new balance at the top of the screen. In the check number column, the next check number will appear as a default (as will the last date entered), making entry fast and easy. You indicate deposits by overriding the check number with a "D", cash purchases with a "C" (in which case they are not deducted from the checkbook balance), and credit card purchases with a two letter symbol (VS and MC are built in; up to ten others can be user-defined). The two extra columns at the right — Tax Deductible and Budget Category — are significant in terms of tracking various expenses.

To begin using *PFM*, make a list of the spending categories you want to track, assign a monthly amount (which can be the same or vary with the month), and give them a two letter abbreviation. Up to 24 are possible. Don't use them all initially, because you may wish to add or delete some, or perhaps break up or combine others as you begin to get a feeling for the kind of feedback *PFM* can provide. Some things are obvious — food, rent, utilities, clothes, and so on. But others depend on your own personal situation and your relationship to the IRS.

Make the categories meaningful and useful, not abstract. If you have a weakness for stopping at the bakery, make that a separate category, and periodically ask for a current total and percentage of total. The daily amounts may seem inconsequential, but the monthly total may really surprise you. The impact on your monthly budget may be more devastating than you believed. If your hobby is collecting country/western records, make a separate account for that too. Sometimes you will be over budget when a new batch is released, but at other times you may have extra dollars to spend. The power of accurate numbers is amazing. To know what you are spending your money for, and how much, helps not only to control your finances, but also to direct your spending toward the things you really want, because you can identify and eliminate wasteful spending.

In my own case, I had a series of unexpected, expensive car repairs (When are they ever expected and cheap?) which nearly destroyed my solvency. With *PFM*, I could get a precise idea of how much over the planned budget I was, see what items were under budget, and could target areas where I could cut spending. Without this help, I would have worried about the problem, cutting back a little here and there when the bills came due. I would have floundered, with no clear idea of what steps I had to take: how much to cut, where, and for how long.

When using *PFM,* a certain amount of discipline helps. It is important to build the necessary record keeping into your daily and weekly routines. I tend to use my computer every day. I always boot up the *PFM* disk first, and enter all the checks, cash purchases, and credit card charges while they are still fresh in my mind. The process takes only a minute or two, and it makes a good warmup exercise before starting serious work. I make a point of collecting register receipts and stuffing them into my wallet with my change. After putting the data into the computer, I toss any receipts for tax deductible expenses into an envelope for that month. This daily procedure is far preferable to trying to enter all the data once or twice a month, and the benefits are enormous.

Practical Benefits

Let's say you have several months' of data stored, and you want to analyze your spending patterns. Looking at the main menu, you select "Budget Category Summary" and another menu appears. You can choose a summary of all categories in a given month. The default is the current month, and a

year-to-date option is available. Another possibility is a summary of a particular budget for the entire year, or a full year budget review. Any of these will give you the totals, the budgeted amount, the percent of the budget spent, and the percent of that amount against your total spending. Pressing "G" gives you a bar graph representation.

You discover that your sweet tooth got the best of you — 452% over budget at the corner bakery. "Hmmm...rent is under budget. Maybe I forgot to pay the rent this month. Better do that right away." Suppose you have a burning desire to acquire a new piece of computer equipment (perhaps that new Holographic Printer Interface Card). It costs $150, you're broke and your credit cards are used to the max. *PFM* can help you acquire it. You will find that most of your expenses are fixed and beyond your control. Dig a little deeper, however, and you discover some areas of spending that could be trimmed. Last month you spent $250 on wild evenings at the Plastic City Disco. Restricting your disco visits to Saturday nights only for a month would save $100, a small sacrifice, since you might prefer to stay home a few more evenings to use your new Graphics Interface Card if you had it. So far, so good. Avoiding the bakery would shave off another $50. There you have it: a plan to change spending temporarily to satisfy a greater desire. My example may be silly, but it illustrates how you can use *PFM* to improve the quality of your life. The potential is in your computer; all you have to do is make it work for you.

As an extra benefit, *PFM* will help you reconcile your checkbook each month. It presents each outstanding check and deposit in order and asks if it is on the bank's statement. You simply press "Y" or "N" and *PFM* takes care of the rest. When finished, it presents a summary of deposits and withdrawals, comparing your checkbook's balance with the bank statement. If you have made an error, it indicates how much over or under you are, which usually points directly to the reason (a check or deposit not entered or posted incorrectly), enabling you to make the necessary correction. If the reconciliation is successful, it asks if you want to make it permanent and stores the result.

Editing is easy. When the entry is on the screen, push "E" to get into the edit mode, move the cursor to the desired line, push "E" again to edit that line and make the changes. If the entry is buried somewhere in the year's records, the Search and Sort routine will find it quickly, using a series of menu choices.

Search and Sort is also good for finding specific checks. Let's say that the bill for your Guzzler's Gas credit card arrives with a pointed note at the bottom saying you failed to pay the bill last month. Use the search routine to find out whether you wrote the check, or have it print a list of all the checks you wrote to Guzzler's Gas, with numbers, dates and amounts, sorted by date for the entire year. (This is also a good way of finding out if you have been spending too much on gas, because it gives you a total on those checks.) One of the search choices is Unreconciled Checks Only, so you can find out if the check has been cashed.

Evaluation

One of the pitfalls I found in using *PFM* was that I defined a budget category called Spending Money. Every time I wrote a check for cash, it would go into this budget. So far so good, because it put a little restraint on cashing checks at the bank too often. But I then took the cash and spent it on things belonging in other categories (food, books, movies, etc.), giving me a double entry for many items. This led to misleading totals, since outgo for a month would be inflated by these extra entries. I had to find a way to track these expenses, and all kinds of complicated schemes came to mind. Ultimately, I discovered that subtracting the category Spending Money from the total amount spent brought me fairly close to the correct amount. (A few dollars would not be accounted for since I don't keep track of every penny going in and out of my pocket for insignificant items. If I needed to know that amount, however, I could take the difference between all the checks written for cash, and all cash expenditures.)

Another way to deal with this is to disguise Spending Money as a Credit Card Account. This routine will not deduct payments to the account, because that money has already been accounted for when the individual charges were entered. The *PFM* manual fails to warn users against this rather obvious trap. A few words on how to avoid this problem would be very helpful.

If you maintain more than one bank account, *PFM* won't be as helpful as some other systems. Nor will it reconcile statements from credit cards. If you want to be clever, you can disguise a savings account by using a check number prefix. Most people will use only three of the four digits for check numbers. Your regular checking account could have the normal three digits, a savings account could use all numbers starting with 9000, and your IRA could use numbers in the 8000-8999 series. In this way, you could track ten separate bank

"Editing is easy...If the entry is buried somewhere in the year's records, the Search and Sort routine will find it quickly, using a series of menu choices."

accounts. Again, the manual hints at this possibility, but doesn't explain it fully.

Perhaps the most disappointing feature of this otherwise well-thought-out package is the lack of a check writing routine. Everything else is taken care of nicely, so why not include this convenience? You feel stupid sitting at your Apple writing checks by hand, then entering them into *PFM*, knowing that the computer is perfectly capable of doing a beautiful job of printing checks while simultaneously entering the data into the program. This is a relatively easy thing to add, and Apple should consider an update.

An especially nice feature is the ease of obtaining a printout. When you want a copy of what is on the screen, press Control-P. The procedure becomes a bit tedious at the end of the year when you have multiple screens of data to print out for each budget. A routine to print all the information gathered by a search procedure in one continuous pass might improve this. Nonetheless, having an orderly, detailed listing of all your tax deductible expenses at the end of the year is very satisfying.

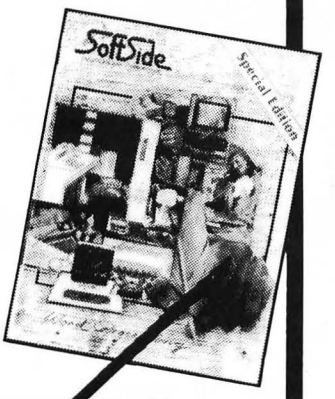
"Talking about money and juggling numbers in a checkbook can quickly bog down in murky prose, but the manual carefully avoids this."

The manual is easy to use, and all the major elements of *PFM* are explained in a straightforward style. Talking about money and juggling numbers in a checkbook can quickly bog down in murky prose, but the manual carefully avoids this. You can find out what you need to know about a forgotten section without wading through extraneous material. This virtue can become a fault when things are left unexplained, however, leaving the user to puzzle it out for himself.

Summary

Anyone who has a home computer should use a money management system such as the *PFM*. Although most people don't regard this as the primary reason for buying a computer, the tax records and budget data it can generate are valuable. The personal computer has many essentially free applications, so let it do some of your boring and tedious chores for you.

When does your *SoftSide* subscription expire?



49007STANTG97*B00F12S1OCT 82
1102691 017 M
GEOFFREY STANTON
97 BAYSIDE COURT
KALAMAZOO, MICH. 49007

The last five characters (three letters for month, two numbers for year) on the top line of your mailing label will tell you when your subscription ends.

For more information, write:

SoftSide™
100 Pine Street, Holmes, PA 19043

See page 69 for ordering information.

April 15th is just around the corner

The ACCOUNTANT

Finance Data Base System

Optional VisiCalc™ Interface Available

BE A MONEY MASTER

Define up to 63 tax codes and save when you prepare your returns. Available for APPLE™ and IBM™PC.

"Among bookkeeping programs, Decision Support Software's ACCOUNTANT ($129) earns high marks and is easy to use." — Money Magazine, Nov. 1982

(800) 368-2022

Decision Support Software

1438 Ironwood Drive, McLean, VA 22101 (703) 241-8316

Add $3.00 shipping and handling. Virginia residents add 4%

APPLE™, IBM™, and VisiCalc™ are trademarks of APPLE Computer, Inc., IBM Corp., and VisiCorp, Inc. respectively.

BLACKJACK

by Edmund R. Malinowski

***Blackjack* is a card game for one to five players, and requires an Apple® with 48K RAM and Applesoft.**

Rules

Blackjack is based on the rules of the Atlantic City casinos. In this version, as many as five players can challenge the dealer.

The dealer shuffles four decks of cards together and deals each player two cards, face up. The dealer's first card is dealt face up but his second card, called the "hole" card, is dealt face down. The dealer exposes his "hole" card only after all players finish drawing their cards.

To win the hand, your cards must total as close as possible to 21 without exceeding 21. Face cards count as 10 and an ace counts as 1 or 11, whichever is the best move. All other cards count according to their face value.

You may "stand" (draw no more cards) or "hit" (draw a card) until satisfied, or until your total exceeds 21. This is a "bust." It automatically loses, even if the dealer also draws a "bust." You win if your total is closer to 21 than the dealer's total. If the totals are the same, neither wins.

If a player's first two cards total 21 (called Blackjack) the dealer pays 1.5 times the bet, unless the Dealer also has Blackjack, in which case neither wins.

Betting

Insurance — If the dealer's face-up card is an ace, you can take "insurance" by placing a side bet equal to one-half the original bet. You win your insurance bet at odds of 2 to 1 if the dealer's "hole" card yields Blackjack; otherwise, you lose your insurance bet.

Surrender — After receiving your first two cards, you may opt to "surrender" your hand, automatically losing 1/2 of your bet and ending your play.

Doubling down — After receiving your first two cards, you may opt to

"double down"; that is, double your bet and draw only one additional card.

Split — If the first two cards are identical, you can "split" the pair into two hands. The bet on each hand is the same as the original bet. Each hand receives another face-up card. If the split pair consists of aces, no further play is allowed. Play on the first split pair must be completed before play on the second pair begins. Further splitting of split pairs is not permitted.

The dealer must "hit" on 16 or less and "stand" on 17 or more.

Program Notes

This program uses Larry Williams's high-resolution playing card graphics (*Solitaire, SoftSide,* May 1982). Like *Solitaire,* this program consists of two parts. The first part generates shape tables for the cards, and sets BASIC's start-of-program text pointer at 16385 so that the main program will reside above the first hi-res screen. The POKE 16384,0 places a zero in memory where Applesoft requires it, just before the program text.

Those with disk drives can speed up the loader program with the following procedure. After you enter the loader program, and test it with *SWAT*, type line 55 as shown below:

```
55 STOP
```

Run the program, then type, "BSAVE BLACKJACK.SHAPES, A7569, L623". Now type "NEW", and enter the following short program.

```
10 HOME:D$ = CHR$(4)
20 PRINT D$;"BLOAD
SOLITAIRE.SHAPES"
30 POKE 232,145:POKE 233,29
40 POKE 103,1:POKE 104,64:
POKE 16384,0
50 PRINT D$;"RUN
SOLITAIRE.PGM"
```

Use Paddle (1) and its pushbutton to place bets and make selections. By turning the control paddle after the "PLACE BETS" prompt, you can scan from a $2 minimum to a $50 maximum. (To avoid handling small change, the game allows only even dollar bets.) Pressing and releasing the game button enters the bet. Lines 1110 to 1180 do this. B(I) scans from 2 to 50 as the game paddle rotates (see lines 1120 to 1150). If the paddle button has not been pressed (see line 1150) the program goes to line 1120 and reads the paddle position again. When you depress the paddle button, line 1170 takes control until you release the button. This internal recycling of line 1170 is necessary, otherwise the program could complete the FOR-NEXT loop before the button is released and enter the same bet for all players.

Similar paddle-pushbutton routines scan and enter players' options (stand, hit, surrender 1/2, double down or split — see lines 860 to 950) and place insurance bets (see lines 1340 to 1390.)

The program also generates sound effects for wins, losses, busts, blackjacks and ties. The sound generator subroutines appear in lines 2990 to 3190. Lines 3050 to 3080 POKE a Machine Language subroutine into locations 770 through 725 which generates the sounds by toggling the speaker. Lines 3090 to 3185 create five different sounds, each having a different number of notes, NN(K). Each note in a sound has a unique frequency FQ(I,K) and duration DU(I,K). You will find these frequency-duration pairs for each note of the five sounds (SD = 1 for lose; SD = 2 for win; SD = 3 for blackjack; SD = 4 for tie and SD = 5 for dealer's bust) in lines 3150 through 3185. High-pitched sounds indicate wins, and low-pitched sounds indicate losses.

Variables

B(I): Bet made by player I.
BL: Integer 4, used to set HCOLOR = 4 (black).
BL39$: String of 39 spaces.
BU: Integer 6, used to set HCOLOR = 6 (blue).
BX(I): New bet due to options selected by player I.
D(M): Code number (same as NUM) for the M-th card in the deck.
DN: Dealer's seat number (DN = N + 1).
DU(9,5): Duration of note used in sound subroutine.
FQ(9,5): Frequency of note used in sound subroutine.
I: Player's seat number.
M: Position of a card in the stack of four shuffled decks (top card is 0, bottom card is 207).
N: Number of players.
NC(I): Number of cards held by player I.
NN(SD): Number of notes in sound SD.
NUM: Three-digit code number for a card (NUM = 100*SU + VA).
OPT: Players' options (1 = stand, 2 = hit, 3 = surrender, 4 = double down, 5 = split).
RD: Integer 5, used to set HCOLOR = 5 (orange).
S(I): Sum of card values held by player I.
SD: Designates a particular sound (SD = 1 for lose; SD = 2 for win; SD = 3 for blackjack; SD = 4 for tie; SD = 5 for dealer's bust).
SPLIT: Designates first (SPLIT = 1) hand or second (SPLIT = 2) hand of a split hand.
SU: Card suit (1 = spade, 2 = heart, 3 = club, 4 = diamond).
T(I): Total winnings (or losses) of player I.
V(I,J): Value of the J-th card held by player I.
VA: Card value (1-13).
W(I): Amount won (or lost) by player I upon completion of the hand.
WH: Integer 7, used to set HCOLOR = 7 (white).
X,Y: Horizontal and vertical coordinate positions used to draw cards on the HGR screen.
All other variables are either loop variables or counters.

```
SS SS SS SS SS SS SS SS SS SS SS
SS                             SS
SS      APPLESOFT BASIC        SS
SS 'BLACKJACK LOADER PROGRAM' SS
SS AUTHOR: EDMUND MALINOWSKI   SS
SS     COPYRIGHT (C) 1982      SS
SS SOFTSIDE PUBLICATIONS, INC SS
SS                             SS
SS SS SS SS SS SS SS SS SS SS SS
```

If you don't wish to type this program, it is also included in this month's SoftSide CV and DV.

```
10  NOTRACE : TEXT : HOME : SPEED=
    255
20 D$ =  CHR$ (4)
30  POKE 232,145: POKE 233,29
40  VTAB 7: HTAB 11: PRINT "READI
    NG SHAPE DATA"
50  FOR A = 7569 TO 8191: READ I:
     POKE A,I: NEXT
60  HOME : TEXT : VTAB 10: PRINT
    "  1. LOAD FROM TAPE": PRINT
    : PRINT "  2. LOAD FROM DISK
    "
70  VTAB 20: INPUT "YOUR CHOICE:
    ";A$
80 A =  VAL (A$): IF A < 1 OR A >
    2 THEN 70
90  ON A GOTO 100,150
100  HOME : VTAB 10: PRINT "CUE T
    HE TAPE TO THE BEGINNING OF
    THE    BLACKJACK PROGRAM. TH
    EN START THE TAPE, AND PRESS
     <RETURN>.": PRINT
110  PRINT "AFTER TWO BEEPS, THE
    PROMPT SYMBOL WILL RETURN. T
    URN OFF THE RECORDER, AND TY
    PE 'RUN' TO BEGIN THE GAME."
120  INPUT "";A$
130  POKE 103,1: POKE 104,64: POKE
    16384,0
140  LOAD
150  HOME : VTAB 10: PRINT "NOW R
    UNNING THE BLACKJACK PROGRAM
    "
160  POKE 103,1: POKE 104,64: POKE
    16384,0
170  PRINT D$;"RUN BLACKJACK.PGM"
180  DATA 18,0,38,0,50,0,65,0,77,
     0,89,0,101,0,114,0,123,0,136
     ,0,148,0,164,0,173,0,186,0,1
     99,0,50,1,148,1,241,1,95,2,3
     3,36,100,12,14,14,54,63,119,
     9,46,0,41,45,45,216,219,16,1
     2,12,45,32,28,63,30,7,0,1,16
     8,45,5,32,28,103,12,60
190  DATA 63,63,0,73,33,5,56,63,3
     9,12,12,12,54,46,0,1,112,45,
     5,32,228,63,39,44,45,45,0,9,
     45,5,32,28,63,214,36,100,12,
     45,5,0,33,100,12,12,228,58,6
     3,7,0,9,45,12,228,63,214,36,
     32,12,45,14,54,0,41,101,12,6
     0,63,7,32,12,45,21,46
200  DATA 0,33,36,36,108,9,45,14,
     54,54,30,63,7,32,36,36,0,1,1
     12,45,5,32,36,36,4,0,9,109,2
     8,223,108,13,36,228,95,191,5
     4,7,0,33,36,36,108,9,30,30,3
     0,14,14,14,5,0,73,9,45,45,45
     ,229,59,63,12,109,73,56,255,
     59,223,63,7,40,45,109
210  DATA 109,45,45,5,56,63,63,63
     ,63,63,63,63,7,40,45,45,45,4
     5,45,45,45,45,60,63,63,63,63
     ,63,63,63,63,44,45,45,45,45,
     45,45,45,45,28,63,63,63,63,6
     3,63,63,39,45,45,45,45,45,45
     ,45,229,63,63,63,63,63,63,10
     3,45,45,45,45,45,229
220  DATA 63,63,63,63,103,45,45,4
     5,229,63,63,103,45,229,39,45
     ,0,73,9,45,45,45,229,59,63,1
     2,109,73,56,255,59,223,63,7,
     40,45,109,109,45,45,5,56,63,
     63,63,63,63,63,63,7,40,45,45
     ,45,45,45,45,45,45,60,63,63,
     63,63,63,63,63,63,63,44
230  DATA 45,45,45,45,45,45,45,45
     ,28,63,63,63,63,63,63,63,103
     ,41,109,45,109,45,220,27,63,
     255,8,45,45,45,56,63,63,63,4
     4,45,45,45,28,63,63,103,45,2
     29,63,0,73,73,9,37,255,40,45
     ,45,60,63,63,31,40,45,45,45,
     45,60,63,63,63,63,31
240  DATA 200,45,45,45,45,45,45,3
     7,63,63,63,63,63,63,255,40,4
     5,45,45,45,45,45,45,45,60,63
     ,63,63,63,63,63,63,63,76,45,
     45,45,45,45,45,37,63,63,63,6
     3,63,63,103,41,45,45,45,45,6
     0,63,63,63,63,76,45,45,45,24
     ,63,63,103,41,60,7,0
250  DATA 73,73,9,37,255,40,45,45
     ,60,63,63,31,40,45,45,45,45,
     60,63,63,63,63,31,40,45,45,4
     5,45,45,45,60,63,63,63,63,63
     ,63,31,40,45,45,45,45,45,45,
     45,45,60,63,63,63,63,63,63,6
     3,63,44,45,45,45,45,45,45,45
     ,45,60,63,63,63,63,63
260  DATA 63,63,63,76,45,45,109,4
     1,45,45,60,63,63,223,63,63,3
     9,45,45,109,41,45,45,220,255
     ,219,27,103,43,29,77,73,201,
     37,255,219,27,63,0,45,45,45,
     45,28,63,63,63,12,45,45,28,6
     3,12,5,0
```

continued on page 108

APPLE™ SWAT TABLE FOR:

BLACKJACK LOADER PROGRAM

(Modified Parameters: NU = 3 B = 500)

LINES	SWAT CODE	LENGTH
10 - 30	CD	48
40 - 60	TL	120
70 - 90	VU	67
100 - 120	VX	238
130 - 150	UD	81
160 - 180	ZF	275
190 - 210	SK	653
220 - 240	ZI	651
250 - 260	UN	395

"My RAMEX 128™ memory expansion card enhances my Apple™ and is a superb product for my Apple Computer.

I wouldn't dream of turning on my Apple without a RAMEX 128 installed.

The software supplied makes the card useful even for me because I can't modify DOS on my own. And it is so easy to install, I do it myself.

I look forward to upcoming products from Omega MicroWare, Inc.™ I get all of them because my dad owns the company."

Joel Alpert
Eighth Grade Student
Deerpath Intermediate School

If Joel thinks this highly of Omega products, we must be doing something right.

See your nearest dealer for the ONE card that LOADs and SAVEs a complete 136k VisiCalc™ file in 20 seconds yet sells for as much as $250 LESS than the others. Of course, you will pay a little more than Joel, but we think you will find the value worthwhile.

222 So. Riverside Plaza
Chicago, IL 60606
(312) 648-4844

© 1982 Omega MicroWare, Inc.

Apple is a registered trademark of Apple Computer Co.
VisiCalc is a registered trademark of VisiCorp.
RAMEX 128 is a trademark of Omega MicroWare, Inc.

S.N.A.P.™

Super Name, Address, & Phone program. Create, maintain, and print your personal phone directory, customized mailing list labels, business propsect lists, etc. Sort by your own special categories. $19.95

HANDS ON™

Tasteful erotic fantasy adventures for adults. Experience the joy of erotic exploration and tactile sensations. Get in touch with the incredible variations in human sensuality. Superb color graphics.
Adults only $29.95

Part of program scene from HANDS ON!

PLEASURE™

An electronic game delight for adult couples. Graphic sensual adventures designed to ignite your imagination and expand your romantic repertoire.
Adults only $29.95

• Mail orders: Check, Money order, Credit card

VILLAGE SOFTWARE

Dept. SS-36
31220 La Baya Drive, Suite 110
Westlake Village, California 91362

• Available at leading computer stores.

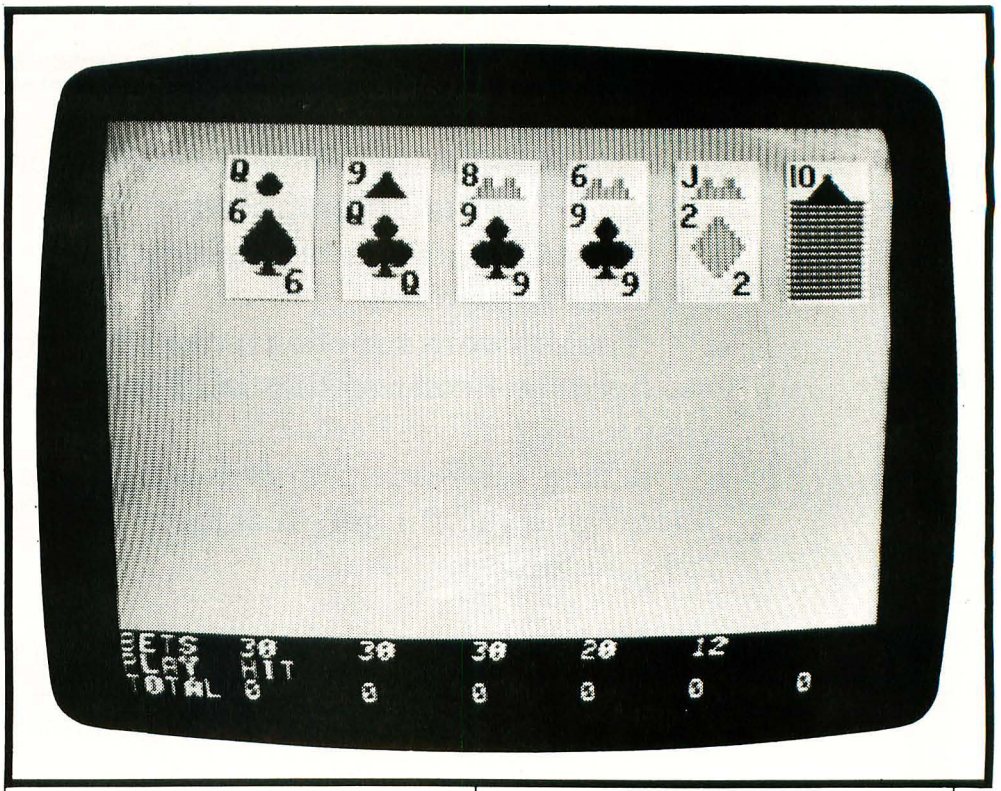
Add $1.50 shipping and handling per order. ($5 overseas.)

California residents add 6½% sales tax.

For credit card orders, include account number, expiration date and signature.

Requirements: 48K Apple II or II+, Single Disk Drive, Applesoft in ROM (Firmware).

Apple II, II+, and Applesoft are trademarks of Apple Computer, Inc. Program titles are trademarks of Village Software Corp.



Blackjack *continued*

```
SS SS SS SS SS SS SS SS SS SS SS
SS                             SS
SS      APPLESOFT BASIC        SS
SS        'BLACKJACK'          SS
SS AUTHOR: EDMUND MALINOWSKI   SS
SS     COPYRIGHT (C) 1982      SS
SS SOFTSIDE PUBLICATIONS, INC  SS
SS                             SS
SS SS SS SS SS SS SS SS SS SS SS
```

If you don't wish to type this program, it is also included on this month's SoftSide CV and DV.

Initialization.

```
100  CALL  - 936: VTAB 2: HTAB 14
     : PRINT "BLACKJACK"
105  GOSUB 3050
110  PRINT : PRINT  TAB( 18)"BY"
120  PRINT : PRINT  TAB( 9)"EDMUN
     D R. MALINOWSKI"
130  PRINT : PRINT : PRINT  TAB(
     12)"PLAYERS' OPTIONS"
140  PRINT : PRINT  TAB( 14)"STAN
     D"
150  PRINT  TAB( 14)"HIT"
160  PRINT  TAB( 14)"SURRENDER 1/
     2"
170  PRINT  TAB( 14)"DOUBLE DOWN"

180  PRINT  TAB( 14)"SPLIT PAIR"
190  PRINT  TAB( 14)"INSURANCE"
200  PRINT : PRINT "PLAY EVEN DOL
     LAR BETS FROM $2 TO $50"
210  PRINT : PRINT "(USE PDL(1) A
     ND PUSHBUTTON(1) TO PLACE"
220  PRINT "BETS AND MAKE SELECTI
     ONS)."
230  VTAB 24: HTAB 6: PRINT "NUMB
     ER OF PLAYERS (1-5) = ";: INPUT
     "";N$
240 N =  VAL (N$):DN = N + 1
250  IF N < 1 OR N > 5 THEN 230
260  DIM V(12,12),D(207),S(12),NC
     (12),B(12),BX(12)
270 BL39$ = "
                                   "
280 BL = 4:WH = 7:RD = 5:BU = 6
290  HOME : VTAB 10: HTAB 9: PRINT
     "SHUFFLING FOUR DECKS": GOSUB
     2500
300  HGR : SCALE= 1: ROT= 0: POKE
     34,20: GOSUB 2350: GOTO 1030
```

Pick a card.

```
320  IF M > 206 THEN  GOSUB 2560
330 M = M + 1:NUM = D(M)
340 SU =  INT (NUM / 100):VA = NU
     M - 100 * SU
350  RETURN
```

Add values of cards.

```
370 S = 0:S1 = 0:S2 = 0:ACE = 0
380  FOR J = 1 TO NC(I):VA = V(I,
     J): GOSUB 420: NEXT
390 S(I) = S
400  RETURN
```

Assign values to J, Q, K, and A.

```
420  IF VA > 10 THEN VA = 10
430  IF VA = 1 THEN ACE = 1
```

APPLE™

```
440 S1 = S1 + VA
450  IF ACE = 1 THEN S2 = S1 + 10

460 S = S1: IF S2 > 0 AND S2 < 22
     THEN S = S2
470  RETURN
```

Print cards.

```
490  IF SPLIT = 1 THEN Y = 20 + 1
    0 * JJ: GOTO 520
500  IF SPLIT = 2 THEN Y = 100 +
    10 * JJ: GOTO 520
510 Y = 12 * (J + JJ)
520 X = 40 * ID
530  GOSUB 2300
540  RETURN
```

Routine to handle split cards.

```
560  GOSUB 860
570  ON OPT GOTO 830,640,1660
590  GOSUB 320:BX(I) = BX(I) * 2
600  GOSUB 490: GOSUB 720
610  IF S < 22 THEN 830
620  RETURN
```

Routine to handle hits.

```
640  GOSUB 320
650  IF SPLIT = 1 THEN Y = 20 + 1
    0 * JJ: GOTO 680
660  IF SPLIT = 2 THEN :Y = 100 +
    10 * JJ: GOTO 680
670 Y = 12 * (J + JJ)
680 X = 40 * ID: IF SPLIT <  > 0 THEN
    JJ = JJ + 1
690  GOSUB 2300: GOSUB 720
700  IF S > 21 THEN  RETURN
710  GOTO 560
720 NC(I) = NC(I) + 1
730 V(I,NC(I)) = VA
740  GOSUB 370
750  IF S < 22 THEN 780
760  VTAB 22: HTAB 6 * ID: PRINT
    "BUST "
762  IF I = DN THEN SD = 5: GOSUB
    3000: GOTO 770
765 SD = 1: GOSUB 3000
770  GOSUB 790
780  RETURN
790  IF NC(I) <  > 0 THEN 810
800  RETURN
810 NC(I) = NC(I) - 1
820  GOTO 790
830  VTAB 22: HTAB 6 * ID: PRINT
    " ";S(I);"  "
840  RETURN
```

Select stand, hit, surrender, double down, or split.

```
860  VTAB 22: HTAB 6 * ID
870 OPT =  INT ( PDL (1) / 52) +
    1
880  IF OPT = 1 THEN  PRINT "STAN
    D"
890  IF OPT = 2 THEN  PRINT " HIT
     "
900  IF OPT = 3 THEN  PRINT "SURR
     "
910  IF OPT = 4 THEN  PRINT "DBLD
     "
920  IF OPT = 5 THEN  PRINT "SPLI
    T"
940  IF  PEEK ( - 16286) <  = 127
     THEN 860
950  IF  PEEK ( - 16286) > 127 THEN
    950
960  VTAB 24: HTAB 1: PRINT BL39$
    ;
970  IF OPT = 3 AND NC(I) > 2 THEN
     VTAB 24: HTAB 3: PRINT "CAN
    NOT SURRENDER AFTER TAKING A
     HIT"; CHR$ (7);: GOTO 860
980  IF OPT = 4 AND NC(I) > 2 THEN
     VTAB 24: HTAB 2: PRINT "CAN
    NOT DOUBLE DOWN AFTER TAKING
     A HIT"; CHR$ (7);: GOTO 860

990  IF OPT = 5 AND NC(I) > 2 THEN
     VTAB 24: HTAB 5: PRINT "ONL
    Y A SINGLE PAIR CAN BE SPLIT
    "; CHR$ (7);: GOTO 860
1000  IF OPT = 5 AND SPLIT > 0 THEN
     VTAB 24: HTAB 4: PRINT "FUR
    THER SPLITTING IS NOT ALLOWE
    D"; CHR$ (7);: GOTO 860
1010  RETURN
```

Place bets.

```
1030  FOR I = 1 TO DN:T(I) = 0: NEXT
1040  IF 2 * DN + M > 207 THEN  GOSUB
    2560
1050  FOR I = 1 TO N:W(I) = 0: NEXT
1060  FOR I = 1 TO 12:B(I) = 0:BX
    (I) = 0:S(I) = 0:NC(I) = 0: NEXT
1070 I = I + DN
1080  VTAB 21: HTAB 1: PRINT "BET
    S"
```

```
1090  VTAB 22: PRINT "PLAY"
1100  VTAB 23: PRINT "TOTAL";: FOR
      I = 1 TO DN: HTAB 6 * I + 1:
      PRINT T(I);: NEXT
1110  FOR I = 1 TO N
1120  VTAB 21: HTAB 6 * I + 1
1130 B(I) = 2 * ( INT ( PDL (1) /
      10.625) + 1)
1140  PRINT B(I);" "
1150  IF  PEEK ( - 16286) <  = 12
      7 THEN 1120
1160 BX(I) = B(I)
1170  IF  PEEK ( - 16286) > 127 THEN
      1170
1180  NEXT I
1190  HCOLOR= BU: HPLOT 0,0: CALL
      62454
```

Deal first two cards.

```
1210  FOR J = 1 TO 2: FOR I = 1 TO
      DN:ID = I
1220  GOSUB 320:V(I,J) = VA:DD(I,
      J) = D(M)
1230  IF J = 1 OR I <  = N THEN  GOSUB
      490
1240  NEXT I: NEXT J
1250  HCOLOR= BL:X = 40 * DN + 1:
      Y = 24
1260  FOR JY = 0 TO 28 STEP 2: HPLOT
      X,Y + JY TO X + 28,Y + JY: NEXT

1270  HCOLOR= RD: FOR JY = 1 TO 2
      9 STEP 2: HPLOT X,Y + JY TO
      X + 28,Y + JY: NEXT
1280  FOR I = 1 TO DN:NC(I) = 2: NEXT

1290  IF V(DN,1) > 1 THEN 1440
```

Insurance bets.

```
1310  VTAB 23: HTAB 1: PRINT BL39
      $;
1320  VTAB 24: HTAB 1: PRINT "
      BUY INSURANCE (1/2 ORIGINAL
      BET)?";
1330  VTAB 23: HTAB 1: PRINT "INS
      .";
1340  FOR I = 1 TO N
1350  VTAB 23: HTAB 6 * I + 1
1360  IF  PDL (1) > 127 THEN  PRINT
      "NO ";:B(I) = 0: GOTO 1380
1370  PRINT "YES";:B(I) = BX(I) /
      2
1380  IF  PEEK ( - 16286) <  = 12
      7 THEN 1350
1390  IF  PEEK ( - 16286) > 127 THEN
      1390
1400  VTAB 23: HTAB 6 * I + 1: PRINT
      B(I);"  ";
1410 W(I) = B(I) * (3 * ((V(DN,2)
      >  = 10)) - 1)
1420  NEXT I
1430  IF V(DN,1) = 1 AND V(DN,2) >
      9 THEN 1460
1440  IF V(DN,2) = 1 AND V(DN,1) >
      9 THEN 1460
1450  GOTO 1510
1460 X = 40 * DN:Y = 24: GOSUB 23
      00
1470  VTAB 24: HTAB 1: PRINT BL39
      $;
1480  VTAB 24: HTAB 12: FLASH : PRINT
      "DEALER BLACKJACK";: NORMAL

1485 SD = 3: GOSUB 3000: GOSUB 30
      00
1490  FOR I = 1 TO DN: GOSUB 370:
      NEXT
1500  GOTO 2090
```

Input and execute options for all human players.

```
1510  FOR I = 1 TO N:ID = I
1520 J = J + 1:JJ = 0:SPLIT = 0
1530  GOSUB 860
1540  ON OPT GOTO 1560,1700,1660,
      1640,1720
1560  GOSUB 370
1570  IF S(I) <  > 21 THEN 1620
1580  VTAB 22: HTAB 6 * I: FLASH
      : PRINT "BLKJK";: NORMAL
1585 SD = 3: GOSUB 3000
1590 W(I) = W(I) + 1.5 * BX(I)
1600 BX(I) = 0
1610  GOSUB 790: GOTO 1950
1620  GOSUB 830: GOTO 1950
1640  GOSUB 370: GOSUB 590: GOTO
      1950
1650  REM     SURRENDER ROUTINE
1660 BX(I) = 0.5 * BX(I)
1670 S(I) = 0: IF SPLIT > 0 THEN
      RETURN
1680  GOTO 1950
1700  GOSUB 370: GOSUB 640: GOTO
      1950
1720  IF V(I,1) = V(I,2) THEN 176
      0
1730  VTAB 24: HTAB 1: PRINT BL39
      $;
1740  VTAB 24: HTAB 8: PRINT "SPL
      ITTING NOT ALLOWED"; CHR$ (7
      );
1750  GOTO 1530
1760 X = 40 * I:Y = 0: GOSUB 2470

1770 X = 40 * I:Y = 0: GOSUB 2470

1780 NUM = DD(I,1): GOSUB 340: GOSUB
      2300
1790 Y = 80:NUM = DD(I,2): GOSUB
      340: GOSUB 2300
1800 IN = I + DN:NC(IN) = 2:V(IN,
      1) = V(I,2):BX(IN) = BX(I)
1810 Y = 10: GOSUB 320: GOSUB 230
      0
1820 V(I,2) = VA: GOSUB 370
1830 Y = 90: GOSUB 320: GOSUB 230
      0
1840 I = IN:V(I,2) = VA: GOSUB 37
      0
1850  IF V(I,1) = 1 AND V(IN,1) =
      1 THEN 1930
1860 I = IN - DN
1870  VTAB 24: HTAB 1: PRINT BL39
      $;
1880  VTAB 24: HTAB 15: PRINT "HA
      ND #1";
1890 SPLIT = 1:JJ = 0: GOSUB 560
1900 I = I + DN
1910  VTAB 24: HTAB 15: INVERSE :
      PRINT "HAND #2";: NORMAL
1920 SPLIT = 2:JJ = 0: GOSUB 560
1930 I = I - DN
1940 SPLIT = 0:JJ = 0
1950  NEXT I
1960  FOR I = 1 TO N
1970  IF NC(I) > 0 OR NC(I + DN) >
      0 THEN 2010
1980  NEXT I
1990  GOTO 2090
```

Play dealer's hand.

```
2010 I = DN:ID = DN:J = 3
2020 NUM = DD(DN,2): GOSUB 340: GOSUB
      370
2030 X = 40 * DN:Y = 24: GOSUB 23
      00
2040  IF S(DN) > 16 THEN 2070
2050  GOSUB 320: GOSUB 490: GOSUB
      720
2060 S(DN) = S
2070  IF S(DN) < 22 AND S(DN) < 1
      7 THEN 2050
2080  IF S(DN) < 22 THEN  VTAB 22
      : HTAB 6 * ID + 1: PRINT S(D
      N);
2090  IF S(DN) = 0 OR S(DN) > 21 THEN
      S(DN) = 1
```

Show wins, losses, and totals.

```
2110  VTAB 23: HTAB 1: PRINT BL39
     $;: VTAB 23: HTAB 1: PRINT "
     TOTAL";
2120  VTAB 21: HTAB 1: PRINT BL39
     $;: VTAB 21: HTAB 1: PRINT "
     WIN";
2130  FOR I = 1 TO N
2140  IF S(I) > 21 THEN S(I) = 0
2150  IF S(I + DN) > 21 THEN S(I +
     DN) = 0
2160 W(I) = W(I) + BX(I) *  SGN (
     S(I) - S(DN)) + BX(I + DN) *
      SGN (S(I + DN) - S(DN))
2170 T(I) = T(I) + W(I)
2180  VTAB 21: HTAB 6 * I + 1: IF
     W(I) > 0 THEN  FLASH
2190  PRINT W(I);: NORMAL
2192  IF W(I) > 0 THEN SD = 2
2194  IF W(I) < 0 THEN SD = 1
2195  IF W(I) = 0 THEN SD = 4
2196  GOSUB 3000
2200  VTAB 23: HTAB 6 * I + 1: PRINT
     T(I);
2210  GOSUB 790
2220 T(DN) = T(DN) - W(I)
2230 I = I + DN: GOSUB 790:I = I -
     DN
2240  NEXT I
2250  VTAB 23: HTAB 6 * DN + 1: PRINT
     T(DN);
2260  VTAB 24: HTAB 1: PRINT "
       PRESS GAME BUTTON FOR NEW
     DEAL";
2270  IF  PEEK ( - 16286) <  = 12
     7 THEN 2270
2280  GOSUB 790: GOSUB 2350: GOTO
     1040
```

Hi-res card graphics.

```
2300  HCOLOR= WH: FOR JY = 0 TO 2
     9: HPLOT X,Y + JY TO X + 29,
     Y + JY: NEXT
2310  HCOLOR= BL: DRAW VA AT X +
     1,Y + 7: DRAW VA AT X + 20,Y
      + 27: DRAW SU + 13 AT X + 6
     ,Y + 22
2320  HCOLOR= BL: DRAW VA AT X +
     2,Y + 7: DRAW VA AT X + 21,Y
      + 27: DRAW SU + 13 AT X + 7
     ,Y + 22
2330  IF SU > 2 THEN  HCOLOR= RD:
      DRAW SU + 13 AT X + 6,Y + 2
     2
2340  RETURN
2350  HOME : HCOLOR= BU: HPLOT 0,
     0: CALL 62454
2360  HCOLOR= BL
2370  HPLOT 90,84 TO 90,76 TO 96,
     76 TO 96,80 TO 90,80
2380  HPLOT 100,76 TO 100,84 TO 1
     06,84
2390  HPLOT 110,84 TO 110,76 TO 1
     16,76 TO 116,84: HPLOT 110,8
     0 TO 116,80
2400  HPLOT 126,76 TO 120,76 TO 1
     20,84 TO 126,84
2410  HPLOT 136,76 TO 130,76 TO 1
     30,84 TO 136,84: HPLOT 130,8
     0 TO 134,80
2420  HPLOT 150,80 TO 156,80 TO 1
     56,84 TO 150,84 TO 150,76 TO
     156,76 TO 156,80
2430  HPLOT 166,76 TO 160,76 TO 1
     60,84 TO 166,84: HPLOT 160,8
     0 TO 164,80
2440  HPLOT 172,76 TO 172,84: HPLOT
     170,76 TO 176,76
2450  HPLOT 186,76 TO 180,76 TO 1
     80,80 TO 186,80 TO 186,84 TO
     180,84
2460  RETURN
2470  HCOLOR= BU: FOR JY = 0 TO 5
     4: HPLOT X,Y + JY TO X + 29,
     Y + JY: NEXT
2480  RETURN
```

Shuffle four decks of cards.

```
2500 M = 0
2510  FOR K = 1 TO 4: FOR L = 1 TO
     13:D(M) = 100 * K + L:M = M +
     1: NEXT : NEXT
2520  IF M < 208 THEN 2510
2530  FOR K = 207 TO 0 STEP  - 1:
     M =  INT ( RND (1) * 208):T =
     D(M):D(M) = D(K):D(K) = T: NEXT
2540 M =  - 1
2550  RETURN
2560  VTAB 24: HTAB 1: PRINT BL39
     $;
2570  VTAB 24: HTAB 15: FLASH :: PRINT
     "NEW DECK";: GOSUB 2500
2580  NORMAL : VTAB 24: HTAB 15: PRINT
     "        ";
2590  RETURN
```

Sound routines.

```
3000  FOR J = 1 TO NN(SD)
3020  POKE 768,DU(J,SD): POKE 769
     ,FQ(J,SD): CALL 770
3030  NEXT J
3035  FOR PAUSE = 0 TO 200: NEXT
     PAUSE
3040  RETURN
3050  FOR I = 0 TO 25: READ J: POKE
     770 + I,J: NEXT I
3060  DATA 172,1,3,174,1,3,169,4,
     32,168
3070  DATA 252,173,48,192,232,208
     ,253,136,208,239
3080  DATA 206,0,3,208,231,96
3090  DIM FQ(9,5),DU(9,5)
3110 NN(1) = 2:NN(2) = 3:NN(3) =
     8:NN(4) = 1:NN(5) = 9
3120  FOR K = 1 TO 5: FOR I = 1 TO
     NN(K)
3130  READ FQ(I,K),DU(I,K)
3140  NEXT I: NEXT K
3150  DATA 144,2,80,2
3160  DATA 200,2,219,2,229,4
3170  DATA 211,1,211,1,211,1,229,
     2,211,1,229,2,211,1,229,2
3180  DATA 172,4
3185  DATA 172,1,164,1,156,1,150,
     1,144,1,135,1,126,1,117,1,10
     6,2
3190  RETURN
```

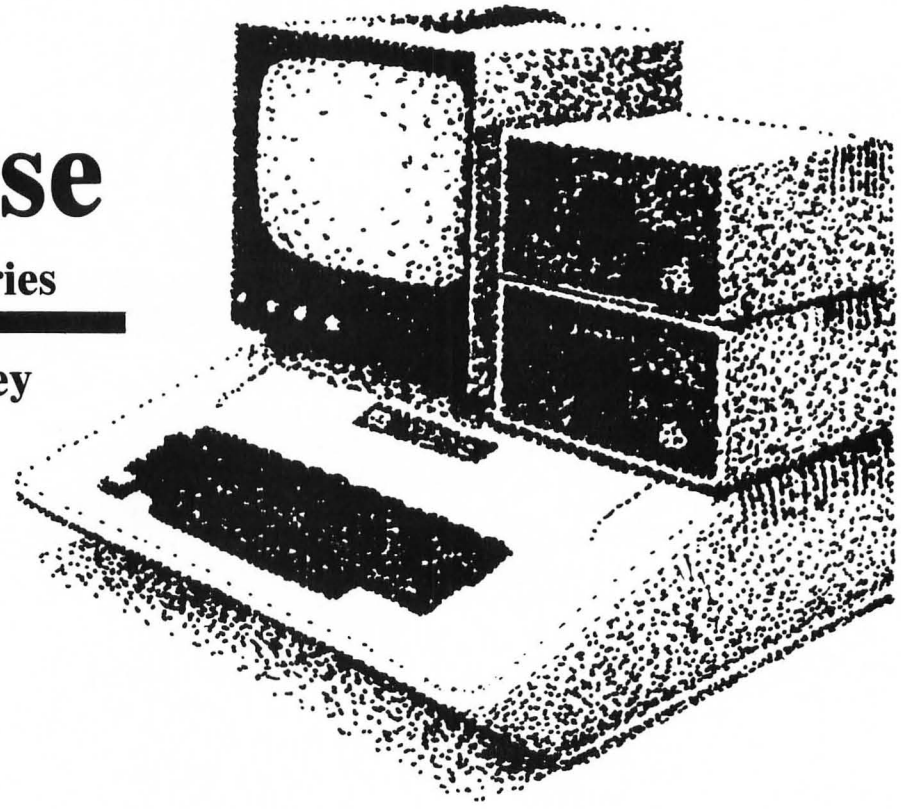
APPLE™ SWAT TABLE FOR:
BLACKJACK

LINES	SWAT CODE	LENGTH
100 - 200	HW	292
210 - 330	QI	400
340 - 470	WW	209
490 - 620	VX	191
640 - 750	SJ	210
760 - 860	GS	179
870 - 990	HQ	413
1000 - 1120	VW	300
1130 - 1250	JQ	243
1260 - 1380	HU	319
1390 - 1490	PE	283
1500 - 1610	ZH	213
1620 - 1760	MJ	250
1770 - 1880	AU	296
1890 - 2010	FJ	194
2020 - 2140	PM	293
2150 - 2220	TG	256
2230 - 2350	BB	346
2360 - 2470	HK	424
2480 - 3000	HC	245
3020 - 3130	UZ	311
3140 - 3190	NH	178

Apple Diskourse

Part Four Of A Series

by Cary W. Bradley



"...we can now investigate another aspect of DOS housekeeping, the Volume Table of Contents, or VTOC."

Again this month, we start with a stern warning — DO NOT attempt to run the program presented here on any nonexpendable disk before you have thoroughly tested and retested it in every possible way to make sure it is absolutely perfect. The techniques used are extremely risky, and even when the program runs perfectly it's possible to ruin a disk. Read the entire article before you do anything with the program. Then summon up your courage, use caution, and have fun learning some intimate facts about Apple DOS.

The Volume Table of Contents (VTOC)

Having looked at the way DOS stores file names, types, sector counts and starting locations in the disk catalog, we can now investigate another aspect of DOS housekeeping, the Volume Table of Contents, or VTOC. It is located on sector 0 of track 17 on every normally formatted disk, and is used to keep track of which sectors are in use and which ones are free.

You can refer to the DOS manual for detailed information and diagrams, but here is a general overview of how the VTOC is constructed:

The first 54 bytes of the VTOC (bytes 0-53) contain some general information about the version of DOS you are using and its operating characteristics. We won't concern ourselves with these in what we'll be doing in this article, but you can use the Disk Snooper utility to compare their contents to the values listed in the DOS manual. Bytes 54-195 contain the "track bit maps," and the remaining bytes are unused.

In the track-bit-map portion of the VTOC, each track is represented

by a group of four consecutive bytes, although only the first two of these are actually used. Of these two bytes, each bit stands for one sector of the track corresponding to the group of bytes. Beginning with the highest order bit of the first byte, and working down to the least significant bit of the second byte, the bits represent sectors 15 through 0 of the track. A zero bit shows that a sector is in use, and a one bit shows it is not. For example, the track 34-bit map can be found in bytes 192-195 of the VTOC. If sectors 15 through 6 of that track are part of an active file, (DOS allocates sectors from highest to lowest numbers.) then all eight bits of byte 192, and the two highest order bits of byte 193, would be zeros. The 6 remaining bits of byte 193 would be ones. If you were to examine this hypothetical VTOC with the *Disk Snooper,* you would see that byte 192 of track 17, sector 0, was a 0, while byte 193 was a decimal 63 (hex 3F). Bytes 194 and 195, being unused, would also be zeros.

If it sounds as though this is a rather roundabout way of determining which sectors are used and unused, it's probably because you've got BASIC on your mind. But DOS is a Machine Language program, and in Machine Language it's a simple matter to see which bits within a byte are ones, and which are not. Since we're working in BASIC in this series, we'll plod through the calculations, but we'll leave the tedious part to the computer.

When a file is created through DOS, the catalog entry is made, and whatever sectors are required are duly noted in the VTOC. By constantly referring to the VTOC, DOS knows which sectors not to bother when writing other files. When a file is DELETEd, the sectors it used are liberated for future use.

Putting this together with what we already know about the catalog, we can see that a file deletion is not necessarily final. The file's data are not physically erased by the DELETE command. Rather, a couple of bytes in the catalog entry and some bits in the VTOC are changed. It is often possible to get these things back to their original condition by using the RWTS subroutine. I say "often," because there are some obvious limitations. If either the catalog entry or one or more data sectors are overwritten by a DOS operation subsequent to the deletion, you'll probably not be able to get your file back; certainly not in its entirety. However, you can recover an accidentally deleted file immediately after you delete it, and sometimes later, with the right tools.

This month's utility is designed for that purpose. It simply reverses the deletion process. In the disk catalog, the byte which points to the first track occupied by the file when it was alive has been changed to 255 (hex FF), and the original value has been written at the end of the file name. In addition, all of the file's sectors have been marked with a one-bit in the VTOC. The program, "Recover," locates where these changes were made and undoes them.

"...you can recover an accidentally deleted file immediately after you delete it, and sometimes later, with the right tools."

Recover is modeled after Disk Snooper, and you'll recognize similar procedures if you've been using that utility. This one catalogs files in the same way, but instead of picking only active files out of the catalog, it picks only deleted ones. Each file is assigned a number, and you can designate which file you want to try to recover by either its name or its number.

The Recovery Process

The first step in the recovery process is a "File Integrity Check." This serves several purposes. It steps through the file, as directed by the track/sector list, and determines whether each sector is still free. This means that you can possibly get back a deleted file even if some other things have been written to the disk in the meantime. If any conflicts are found, they are counted, and you are warned before the recovery attempt is made. The total number of sectors in the file is also counted, and compared to the number recorded in the catalog. This check is included because it is possible for an original track/sector list to have been written over by another track/sector list, which would direct us to the wrong file's data.

During the integrity check, the track and sector number of each sector in the file is displayed. Sectors containing track/sector list data are shown in inverse. They are first checked against the VTOC to see whether they are free sectors. Then they are scanned to assure that the data they contain are legitimate t/s list data. If a track/sector list has been written over by something else, there's no way to find the rest of the file, so the recovery is aborted. If the numbers are track/sector numbers, they are examined in pairs to see if the sectors they point to are free. All track and sector numbers are listed on the video screen, followed by an "OK" if they are free and by an "IN USE" if not. The results of the integrity check are printed upon its completion.

Besides being an interesting exercise in the use of the RWTS subroutine, the integrity check eliminates the need to do a lot of error checking during the file recovery process. A recoverable file should always pass the integrity check, but a successful check does not guarantee a successful recovery. However, the odds of an unrecoverable file passing the integrity check are slim.

After the check is completed, whether successful or not, you are given the option of cancelling or proceeding with the recovery attempt. This allows you the opportunity to recover a file even though one or more of its sectors may conflict with another file. *Use extreme caution* in doing this, because the resulting shared sectors can be disastrous. Deleting either file will free any shared sectors, and the ultimate result is unpredictable, but always bad. If you have deleted a valuable file, and the integrity check shows that there are sector conflicts, you may be able to partially recover the file. But FIRST, make a copy of

Translation of the Month Contest

You could win a FREE *SoftSide* DV or CV Subscription!

SoftSide's* Translation of the Month** has been so well received by our readers, we're offering a greater author incentive than ever before. No, we can't give you a job at the U.N., but we will award a one-year subscription to ***SoftSide DV or an 18-month subscription to ***SoftSide CV*** for a high-quality translation of one of our past programs. That's a value of $125 for the ***Disk Version*** or $112.50 for the ***Cassette Version*** — you'll be rewarded every month for your translation efforts!

Here are some of the most important qualifications we look for in a translation winner.

Your entry must be a translation of one of the featured programs from a past issue of ***SoftSide***. (We're particularly interested in Apple™, ATARI® and IBM® PC translations of some of our older TRS-80® only issues. Write for a list of suggested candidates.) In general, we're looking for translations of programs which are a CHALLENGE to translate. Some of the programs we publish are written in more or less "generic" BASIC, which can be typed into another computer with very few changes. Although these programs require the least effort to translate, they are also the least likely candidates for contest winners.

Your translation should be thoroughly tested and completely bug-free. Just converting program lines doesn't automatically ensure a workable translation. Be sure to use-test your translation as carefully as you would test a program you had written entirely from scratch.

Your translation should fully utilize the unique features of the computer for which it is written. The objective of a translation is to "fit" the capability and convention of its host computer, not simply mechanically duplicate the operation of the original program. This is especially true of programs which use graphics, and should be kept in mind for such minor features as keyboard layout (use of such special keys as arrows, ESC, CTRL, CLEAR, etc.). Also be careful with screen formatting; a word that spills over into the next line because of a PRINT statement that wasn't properly rewritten betrays such carelessness that we'll probably reject your translation automatically.

Your entry should incorporate any improvements and enhancements you can add to the original program. Don't feel that you have to limit yourself to the boundaries of the original. (On the other hand, don't go overboard and destroy the character of the original by completely rewriting it!) An enhanced translation is much more likely to catch our attention than a line-for-line duplicate, and it will have more value to our readers.

It's not necessary to include extensive documentation with your translation, only that which is different from the original. If most of the originally published documentation applies to your translation, simply say so. You should, however, include descriptions and explanations of any changes or enhancements you've made.

All **Translation Contest** entries must be submitted on disk, with documentation in printed or typed form. Disks will be returned only if accompanied by a self-addressed, stamped envelope. Send your entries to:

SoftSide

Translation Contest
6 South Street, Milford, NH 03055

the entire disk, and try the recovery on the copy. If you are able to reconstruct the file, use a normal DOS procedure (such as FID) to get it back to your original disk, so that any sector conflicts will be eliminated. In fact, it is a good idea to make a backup copy of any disk before you attempt to recover a deleted file.

The second step in the recovery process is to reconstruct the catalog entry. This happens quickly, as you will see when you run the program. The third step, updating the VTOC, can take a few minutes, depending on the length of your file. This is because we have to convert back and forth from decimal to binary for each of the file's sectors. The longest file I tested was 434 sectors long, and it took about 2 1/2 minutes to update the VTOC. Be patient.

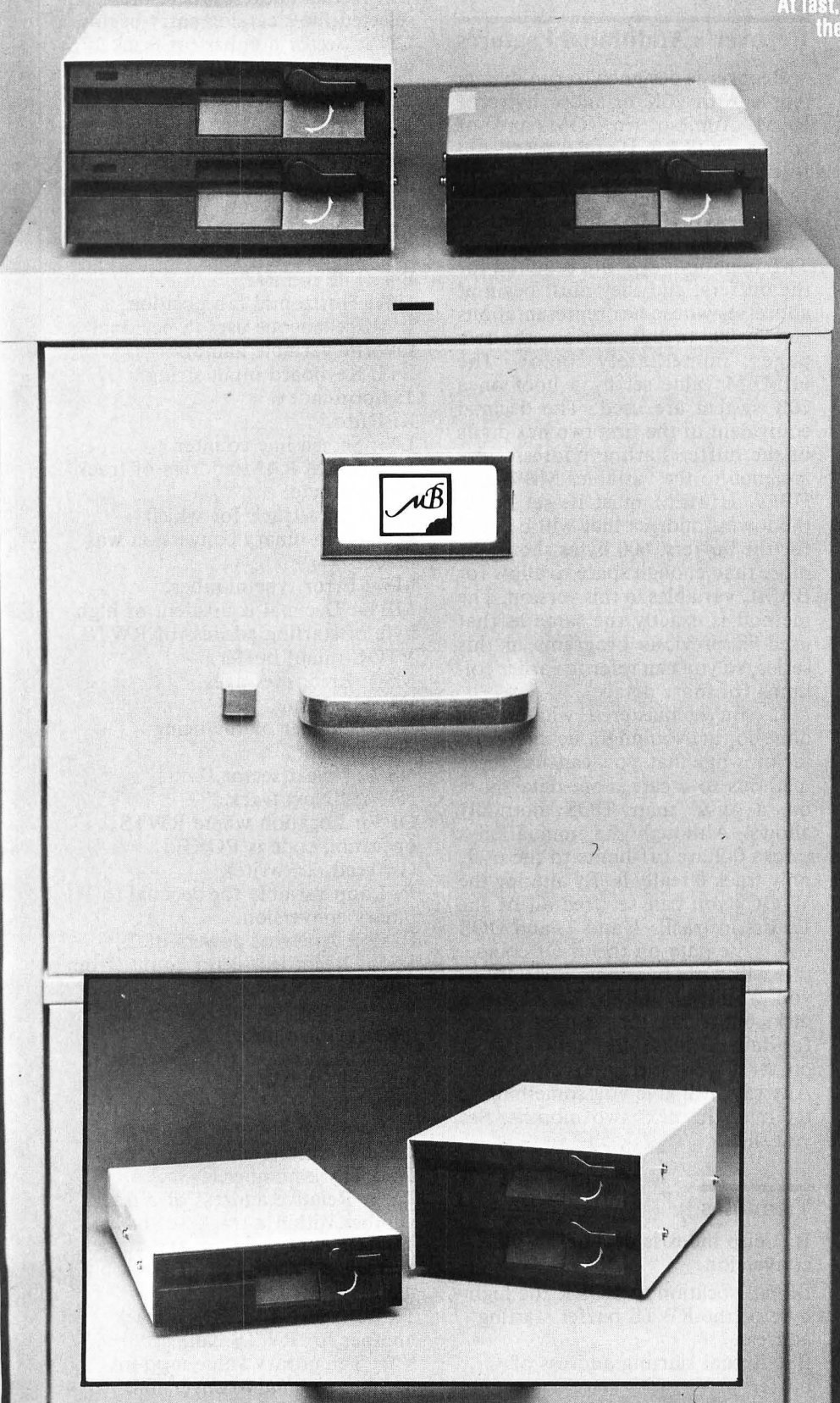
If you happen to have a deleted file that's over 255 sectors long, you will notice that the integrity check gives you a true sector count, rather than the MOD 256 count shown in the DOS catalog. Actually, the catalog entry contains the true count, in bytes 33 and 34, relative to the beginning of the catalog entry. (See Appendix C of the DOS manual.) Only the first of these is displayed by CATALOG.

Another interesting tidbit: The limit on the length of names for deleted files is 29 characters, rather than the regular 30. This is because, when a file is deleted, the last byte of the catalog space normally reserved for the file name is used to hold the number of the track where the track/sector list begins. If, for some perverse reason, your original file had a 30-character name, it will be shortened by one character when you use this utility.

You can discover many things about DOS by fooling around with this method. But remember that doing so can be hazardous to the health of your disks. Painstakingly thorough testing should be done on an expendable disk before this, or any similar utility, is run on one of your good disks. I clobbered more disks in creating this program than in anything else I've ever done on my Apple. Of course, they were all created to be clobbered, so nothing was really lost. Most of them bit the dust because of a simple, one-letter

continued on page 116

Space Saving Storage



At last, Microbyte has created the perfect storage unit for the Apple II. This new data drive slimline out performs any of its competition. This pint-size drive, works 8 times faster than most other drives, saving you space and time. Compare the features of the new ASAP slimline to what you're using now. We're sure you'll agree, this slimline out performs the others.

- 8 times faster than APPLE II Drives.
- Direct drive motor of extremely high quality.
- Only ½ the size of a regular drive.
- 100% APPLE[1] compatible (including "half track").
- Can be used with IBM PC[2] and other computers by simply disconnecting Interface Board.
- Mechanism and read - write electronics made by TEAC the world leader in Audio recording machines.
- 1 (one) full year warranty at no extra cost!!!!!
- The SLIMLINE is a 35/40 track drive and can take advantage of our ENHANCER Diskette which will give the user 15% more storage capacity & up to 163K Bytes.
- Slimline ½ height 163K.
- Dual Slimline drives ½ height in one box.
- Slimline ½ height (1) Megabyte with controller.

Contact your local dealer for more information.
Dealer inquiries invited.

MICROBYTE

1198 E. Willow, Signal Hill, CA 90806

Call Toll Free (800) 421-7701
In California (213) 595-6431
or (714) 891-2663

[1]APPLE is TM of APPLE Corporation. [2]IBM is TM of International Business Machines.

typo buried in the middle of my code. It could easily happen to you, too, so please be careful.

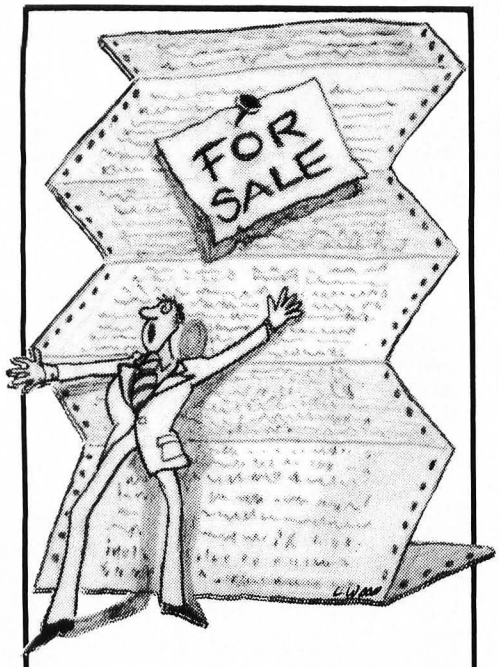
Recover's Additional Features

Recover is designed to run on any Apple with 20K or more bytes of RAM, Applesoft in ROM, and, of course, DOS 3.3. If you want to add features to the program, you can easily alter the memory areas used for RWTS buffers and BASIC variables by noting the following: Two memory pages are required for the buffers, and they must begin at addresses whose hex representations end in 00. In this version, the two pages immediately below the HIMEM value set by a boot on a 20K system are used. The decimal equivalent of the first two hex digits of the buffer starting addresses are assigned to the variables MB% and DB%. HIMEM must be set below the lowest address that will be used for the buffers. 600 bytes should be more than enough space to allow for BASIC variables in this version. The method is exactly the same as that used in previous programs in this series, so you can refer to earlier columns for more detail.

If you've mastered what we've done so far, you might be interested in knowing that you can use these methods to create more data space on a disk than DOS normally allows. Although the manual says tracks 0-2 are off-limits to the user, only track 0 really is. By altering the VTOC, you can set free all of the sectors in tracks 1 and 2, and DOS will write data on them. Of course, this will write over part of DOS, so you won't be able to boot such a disk, but a disk that is used strictly for data does not need to have DOS on it if your program disk does. Anyway, it'll give you something to try over the next two months. See you then.

Variables

B: Loop index is decimal to binary conversion.
BF%: Location to POKE the high byte of the RWTS buffer starting address.
BS: Actual starting address of buffer.
BY%: Value of a track bit map byte.
CF%: Sector conflict counter.
CP: Actual address where the selected file's catalog entry begins.
CS%: Sector number (of track 17) where the selected file's catalog entry resides.
C$(): Menu choices.
DB%: Decimal equivalent of high byte of starting address of RWTS data buffer.
E%: Error flag.
ER%: Address where RWTS error code is stored.
F%: File counter.
H%: Horizontal tab position.
I: Miscellaneous uses (a personal favorite variable name).
IN$: Keyboard input string.
J: Loop index.
K: Ditto.
L%: Screen line counter.
LC: Actual RAM address of track bit map byte.
LT%: Last track for which decimal to binary conversion was made.
M%: Error type number.
MB%: Decimal equivalent of high byte of starting address of RWTS VTOC (map) buffer.
M$(): Error messages.
N$: File name.
N%: Number of file being searched for.
NS%: "Next sector."
NT%: "Next track."
OP%: Location where RWTS operation code is POKEd (1 = read, 2 = write).
P: Loop variable for decimal to binary conversion.
P2%(): Array of powers of 2.
RS%: Recorded sector count (from catalog).
RW%: Address for RWTS call.
S%: Sector number.
SC%: Address to POKE sector number for RWTS.
SS%: Two uses — sector counter and sum variable for binary to decimal conversion.
T%: Track number.
T1%: Relative address of a track number within a track/sector list.
TB%(): Array of ones and zeroes, corresponding to bits of track bit map.
TK%: Address to POKE track number for RWTS call.
V%: Temporary value used in decimal to binary conversion.



SELL YOUR PROGRAM

AND KEEP IT TOO!

One of the nicest things about selling your program to **SoftSide** is that it's still your program after we buy it. Actually, what we are buying is the right to publish your program once in our magazine and on subscription disk and tape. This is what we call "one-time rights." Three months after your program appears in **SoftSide**, you are free to sell it again to anyone. And, now that it's been published, your program is worth more. So send today for a copy of our free author's guide and find out how you can sell your program and keep it too.

Write to:

SoftSide Publications
Dept AG1
6 South Street
Milford, NH 03055

```
SS SS SS SS SS SS SS SS SS SS SS
SS                             SS
SS      APPLESOFT BASIC        SS
SS         'RECOVER'           SS
SS  AUTHOR: CARY W. BRADLEY    SS
SS     COPYRIGHT (C) 1982      SS
SS SOFTSIDE PUBLICATIONS, INC SS
SS                             SS
SS SS SS SS SS SS SS SS SS SS SS
```

If you don't wish to type this program, it is also included on this month's SoftSide DV.

Jump to beginning.

```
1  GOTO 1300
```

Subroutine to convert from decimal to binary.

```
10 LC = 256 * MB% + 4 * T% + 56: FOR
     B = 1 TO 0 STEP  - 1:BY% =  PEEK
     (LC + 1 - B)
20  FOR P = 7 TO 0 STEP  - 1
30 V% = BY% - P2%(P)
40  IF V% >  = 0 THEN TB%(P + 8 *
     B) = 1:BY% = V%: NEXT P,B: RETURN
50 TB%(P + 8 * B) = 0: NEXT P,B: RETURN
```

Subroutine to convert from binary to decimal.

```
60 SS% = 0: FOR P = 0 TO 7:SS% =
     SS% + TB%(P) * P2%(P): NEXT
     P: POKE LC + 1,SS%
70 SS% = 0: FOR P = 8 TO 15:SS% =
     SS% + TB%(P) * P2%(P - 8): NEXT
     P: POKE LC,SS%: RETURN
```

Subroutine to count sectors and determine position on screen.

```
80 SS% = SS% + 1:L% = L% + 1
90  IF L% < 23 THEN  RETURN
100 L% = 7: IF H% = 1 THEN H% = 2
     1: RETURN
110  VTAB 24: HTAB 8: GOSUB 1540:
      HOME :H% = 1: RETURN
```

Deleted file catalog.

```
200  HOME : POKE OP%,1: POKE TK%,
     17: POKE SC%,15: POKE BF%,DB
     %: CALL RW%: GOSUB 1580
210  PRINT "DELETED FILES";: HTAB
     25: PRINT "DISK VOLUME "; PEEK
     (791): POKE 34,2: HOME
220 L% = 0:F% = 0:BS = 256 * DB%
230  FOR I = BS + 11 TO BS + 221 STEP
     35:T% =  PEEK (I): IF T% <  >
     255 THEN 280
240  IF L% = 18 THEN  GOSUB 1530
250 L% = L% + 1:F% = F% + 1
260  PRINT  CHR$ (91) +  RIGHT$ (
     "  " +  STR$ (F%) +  CHR$ (9
     3),4); SPC( 4);
270  FOR J = I + 3 TO I + 31: PRINT
      CHR$ ( PEEK (J));: NEXT : PRINT
280  NEXT
290 NT% =  PEEK (BS + 1):NS% =  PEEK
     (BS + 2): IF NS% = 0 THEN  PRINT
     : IF F% = 0 THEN  PRINT "NON
     E FOUND": PRINT
300  IF NS% = 0 THEN : GOSUB 1540
     : TEXT : RETURN
```

File integrity check.

```
310  POKE TK%,NT%: POKE SC%,NS%: CALL
     RW%: GOSUB 1580: GOTO 230
400  HOME : INVERSE : PRINT "FILE
     :";: NORMAL : PRINT " "N$: POKE
     34,4
410  HOME : PRINT "FILE INTEGRITY
      CHECK": POKE 34,6: HOME
420  POKE OP%,1: POKE BF%,MB%: POKE
     TK%,17: POKE SC%,0: CALL RW%
     : GOSUB 1580: POKE BF%,DB%
430 SS% = 0:RS% = 256 *  PEEK (I +
     34) +  PEEK (I + 33):L% = 7:
     H% = 1:CF% = 0:S% =  PEEK (I
      + 1):T% =  PEEK (I + 32):LT
     % = T%: GOSUB 10
440 NT% = T%:NS% = S%
450  IF NS% > 15 OR NT% > 34 OR (
     NT% < 1 AND NS% > 0) THEN 61
     0
460  IF NS% = 0 AND NT% = 0 THEN
      VTAB 24: HTAB 8: GOSUB 1540
     : GOTO 630
470  VTAB L%: HTAB H%: INVERSE : PRINT
     NT%"/"NS%;: NORMAL
480  IF T% <  > LT% THEN  GOSUB 1
     0
490  IF TB%(NS%) THEN  PRINT " OK
     ": GOTO 510
500  PRINT " IN USE":CF% = CF% +
     1
510  GOSUB 80: POKE TK%,NT%: POKE
     SC%,NS%: CALL RW%: GOSUB 158
     0:T1% = 12
520 T% =  PEEK (BS + T1%): IF T% =
     0 THEN 590
530 S% =  PEEK (BS + T1% + 1): IF
     S% > 15 OR T% > 34 THEN 610
540  VTAB L%: HTAB H%: PRINT T%"/
     "S%;
550  IF T% <  > LT% THEN  GOSUB 1
     0:LT% = T%
```


EDUCATIONAL ACTIVITIES, INC.

Teacher Authored Programs Classroom and Home for TRS-80, APPLE II, PET

MATH • LANGUAGE ARTS READING • SPELLING LITERACY • PROGRAMMING

EASY TO USE
TUTORIAL & PRACTICE
USEFUL FOR ALL AGES
FULL COMPUTER CAPABILITIES
EDUCATIONALLY SOUND
REMEDIAL & DEVELOPMENTAL

Send for FREE complete microcomputer software catalog.

EDUCATIONAL ACTIVITIES, INC.
P.O. Box 87, Baldwin, New York 11510
(516) 223-4666
CALL TOLL FREE OUTSIDE N.Y. STATE
800-645-3739
IN CANADA
CORONET INSTRUCTIONAL MEDIA, Ltd.
200 Steelcase Road East
Markham, Ontario L3R 1G2

NEW

COLOR SLIDES FROM YOUR APPLE*

COMPUTER SLIDE EXPRESS™

Turns your Apple II* Hi Res Graphics into 35mm Color Slides

Have slides made from:
- **Apple Business Graphics***
- **Executive Briefing System†**
- **Visiplot‡**
- **Other 33 or 34 Sector Binary Picture Files**

Slides for
•Meetings•Conferences •Lectures•Trade Shows
for only **$6.00 per slide**
($30.00 minimum)
Turnabout Time - 5 Days
For information call or write:
VISUAL HORIZONS
180 Metro Park, Rochester, NY 14623
(716) 424-5300

*Trademarks of Apple Computer Corp. †Trademark of Lotus Corp. ‡Trademark of Personal Computer, Inc. Computer Slide Express is a trademark of Visual Horizons, Inc.

Apple Diskourse ***continued***

```
560  IF TB%(S%) THEN  PRINT " OK"
     : GOTO 580
570  PRINT " IN USE":CF% = CF% +
     1
580  GOSUB 80
590 T1% = T1% + 2: IF T1% < 255 THEN
     520
600 NT% =  PEEK (BS + 1):NS% =  PEEK
     (BS + 2): GOTO 450
610  HOME : PRINT  CHR$ (7)"TRACK
     /SECTOR LIST CLOBBERED.": PRINT
620  PRINT "FILE RECOVERY ABORTED
     ...";: VTAB 24: HTAB 9: GOSUB
     1540: RETURN
630  HOME : PRINT CF%" SECTOR CON
     FLICT";: IF CF% <  > 1 THEN
      PRINT "S";
640  PRINT " DETECTED": PRINT : PRINT
     "SECTOR COUNTS:"
650  PRINT "CATALOG="RS%,"CHECK="
     SS%: PRINT
660  IF CF% OR RS% <  > SS% THEN
      FLASH : PRINT  CHR$ (7)"WAR
     NING: FILE IS NOT INTACT..."
     : NORMAL
670  VTAB 14: INPUT "CONTINUE WIT
     H RECOVERY?(Y/N):";IN$
680  IF  LEFT$ (IN$,1) = "N" THEN
      RETURN
690  IF  LEFT$ (IN$,1) <  > "Y" THEN
      PRINT  CHR$ (7): GOTO 670
```

Recovery attempt.

```
700  POKE 34,4: HOME : PRINT "REC
     OVERY ATTEMPT UNDERWAY.": PRINT

710  PRINT "CHANGING CATALOG ENTR
     Y...": PRINT :BS = DB% * 256
720  POKE TK%,17: POKE SC%,CS%: CALL
     RW%: GOSUB 1580
730  POKE CP, PEEK (CP + 32): POKE
     CP + 32,160: POKE OP%,2: CALL
     RW%: GOSUB 1580
740  PRINT "UPDATING VTOC. PLEASE
      WAIT...": PRINT
750 T% =  PEEK (CP):S% =  PEEK (C
     P + 1):LT% = T%: GOSUB 10
760 TB%(S%) = 0: GOSUB 60
770 NT% = T%:NS% = S%
780  POKE TK%,NT%: POKE SC%,NS%: POKE
     BF%,DB%: POKE OP%,1: CALL RW
     %: GOSUB 1580:T1% = 12
790 T% =  PEEK (BS + T1%): IF T% =
     0 THEN 820
800  IF T% <  > LT% THEN  GOSUB 1
     0:LT% = T%
810 S% =  PEEK (BS + T1% + 1):TB%
     (S%) = 0: GOSUB 60
820 T1% = T1% + 2: IF T1% < 255 THEN
     790
830 NT% =  PEEK (BS + 1):NS% =  PEEK
     (BS + 2)
840  IF NT% = 0 AND NS% = 0 THEN
     870
850 T% = NT%:S% = NS%: IF T% <  >
     LT% THEN  GOSUB 10:LT% = T%
860 TB%(S%) = 0: GOSUB 60: GOTO 7
     80
870  POKE TK%,17: POKE SC%,0: POKE
     OP%,2: POKE BF%,MB%: CALL RW
     %: GOSUB 1580: PRINT "FINISH
     ED.": VTAB 24: HTAB 8: GOSUB
     1540: RETURN
```

Select file to be recovered.

```
900  HOME : VTAB 6: INVERSE : HTAB
     17: PRINT " RECOVER "
910  VTAB 8: HTAB 6: PRINT " DELE
     TED FILE RECOVERY UTILITY ":
      NORMAL
920  VTAB 11: HTAB 8: PRINT "ENTE
     R FILE NAME OR NUMBER:"
930  VTAB 16: HTAB 11: PRINT "YOU
      MAY ALSO ENTER:"
940  VTAB 18: HTAB 11: PRINT "'CA
     T' FOR CATALOG": HTAB 11: PRINT
     "OR <RETURN> TO END"
950  POKE  - 16368,0: VTAB 13: HTAB
     8: INPUT "";IN$: IF IN$ = ""
      THEN  RETURN
960  IF IN$ = "CAT" THEN  GOSUB 2
     00: GOTO 900
970  IF  LEN (IN$) > 29 THEN IN$ =
      LEFT$ (IN$,29)
980  IF  LEFT$ (IN$,1) < "0" OR  LEFT$
     (IN$,1) > "9" THEN 1100
990 E% = 0: FOR I = 1 TO  LEN (IN
     $)
1000  IF  MID$ (IN$,I,1) < "0" OR
      MID$ (IN$,I,1) > "9" THEN E
     % = 1
1010  NEXT : IF E% THEN M% = 0: GOSUB
     1500: GOTO 900
1020 N% =  VAL (IN$): POKE TK%,17
     : POKE SC%,15: POKE OP%,1: POKE
     BF%,DB%:F% = 1:BS = 256 * DB
     %
1030  CALL RW%: GOSUB 1580:I = BS
      + 11
1040 T% =  PEEK (I): IF T% <  > 2
     55 THEN 1070
1050  IF N% = F% THEN N$ = "": FOR
     J = I + 3 TO I + 31:N$ = N$ +
      CHR$ ( PEEK (J)): NEXT :CS%
      =  PEEK (SC%):CP = I: GOTO
     400
1060 F% = F% + 1
1070 I = I + 35: IF I <  = BS + 2
     21 THEN 1040
1080 NT% =  PEEK (BS + 1):NS% =  PEEK
     (BS + 2): IF NS% = 0 THEN M%
      = 1: GOSUB 1500: GOTO 900
1090  POKE TK%,NT%: POKE SC%,NS%:
      GOTO 1030
1100  POKE TK%,17: POKE SC%,15: POKE
     BF%,DB%: POKE OP%,1:BS = 256
      * DB%
1110  CALL RW%: GOSUB 1580:I = BS
      + 11
1120 T% =  PEEK (I): IF T% <  > 2
     55 THEN 1190
1130 J = I + 2:K = 1
1140  IF  MID$ (IN$,K,1) <  > CHR$
     ( PEEK (J + K) - 128) THEN 1
     190
1150 K = K + 1: IF K <  =  LEN (I
     N$) THEN 1140
1160  IF  PEEK (J + K) <  > 160 THEN
     1190
1170 K = K + 1: IF K <  = 29 THEN
     1160
1180 N$ = IN$:CS% =  PEEK (SC%):C
     P = I: GOTO 400
1190 I = I + 35: IF I <  = BS + 2
     21 THEN 1120
1200 NT% =  PEEK (BS + 1):NS% =  PEEK
     (BS + 2): IF NS% = 0 THEN M%
      = 1: GOSUB 1500: GOTO 900
1210  POKE TK%,NT%: POKE SC%,NS%:
      GOTO 1110
```

Program begins here.

```
1300  POKE 235, PEEK (115): POKE
     236, PEEK (116)
1310  DIM M$(1),TB%(15),P2%(7)
1320  HIMEM: 9215
1330 RW% = 768:TK% = 781:SC% = 78
     2:BF% = 786:OP% = 789:ER% =
     790:DB% = 36:MB% = 37
1340  FOR I = 0 TO 2: READ C$(I):
      NEXT
1350  FOR I = 0 TO 1: READ M$(I):
      NEXT
1360  FOR I = 0 TO 7: READ P2%(I)
     : NEXT
```

```
1370  FOR I = RW% TO RW% + 29: READ
      T%: POKE I,T%: NEXT
1380  TEXT : HOME : HTAB 17: INVERSE
      : PRINT " RECOVER ": NORMAL
      : PRINT
1390  PRINT "  DOS 3.3 DELETED FI
      LE RECOVERY PROGRAM"
1400  VTAB 7: HTAB 18: INVERSE : PRINT
      " MENU ": NORMAL
1410  FOR I = 0 TO 3: VTAB 11 + 2
      * I: HTAB 11: PRINT C$(I);
1420  HTAB 11: INVERSE : PRINT  LEFT$
      (C$(I),1);: NORMAL : NEXT
1430  VTAB 20: HTAB 15: PRINT "SE
      LECTION:";: CALL  - 868: POKE
       - 16368,0
1440  GET IN$: PRINT IN$
1450  IF IN$ = "C" THEN  GOSUB 20
      0: GOTO 1380
1460  IF IN$ = "R" THEN  GOSUB 90
      0: GOTO 1380
1470  IF IN$ = "Q" THEN  POKE 115
      , PEEK (235): POKE 116, PEEK
      (236): HOME : PRINT "END REC
      OVER": END
```

Miscellaneous short subroutines.

```
1480  PRINT  CHR$ (7): GOTO 1430
1500 : VTAB 14: HTAB 8: FLASH
1510  PRINT  CHR$ (7);M$(M%): FOR
      I = 1 TO 3000: NEXT
1520  NORMAL : TEXT : RETURN
1530 L% =  - 1: VTAB 24: PRINT "O
      R <ESC> TO ABORT";: VTAB 23:
       HTAB 1: PRINT "(MORE) ";
1540  PRINT "PRESS ANY KEY TO CON
      TINUE";: POKE  - 16368,0
1550  IF  PEEK ( - 16384) < 128 THEN
      1550
1560  IF  PEEK ( - 16384) = 155 AND
      L% =  - 1 THEN  POP : TEXT
1570  HOME : RETURN
1580 E% =  PEEK (ER%): IF E% <  >
      16 AND E% <  > 32 AND E% <  >
      64 AND E% <  > 128 THEN  RETURN

1590  INVERSE : VTAB 23: HTAB 14:
       PRINT  CHR$ (7);" DISK ERRO
      R "
1600  NORMAL : HTAB 8: GOSUB 1540
      : POKE ER%,255: POP : RETURN
```

The data.

```
1610  DATA CATALOG DELETED FILES,
      RECOVER A FILE,QUIT
1620  DATA ILLEGAL ENTRY,FILE NOT
       FOUND
1630  DATA 1,2,4,8,16,32,64,128
1640  DATA 169,3,160,9,32,217,3,9
      6,0,1,96,1,0,0,0,26,3,0,53,0
      ,0,1,255,0,96,1,0,1,239,216
```

APPLE™ SWAT TABLE FOR: **RECOVER**

LINES	SWAT CODE	LENGTH
1 - 110	PK	342
200 - 310	VE	405
400 - 510	CU	438
520 - 630	LP	363
640 - 750	PW	467
760 - 870	TY	387
900 - 1010	CX	426
1020 - 1130	AM	405
1140 - 1330	VA	374
1340 - 1450	XZ	333
1460 - 1580	FI	355
1590 - 1640	WE	253

Ed•Com /SPRING '83

APRIL 28-30, 1983
WASHINGTON D.C. CONVENTION CENTER

NATIONAL COMPUTER CONFERENCE AND EXPOSITION FOR EDUCATORS AT ALL LEVELS

SEMINARS HANDS-ON SESSIONS MICROCOURSES PANELS DEMONSTRATIONS

• SOFTWARE • HARDWARE • FUNDING • PROGRAM DESIGN
DATA MANAGEMENT • BID SPECIFICATION • INSTRUCTIONAL MANAGEMENT
WORD PROCESSING • HIGHER EDUCATION APPLICATION • MATH AND SCIENCE
MICRO-MINI CONNECTION • TELECOMMUNICATION • COMPUTER ASSISTED INSTRUCTION
SIMULATION • DISTRIBUTED DATA PROCESSING • PURCHASE AND POLICY • TEACHER TRAINING

MAIL TO: Judco Computer Expos, Inc.
2629 North Scottsdale Road, Suite 201, Scottsdale, Arizona 85257
(800) 528-2355

Name_____
Address_____
City_____
State_____ Zip_____
[ ] ATTENDEE INFORMATION
[ ] INFORMATION TO EXHIBIT 01

Ed•Com /FALL '83

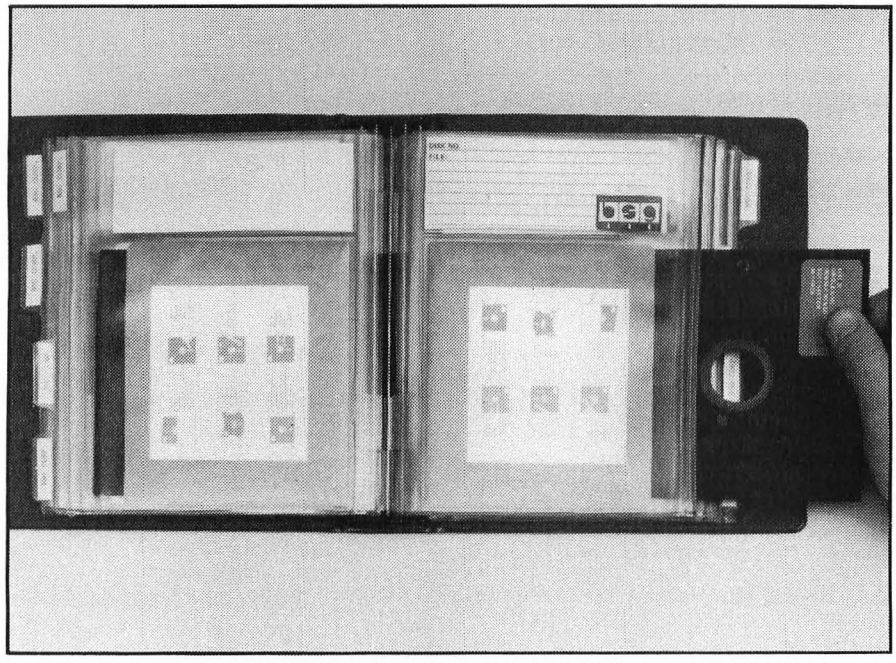
NOVEMBER 17-19, 1983
Los Angeles Convention Center

More Games For Your Apple®

FROBCO
P.O. Box 2780
Santa Cruz, CA 95063
(408) 429-5138

The Frob is an Apple II game development system that interfaces directly to the Atari® VCS 2600 game console. The package features a single board unit which conveniently plugs into any available slot of an Apple 48K host. *The Frob* allows a game designer to edit and assemble Atari VCS programs on the Apple, using any of the currently available Apple-based 6502 assemblers and editors.

With *The Frob* comes a User's Guide and a one-year update service on the software development tool kit. *The Frob* owner also receives a year's subscription to the game designer communique, *The FROBBER*. *The Frob* is available for a retail price of $995 from FROBCO.



Dust Protection For Your Disks

BSG CORPORATION
7674 Washington Avenue South
Minneapolis, MN 55344
(612) 944-8180

The *DUSTLOC*™ hard cover disk storage album protects disks from environmental damage caused by dust. Each page of the album is heavy gauge vinyl with a locking device. When sealed, this lock totally encloses each disk, preventing dust from collecting on the magnetic media. In addition, each page is indexed for easy identification. The album's spine assembly allows independent disk suspension for quick removal.

Standard *DUSTLOC* albums house 10, 20 or 40 disks and are available for 5 1/4 and 3 1/2 inch sizes. Albums for 8 inch disks are also available on request. Album prices start at $29.95. For more information, or to order the *DUSTLOC,* contact BSG Corporation.

Do Your Own Taxes Like an Expert

MICROMATIC PROGRAMMING COMPANY
P.O. Box 158
Georgetown, CT 06829
(203) 324-3009

Tax/Saver offers a different approach to preparing a tax return. It has the tax regulations programmed in, making it more than just a calculator.

Tax/Saver asks you questions, just like an accountant does. Based on your answers, it leads you through the tax maze to your lowest legal tax. Then it tells you how to fill out your return, line by line, or it will output to a printer. Special screen formatting makes data entry, verification and correction easy. You are always in control, and can skip features of the program that you don't need.

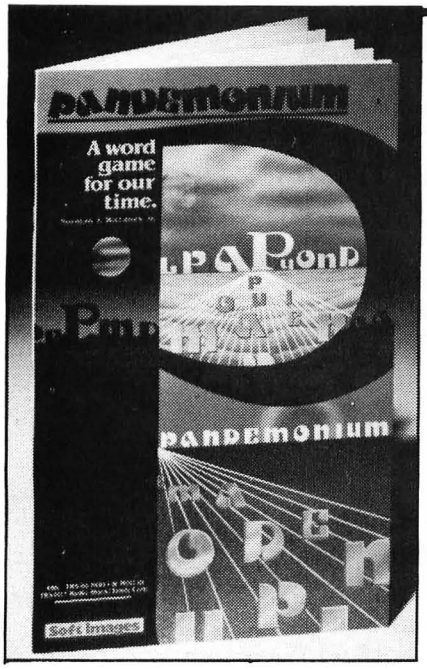
Tax/Saver is available in two versions — I and II. It requires a TRS-80® Model I or III with 32K and two disk drives. The price of *Tax/Saver* I is $89.95 (manual included). *Tax/Saver* II retails for $139.95 (manual included). Both packages are available from Micromatic Programming Company. Please include $3.50 to cover postage and handling.

Let Your PC Talk To Your Apple®

ALPHA SOFTWARE CORPORATION
6 New England Executive Park
Burlington, MA 01803
(617) 229-2924

The Apple-IBM Connection is the first software package enabling computer users to transfer information back and forth easily between the Apple II and the IBM® PC. In minutes, the software duplicates files created on one machine and transfers them into files that can be used on the other machine. In addition to transferring files, the package also sends and receives electronic mail messages between the two computers, as well as from Apple to Apple and IBM to IBM.

The Apple-IBM Connection is designed for non-technical users and includes complete documentation as well as a cassette demonstration tape. It is priced at $195 and is available in Computerland stores as well as many independent computer retail stores offering the IBM PC.



Challenging New Word Game Introduced

SOFT IMAGES
200 Route 17
Mahwah, NJ 07430
(201) 529-1440

PANDEMONIUM is an educational game containing a 6000 word dictionary. It requires a considerable amount of strategy and playing skill and can be played by any number of players. Possible letter combinations are endless, insuring that no two games are alike. The object of the game is to place twenty-five randomly generated letters onto a five by five playing board matrix in order to construct three, four and five letter words, arranged horizontally, vertically and along both top to bottom diagonals.

PANDEMONIUM runs on a 48K Model I or III with disk drive, or a 48K Apple II or II+ with Applesoft and DOS 3.3. The retail price is $39.95.

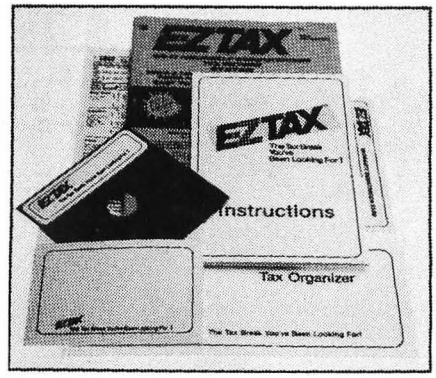
SoftSide's New Products section is an effort to inform our readers of newly available microcomputer products. However, ***SoftSide*** assumes no responsibility for product quality, company reliability or data accuracy. The information printed is submitted by promoters and selected for publication on a first-come, editorial preference, and computer compatible basis. Due to space limitations, we reserve the right to modify content of submissions. Submissions may be sent to:

***SoftSide* Magazine**
New Products Manager
6 South Street
Milford, NH 03055

The Tax Break You've Been Looking For

E-Z TAX
2444 Moorpark Avenue
San Jose, CA 95128
(408) 264-1040

E-Z TAX is a complete personal income tax preparation program. It has the capability of doing 1040-A (Short Form), 1040-EZ (the new IRS form for single taxpayers), and over 25 IRS forms and schedules (Long Form). The program includes a minimal-error checkpoint feature. After each major section of the income tax form is completed, the program re-displays all information for proof-reading and correction, allowing the user to correct any errors.



E-Z TAX is compatible with the Apple®, Atari® and IBM® PC, and requires 48K RAM memory and one disk drive (printer recommended). Its price is $69.95.

Home Control System For The Apple®

COMPU-HOME SYSTEMS, INC.
3333 E. Florida Avenue
Denver, CO 80210
(303) 777-6600

TomorrowHouse is the first computerized, turnkey home monitoring and control system within reach of the average consumer regardless of technical ability or familiarity with computers. It is based on the Apple II and consists of a specially built plug-in circuit card, sensors, a junction box and related hardware for easy hookup, all the programs to set up and control any house, and installation and user manuals.

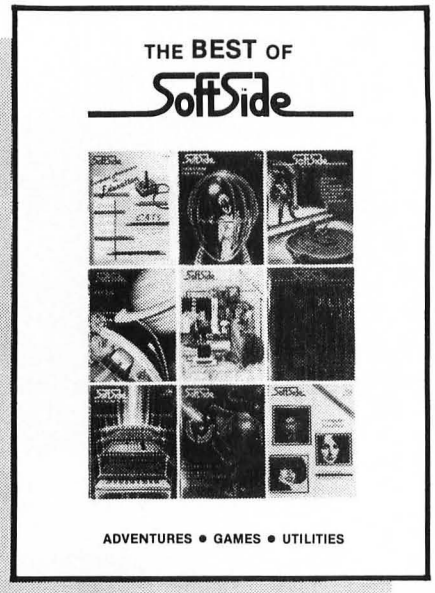
TomorrowHouse monitors heating, air conditioning and lights and includes such convenience features as an appointment calendar and voice wakeup alarm. A glance at the monitor display allows a status check of all security and energy functions. Future enhancements are planned which will allow remote monitoring and control via any Touch-Tone phone.

Tomorrowhouse can be purchased as a do-it-yourself kit or dealers can provide turnkey installation. Prices start at about $800.00.

Like SoftSide?
Wait 'til you see our Best!

The Best of SoftSide.
Now available on Disk!

For over four years, ***SoftSide*** Magazine has brought the very best in BASIC software to Apple®, Atari®, and TRS-80® owners. Now we've selected the most useful...the most entertaining...the most fun programs from our history, and put them in ***The Best of SoftSide.***



- Check out the random-access version of the *Developing Database* program. It'll let you store and retrieve enormous quantities of information quickly and easily — and it's a real bargain, too.
- Use *Microtext, SoftSide's* BASIC word processor, to simplify the task of composing letters and other documents, storing them on disk or cassette, and printing them out on paper.
- Not in a practical mood? Try to outwit the sinister aliens from the planet Zekloke in *Operation: Sabotage*, or spend a leisurely evening with "just one more" game of *Solitaire*. There's much more.

In all, each version of ***The Best of SoftSide*** offers you over 190 pages of ***SoftSide's*** best. That's a lot of typing, and even *SWAT, SoftSide's* official debugging utility, can't make typing effortless... What to do?

Fortunately, modern science has developed an almost miraculous device to virtually instantaneously type vast amounts of material into your computer's memory — the floppy disk. Now, you can order ***The Best of SoftSide*** and its Disk Version together, with the convenient coupon below. The price? — Only $68.95.

With the Disk Version of ***The Best of SoftSide***, all you do is pop the disk in the drive, load, run, enjoy! Order your copy of ***The Best of SoftSide*** Disk Version... today!

If you've already ordered ***The Best of SoftSide*** book, you can order the disks separately. Just check the appropriate box on the coupon below, and send $49 for each copy ordered.

To order ***The Best of SoftSide*** book, see page 68 of this issue.

Send orders to:

SoftSide™

SoftSide Publications, Inc.
6 South Street
Milford, NH 03055

Yes! I want to enjoy to enjoy the best games, utilities, and adventures from ***SoftSide's*** past.

Send the ☐ Apple® ☐ Atari® ☐ TRS-80® Mod. I ☐ TRS-80® Mod. III. Version

I'm enclosing $68.95 for each copy ordered ($19.95 book + $49 disk). _____ copies x $68.95 = $_____ enclosed or charged to my credit card.

☐ I've already ordered ***The Best of SoftSide*** book, and am enclosing $49 for the disks only.

Name _____

Address _____

City _____ State_____ ZIP_____

Payment by ☐ Check ☐ M.O. ☐ VISA ☐ MasterCard
(Foreign orders please include $5 postage. USA FUNDS ONLY.)

Name of Cardholder_____

MC# and Interbank#/VISA# _____

Exp. Date _____

Signature _____

I-SEE

Monitor Debug Trace

Monitor-Command Menu, ASCII & Hex Display, Edit, Zero, & Block Move

Debug-All monitor cmnds Plus Display & Edit for Registers & Flags, Step & Break Point, Full Video

Trace-All Debug Cmnds Plus Dissassemble Trace

Monitor $12 Debug $18 Trace $24

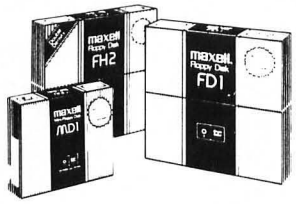
Specify Disk/Cassette & Mem.

HALLWAY SOFTWARE
6625 Morrow Dr.
Dayton, Ohio 45415
Ohio residents add 6% sales tax.

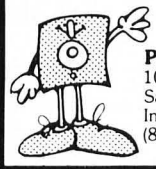
TRS 80 I—III

Maxell Floppy Disks

The Mini-Disks with maximum quality.



Dealer inquiries invited. C.O.D's accepted.
Call FREE (800) 235-4137.



PACIFIC EXCHANGES
100 Foothill Blvd.,
San Luis Obispo, CA 93401.
In Cal. call (800) 592-5935 or (805)543-1037.

ARE YOU INTO SPORTS?

DO YOU-Jog? Swim? Walk? Cycle? Run? Play Racquet Sports?

The Aerobics Master for the Apple* is for you. A full year exercise diary that computes the improvements and values of your exercises day-by-day, all year long.

$24.95 plus $1.00 postage and handling. Mich. residents add 4% sales tax.

FREE LANCE INK
1806 Wickham • Royal Oak, MI 48073
REQUIRES APPLE II* WITH: Applesoft in Rom • 48K Memory 1 or 2 Disk Drives • DOS 3.3

*Apple is a registered trademark of Apple Computer, Inc.

ELEPHANT DISKS

5SSD 5"	21.97/BX	20.97/2 OR MORE
5SDD 5"	24.97/BX	23.97/2 OR MORE

SOFTWARE 5BO I/III SPECIFY

NEWSCRIPT	DISK	99.97
GEAP/DOTPRINT 1.5	DISK	84.97
TRASHMAN	DISK	31.97
REGILIAN WORM CAS	13.97/DSK	16.97
PROWRITER 1 PARALLEL		479.97
PROWRITER 2 15"		679.97
CABLE 5BO I/III		31.97
PRINTER CARD/APPLE		99.97
HOLMES SPRINTER 5BO I/III		89.97

UPS $2.00/ORDER FREE CATALOG!!!

CASH/CHECK/MONEY ORDER

DIRECT-TO-TAPE, 14 STATION AV

HADDON HEIGHTS, NEW JERSEY 08035

APPLE PHYSICS

11 Disks - 75 Programs - $203

These programs contain extensive graphics. Each diskette has 5 to 10 programs requiring 48K memory with Applesoft.

V 1 Vectors & Graphing	$10.00
V 2 Statics	$12.00
V 3 Motion	$12.00
V 4 Conservation Laws	$12.00
V 5 Circular Motion	$15.00
V 6 Thermodynamics	$20.00
V 7 Electricity	$12.00
V 8 Optics	$20.00
V 9 Atomic Physics	$30.00
V10 Solar System Astronomy	$30.00
V11 Stellar Astronomy	$30.00

Ask for Atari IBM information

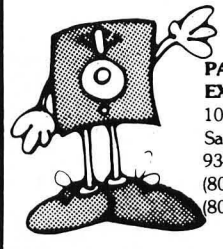
AQUARIUM: This is an aquarium simulation in which the fish swim, breed, eat and interact. The full disk includes the Community Aquarium plus 4 games. 48K $25.

DINOSAURS: 6 games and demos. Includes Dinosaur Matching, Dinosaur Hangman and Paddle Graphics. 48K $15.

CROSS EDUCATIONAL SOFTWARE
P. O. Box 1536
Ruston, LA 71270
318-255-8921
Write today for a FREE Catalog.

MEMOREX
FLEXIBLE DISCS

WE WILL NOT BE UNDER-SOLD!! Call Free **(800)235-4137** for prices and information. Dealer inquiries invited and C.O.D.'s accepted.



PACIFIC EXCHANGES
100 Foothill Blvd.
San Luis Obispo, CA 93401. In Cal. call (800)592-5935 or (805)543-1037

wabash®

When it comes to Flexible Disks, nobody does it better than Wabash.

MasterCard, Visa Accepted.
Call Free: (800) 235-4137

PACIFIC EXCHANGES
100 Foothill Blvd.
San Luis Obispo, CA 93401 (In Cal. call (805) 543-1037.)

OUT OF THE BLUE

The **SoftSide** family is growing. We are now covering the IBM Personal Computer and we cordially invite PC people everywhere to join the ranks of **SoftSide** contributors. If you have a program, a review, or just a story to tell, let us have a look at it. Send submissions to:

SoftSide Publications, Inc.
Department Z
6 South Street
Milford, NH 03055

FAMILY TREE

Features:

Not only holds parents, but will hold children, siblings & mate for any member in your tree.

Will display or print a direct 4 ply tree for any member.

Family Tree on Disc $18

HALLWAY SOFTWARE
6625 Morrow Dr.
Dayton, Ohio 45415
Ohio residents add 6% sales tax.

TRS 80 I—III

MAXLIFE™

THE ONE YOU'VE HEARD ABOUT
"GUARANTEED TO WORK"

MX-80 CARTRIDGE WITH MOBIUS LOOP

ONLY $7.99 EA

MIN. ORDER 4 CARTRIDGES
PRICE INCLUDES SHIPPING WHEN CHECK ACCOMPANIES ORDER
MASS RESIDENTS ADD 5% SALES TAX

JAN TECH

. . . WHEN YOU NEEDED IT YESTERDAY

P.O. BOX 647, RANDOLPH, MA 02368

(617) 961-4210

CHESS FILE

Features:

Names & Ratings
Tournament & Date
Round # & # Moves
Opening and and and

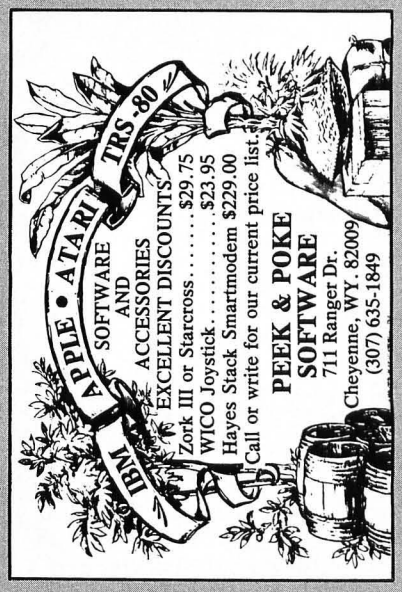
YOUR GAME

Which can be played automatically or stepped fwrd or bwrd

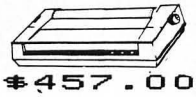
Comes loaded with Fischers World Championship Games
Chess File on Disk $24.95

HALLWAY SOFTWARE
6625 Morrow Dr.
Dayton, Ohio 45415
Ohio residents add 6% sales tax.

TRS 80 I—III



C.ITOH
PROWRITER 8510A
♠ DOT GRAPHICS
♥ 125 CHR/SEC



PARRALEL $457.00
15"PROWRITER II $689.00
CALL FOR DISCOUNT PRICES FOR INTERFACES
ATARI 800 48K COMPUTER $669.00
BMC 12" GREEN SCREEN MONITOR
16 MHz 80 chr.X 24 line Res.
$89.95
BMC #12A

BMC HIGH RES. 12"MONITOR $125.00
TSK VISION I RGB 12"COLOR $316.25
JULTRONICS
PO BOX 584 WARRINGTON PA 18976
ORDERS CALL COLLECT 215-343-4771
WE ACCEPT VISA & MASTER CARD

Continental Adventures

for ATARI® Computer Owners

Each adventure takes up to an hour and is programmed with random variables to change the game every time you play.

- **THE TALISMAN OF POWER**
A search through many obstacles for the Four Keys of Gremlock. Will you get out alive? 16K-$17.95
- **THE GHOST TOWER**
Find a Magical Gem in a tower haunted with Orcs, goblins, etc. 16K-$16.95.
- **GALACTIC ADVENTURE**
A space fantasy adventure in which the aliens have galaxial control. Your mission, as starship commander, is to destroy their starbase. 16K-$19.95.

SATISFACTION GUARANTEED

Send for Complete Software List

DEALER INQUIRIES INVITED
CONTINENTAL ADVENTURES, 4975 Brookdale
Bloomfield Hills, MI 48013 (313) 645-2140

ATARI SALE

	LIST	SALE
CENTIPEDE Rom Cartridge	44.95	34.00
CHOPLIFTER Disk	34.95	26.00
FROGGER Tape/Disk	34.95	26.00
PREPPIE Tape/Disk	29.95	22.00
CANYON CLIMBER Tape/Disk	29.95	22.00
RASTER BLASTER Disk	29.95	22.00
ZORK I/II/III Disk	39.95	30.00
DEADLINE Disk	49.95	37.00
SHOOTING ARCADE Tape/Disk	29.95	22.00
APPLE PANIC Tape/Disk	29.95	22.00
CRUSH, CRUMBLE & CHOMP T/D	29.95	22.00
PACIFIC COAST HIGHWAY T/D	29.95	22.00
MOUSKATTACK Disk	34.95	26.00

OVER 1,300 PROGRAMS!

We have a complete line of Atari programs, also IBM PC, Apple, TRS-80 I/III & TRS-80 Color Computer. Write for price list for specific computer.

SHIPPING & HANDLING: $2.00 per order.
TAX: 5% (Maryland residents only)
CREDIT CARDS: MasterCard and Visa accepted. Add 3% for service charges. Include name, address, credit card number, and bank number (MasterCard only).

FREDERICK SOFTWARE SUPPLY
P.O. BOX 1538 FREDERICK, MD 21701
(301) 662-5600

ATACOMP PRESENTS

Affordable 100% machine language arcade games for the ATARI 400/800*.

- **CANNON COUPE**
Scream down the curving speedway as you blast the rocks, cops, and UFO's: 20 game options.
- **GOLDEN MAZE**
Scour the random maze for gold as you evade the Jaw-Trogs. 1 or 2 player co-op: 6 game options.
- **BUGGIES**
Can you clear the board before the Buggies munch you?
- **THE SWARM**
Defend yourself from the invading hoards: Bonus scoring.
- **MYRIAPOD**
Zap the segmenting Myriapod as it slithers down the screen: Bonus pods and spiders.

PRICE: $14 for each 16K tape; or $11 each game when you order two or more.

ATACOMP
RR3 Box 21
Coggon, IA 52218

Two Of The Best Programs From SoftSide

Quest 1 An exciting journey into an underground maze in search of treasure and adventure. Armed only with sword and bow, you must rely on quick thinking to survive.

Just $4.95

Flip-It is an excellent implementation of the board game Othello™. Match wits with a formidable opponent: your computer.

Just $4.95

MAIL YOUR ORDER TODAY TO:

SoftSide, 6 South Street
Milford, NH 03055

FLIPPERKIT

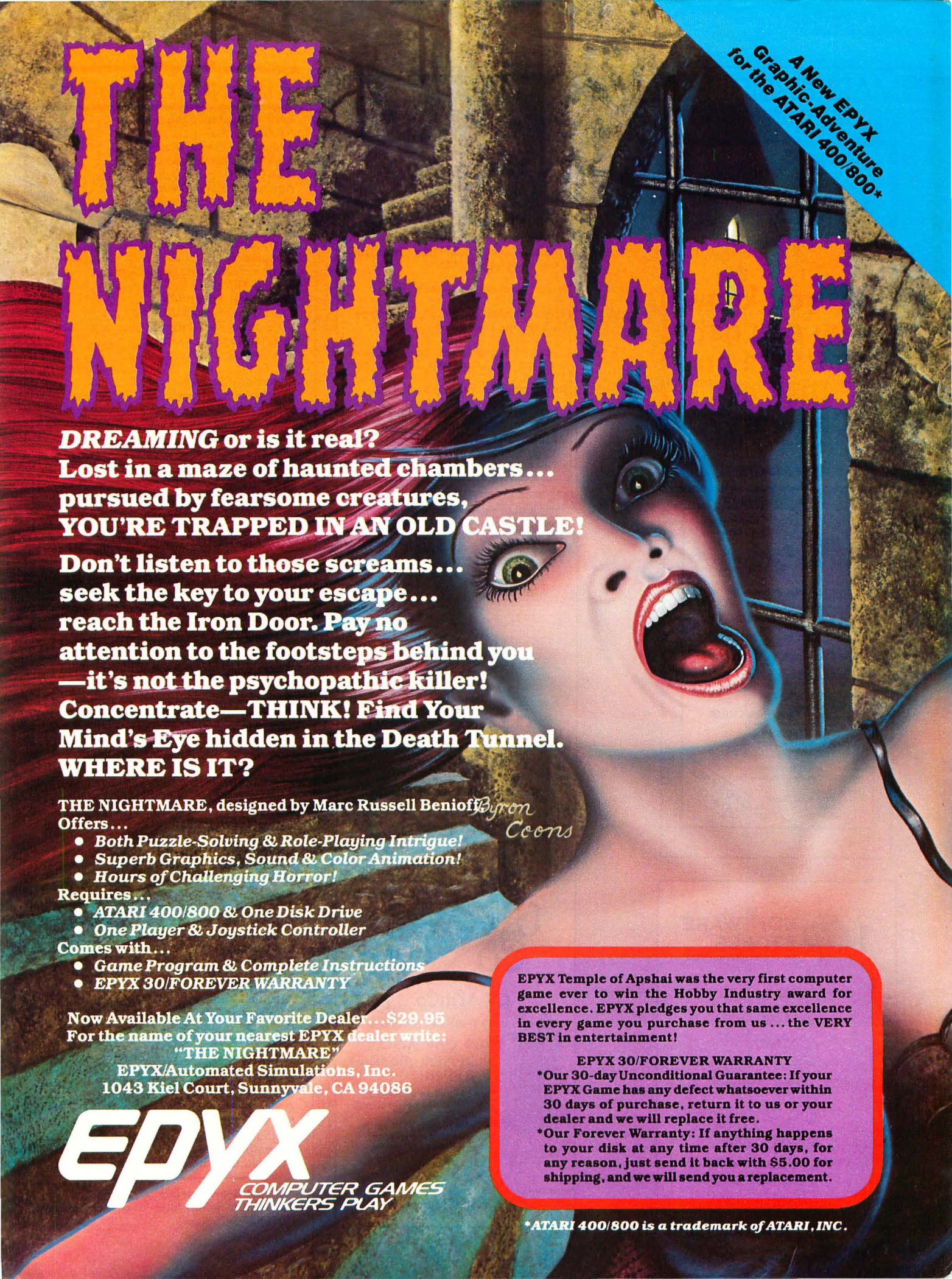
DOUBLE YOUR STORAGE CAPACITY
HALVES THE PRICE OF YOUR DISKS

The FLIPPERKIT doubles the storage capacity of all of your SSSD/SSDD 5¼" and 8" diskettes by allowing physical access to both sides. At this introductory price, FLIPPERKIT pays for itself after the first two uses. ACT NOW. Guaranteed results. Complete instructions provided. Works for all DOS and single headed disk drives.

Send your check or money order for $9.95 (CA, add $.60; CC Cty add $.65) with DISK SIZE to:

PB Industries
P.O. Box 1606
Lafayette, CA 94549

continued on page 126


THE NIGHTMARE
A New EPYX Graphic-Adventure for the ATARI 400/800*
DREAMING or is it real?
Lost in a maze of haunted chambers...
pursued by fearsome creatures,
YOU'RE TRAPPED IN AN OLD CASTLE!
Don't listen to those screams...
seek the key to your escape...
reach the Iron Door. Pay no
attention to the footsteps behind you
—it's not the psychopathic killer!
Concentrate—THINK! Find Your
Mind's Eye hidden in the Death Tunnel.
WHERE IS IT?
THE NIGHTMARE, designed by Marc Russell Benioff
Byron Coons
Offers...
• Both Puzzle-Solving & Role-Playing Intrigue!
• Superb Graphics, Sound & Color Animation!
• Hours of Challenging Horror!
Requires...
• ATARI 400/800 & One Disk Drive
• One Player & Joystick Controller
Comes with...
• Game Program & Complete Instructions
• EPYX 30/FOREVER WARRANTY
Now Available At Your Favorite Dealer...$29.95
For the name of your nearest EPYX dealer write:
"THE NIGHTMARE"
EPYX/Automated Simulations, Inc.
1043 Kiel Court, Sunnyvale, CA 94086
EPYX
COMPUTER GAMES THINKERS PLAY
EPYX Temple of Apshai was the very first computer game ever to win the Hobby Industry award for excellence. EPYX pledges you that same excellence in every game you purchase from us ... the VERY BEST in entertainment!
EPYX 30/FOREVER WARRANTY
*Our 30-day Unconditional Guarantee: If your EPYX Game has any defect whatsoever within 30 days of purchase, return it to us or your dealer and we will replace it free.
*Our Forever Warranty: If anything happens to your disk at any time after 30 days, for any reason, just send it back with $5.00 for shipping, and we will send you a replacement.
*ATARI 400/800 is a trademark of ATARI, INC.

Market/Side *continued*

Did you know you can get SoftSide on Disk or Cassette?

See page 25.

QUIT GUESSING

Use your APPLE II to become a

SUCCESSFUL HANDICAPPER

Accurately project logical contenders with this amazing new rating system

Menu driven for easy use

Input readily available from Daily Racing Form

Complete instructions allows anyone to handicap like a pro

Handicapper on disk $24.00

MARQUESS DISTRIBUTING
8340 WOODLAND RD
MILLERSVILLE, MD. 21108

Requires 48K, 1 disk

SPECIAL SUPER SALE!! PROGRAMS FOR TRS-80 APPLE ATARI VIC-20

Special with this Ad

TRS-80 —
VAULTS OF CYMARRON
(Graphic Adventure)
RETAIL $39.95 NOW $20.00

APPLE — VISI PAK
(Business)
RETAIL $700.00 NOW $550.00

ATARI — LISP 2.0
(Language)
RETAIL $149.95 NOW $110.00

VIC-20 — VIC SYNTH
(Music Synthesiser)
RETAIL $39.95 NOW $20.00

Add $1.50 Shipping
Calif. Res. Add Sales Tax

Oasis West
470 Castro #207-3359C
S.F. CA 94114
(415) 861-8966

SEND FOR FREE CATALOG!!

Hartley
SOFTWARE SHOP

Contact your local Hartley dealer or send for **FREE** catalog.

Hartley Courseware, Inc.

Box 431
Dimondale, MI 48821
616-942-8987

1 STOP SOFTWARE STORES
for the ATARI 400/800*

We've Got Software

Free Catalog Available

ATARI is a registered trademark of ATARI, INC.

P.O. BOX 21828
DENVER, COLORADO 80221
"Your Discount Software Store"
Orderline: (303) 431-6598

THE WISH CO.

GRAPHICS HAT

USES TI-99/4A EXTENDED BASIC

Wave your Wand & Sprites Magically Appear

Two Programs for creating symbols, figures, creatures, shapes.....
make Libraries of Sprites
uses simple joystick and one key commands

Information Only $2.00
Both Programs Only $35.95

Steven M. Ruhl
THE WISH CO. PH. 588-7705
11818 TERRY LK. RD.
TACOMA, WA 98498

Send Orders To:
P. O. Box 98422 Tacoma, WA 98498

Advertiser's Index

***SOFTSIDE* PUBLICATIONS**

National Representative
Christopher Smith, Inter-Marketing Associates
(603) 827-3976

DAISY WHEEL PRINTERS...

- SMITH CORONA TPI
- BROTHER HR-1
- DAISYWRITER 2000
- DIABLO 620/630 KSR
- QUME SPRINT 9 KSR

COMPUTER SYSTEMS...

- ATARI 800
- TRS-80 MODEL III

$$$ CALL FOR PRICE $$$

You won't believe it!!!

Rainbow

P & P CORPORATION

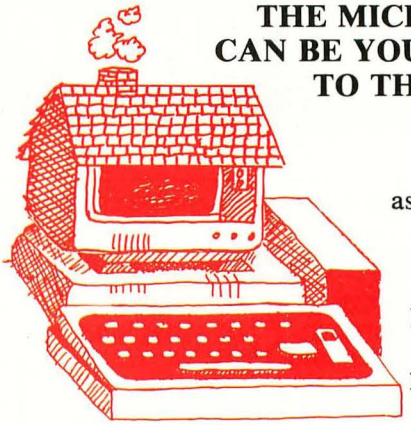
PO BOX 362 • HADDONFIELD, NJ 08033

✓ 264

800-257-6170 *in NJ call* **609-428-3900**

In *SoftSide* 38,

TELECOMMUNICATIONS



THE MICROCOMPUTER CAN BE YOUR CONNECTION TO THE WORLD.

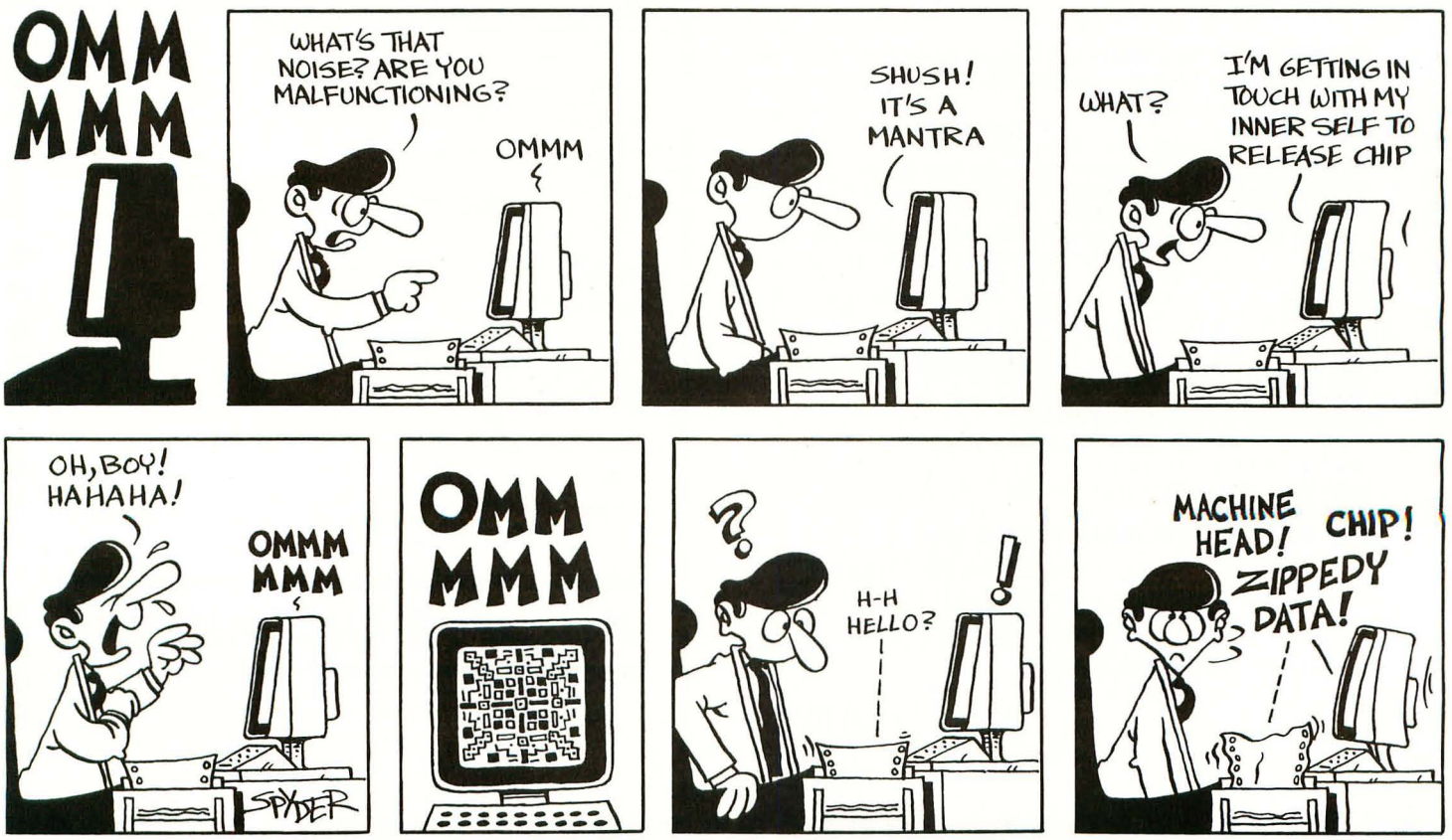
Some experts are predicting that as many as 50% of the American public will be working from their homes by the year 2000. Katherine Ackerman is doing it now and will provide an overview of **TELECOMMUTING.**

Have you ever imagined playing your favorite computer games with competitors from all over the world? It's now possible with the many on-line gaming systems you can access from your computer and modem. Roe Adams will report on his experiences with several of these systems.

PLUS — Lance Micklus on Computer Bulletin Board Systems, the first installment of *The World Connection,* our new column on telecommunications, and *Budget,* the conclusion of *Deluxe Personal Finance.*

MACHINE HEAD

BY SPYDER



ANNOUNCING THE NEW ACCESS UNLIMITED MICRO SHOPPING CENTER

OVER $150,000 IN NEW GAMES & BUSINESS SOFTWARE

WE SUPPORT "TI", "ATARI", "COMMODORE", "IBM", "APPLE", "PERCOM", "WANG", "iBEX", "TRS-80", "EPSON", "OSBORNE", "SANYO", "COLUMBIA", "CHAMELEON".

LESS THAN DISTRIBUTOR COST—BEST BUYS IN U.S.
MANUFACTURER OVER-PRODUCTION

"iBEX" 64K Business System/Word Processor Model 2200

- 9" Hi Resolution 80 x 24 green phosphor green monitor with smoky grey antiglare mask. (Comes with system, no extra charge). (Can be Software switched to 40 characters wide.)
- Dual 5¼" Floppy Disk Drives—That's right; 2 drives at no extra charge
- Full Keyboard with Number Pad & programmable Functions Keys (it's detached & movable at no extra charge)
- Z 80 Micro Processor (Standard)
- Full Internal Memory (Standard) (Nothing to buy additional)
- Centronics Compatible Interface (Parallel) • Serial Interface—(No Extra Charge)
- Format is CP/M Compatible & supports the IBM 3740 (switchable) Industry standard—Data exchange with most other computers is possible. Entire Computer System Reg. Retail: $4,067.00

"CP/M" Operating System **"M/Basic"** Interpreter—Reg. Retail: $550.00
"Perfect Writer" 2nd Generation Word Processing software—Easier to use.
"Perfect Calc" (with 17 different applications programs — Plus split screen spread sheet and "What if" calculations) Includes such programs as: "Stocks & Bonds, Cash Flow & Income Statements, Payroll & Personal Income Tax, and Budgets.
"Perfect Filer" (All 3 fully integrated) All ready for your use.
Reg. Retail: $1,265.00 Total Retail: $5,883.00

YOUR COST: **$1,395.00** Plus Tax (F.O.B. TX or MA)
(Complete System Less Printer)
You Save: $4,688.00

iBEX Model 7202 Word Processing Medium Size Business System same features as the Model 1 2200 PLUS:

12" Hi Resolution Green Phosphor Screen
Dual 8" 1.2 Megabyte drives—For a Total of 2.4 Megabytes External Storage.
Clock Timer W/Calendar and Battery: Reg. Retail: $9,067.00
***CP/M** Operating System ***M/Basic** Interpreter—Reg. Retail: $550.00
****Perfect Writer** } 2nd Generation Word Processing Software — Easier To Use and
Perfect Speller } Better than WordStar!
Perfect Calc (with 17 different applications programs—Plus split screen spread Sheet and "What If" calculations) Includes such programs as: Stocks & Bonds, Cash Flow & Income Statements, Payroll & Personal Income Tax, and Budgets.
Perfect Filer (all 4 fully integrated) All ready for your use! Reg. Retail: $1,666.00
**ADS—Vertical Small Business Accounting Package: $779.00

TOTAL RETAIL: $13,078.00 YOUR COST: **$3,995.00** (F.O.B. TX or MA)
YOU SAVE: $8,083.00

NOTHING ELSE TO BUY BUT A PRINTER

Your Choice of R/O Printers Dot Matrix from $345.00 to $1,400.00
Letter Quality from $645.00 to $1,900.00
OTHER MODEL iBEX COMPUTERS ALSO AVAILABLE AT SPECIAL CLOSE-OUT PRICES.

NEW! 5¼" SOFT SECTOR DISKETTES W/Hub Ring

"BASF" 5¼" Lifetime Limited Warranty **$24.90** SSDD Box of 10

ATHANA 1 Yr. Limited Warranty **$19.90** SSSD Box of 10

8" Diskettes From $22.75

OTHER DISKETTES AT EQUAL SAVINGS

Flip 'N File Boxes—Holds 50 Diskettes		
5¼"	Reg. $34.95	**Sale $19.95**
8"	Reg. $49.95	**Sale $28.95**
5¼" Access File Box—Holds 75 Diskettes		
	Reg. $49.95	**Sale $24.95**

LIBRARY CASES—Holds 10 Diskettes. Sale **$2.65** ea.

"IBM PC" COMPUTERS IN STOCK
Bare Drives (without Controller) For "IBM PC"—Internal or External—**$269.95 ea.**

OVER $1,000,000 IN NEW COMPUTERS, PERIPHERALS AND SUPPLIES! NEW PRINTERS IN STOCK: "CENTRONICS", "C'ITOH", "OKIDATA", "STAR", "TALLY", "RICOH", "SMITH-CORONA TP", "EPSON", "GEMINI". VALUES TO $3500.00. SALE: $345 to $1995

Call for pricing and fast delivery on IBM, OSBORNE, TRS 80II* & III*, "EPSON Hx20", "SANYO", "COLUMBIA", and "CHAMELEON". **1-800/527-3475**
Top 100 Software programs normally maintained in stock and sold at our special Access Unlimited Micro Shopping Center prices.
" " Reg. Trademarks

☐ Please send me a FREE catalog. I'm not ready to order at this time.
☐ YES, I'm taking advantage of your Sales prices.

Name _____
Company Name _____
Address _____
City _____ State _____ Zip _____
Phone Number ( ) _____

Quantity	Item	Unit Price	Subtotal
		Subtotal	
		State Sales Tax (Texas residents only)	
		Total	

Check one:
☐ payment enclosed ☐ Visa ☐ MasterCard*

*If MasterCard, numbers above name:

Expiration Date: ☐☐ — ☐☐

Authorized signature, if charged

ACCESS UNLIMITED

DEPT. D2/401 N. Central Expwy. #600/Richardson, Texas 75080
Tel. 1-800/527-3475 214/340-5366
214/690-0207 Sat. and Evenings Only

Compatible color . . . monitors

. . . for demanding graphics or text display.

For high resolution (560H X 240V) color graphics, you can't beat the Amdek Color-II Monitor. And if you're looking for economy, the Color-III Monitor with 260H X 300V resolution is a superb buy.

Both monitors feature RGB video input for computer controlled color . . . and Amdek's easy-to-install Digital Video Multiplexor (DVM) board permits interface with the most popular 80 column boards.

Just call, or write for full details.

- Color-II Monitor has RGB input and 560(H) X 240(V) line resolution.
- Color-III Monitor has RGB input and 260(H) X 300(V) line Resolution.
- Digital Video Multiplexor (DVM) assures color graphics interface with most popular 80 column boards, such as: Videx "Videoterm", Advanced Logic "Smart-term", M&R "SUP 'R' TERMINAL", Bit-3 Computer Corporation "FULL VIEW-80", and the "Doublevision" boards.

2201 Lively Blvd. • Elk Grove Village, IL 60007
(312) 364-1180 TLX: 25-4786

AMDEK CORP.

Amdek . . . your guide to innovative computing!

SINGLES' NIGHT AT MOLLY'S

By Norman J. Wazaney Jr.

You never dreamed Solitaire could be so fascinating.

Solitaire in a saloon can be fun but it's better on your Apple*. Fair warning: if you get hooked on Solitaire, beware of this game! "Singles' Night at Molly's" is actually two basic solitaire card games with several variations permitting you to use the skill level and strategy you enjoy most. Play alone or against other players, where a rating system declares the winner. Features High Resolution color graphics, full user documentation and various scoring potentials.

There are hours, days, even years of pleasure to be derived from this intriguing game. Available now for only $29.95 at computer stores.

To order by phone, call 800-526-9042 and use your Visa or MasterCard. All shipments made the same day in which orders are received. To order by mail, add $1 for shipping charges and send your check to:
Soft Images, 200 Route 17, Mahwah, NJ 07430.

"Singles' Night" runs on 48K Apple II, Applesoft in ROM.
3.3 DOS/One Disk Drive
*Apple is a registered trademark of Apple Computer Co.

Soft images
DIVISION OF DECISION SYSTEMS INC