

Hardcore

# COMPUTIST

Issue No. 18

\$2.50

Softkey  
For The  
Scholastic Version  
of Bankstreet  
Writer  
Pg. 9

Softkeying  
Applewriter //e  
Pg. 12

Checksoft v2  
Pg. 22

RKCopy:  
Backing-up SSI's  
Non-RDOS Disks  
Pg. 27



The Games of 1984:  
In Review  
Pg. 16

Hardcore COMPUTIST  
PO Box 110846-T  
Tacoma, WA 98411

BULK RATE  
U.S. Postage  
**PAID**  
Tacoma, WA  
Permit No. 269

# TURBO CHARGE YOUR IIe AND EXPAND APPLE WORKS TO 101K

## MEMORYMASTER IIe™ 128K RAM CARD

As an Apple user, you already know all the things your IIe can do. Now Applied Engineering expands that list with MemoryMaster IIe™.

With the MemoryMaster IIe, you'll have up to 192K RAM at your command. Designed expressly for the auxiliary slot in the IIe, the MemoryMaster IIe provides an 80 column video display and up to 128K of memory using just one slot. But the MemoryMaster IIe differs from other large memory cards in one important way, both the 64K and 128K configurations are TOTALLY compatible with software written for the Apple IIe 80 column and extended 80 column cards. In fact, it's not possible to write a program that uses the smaller cards from Apple that won't work with the MemoryMaster IIe.

But the MemoryMaster IIe is not just another piece of unsupported

hardware. Each MemoryMaster IIe includes a multi-programming environment program which will enable you to have three different programs ready to run at any moment.

Many of today's software packages for data-base management, word processing, business applications and spread sheets either require or are enhanced by the MemoryMaster IIe. And more and more programs are using double high resolution graphics for which the MemoryMaster IIe is required.

If you already have Apple's 64K card, just order the MemoryMaster IIe with 64K and use the 64K from your old board to give you a full 128K. The board is fully socketed so you simply plug in your chips.

- Expands your Apple IIe to 192K memory.
- Provides an 80 column text display.
- TOTALLY compatible with all Apple IIe 80 column and extended 80 column card software—there are NO exceptions.
- Available in 64K and 128K configurations.
- The 64K configuration is USER upgradeable to 128K.
- Can be used as a solid state disk drive to make your programs run over 20 times faster (the 64K configuration will act as half a drive).
- Bank select LED's for each 64K bank.
- Permits your IIe to use double high resolution graphics.
- Uses the same commands as the Apple 80 column card.
- Plugs into the auxiliary slot in the Apple IIe.
- The 64K MemoryMaster IIe will automatically expand VisiCalc to 95K storage, in 80 columns! The 128K MemoryMaster IIe will expand VisiCalc to 141K storage with optional pre-boot disk.
- The 64K MemoryMaster IIe will automatically expand Apple Works to 55K storage, in 80 columns! The 128K MemoryMaster IIe will expand Apple Works to 101K storage with optional expand disk.
- Fully PASCAL and CP/M compatible.
- Lowest power consuming 128K card available.
- Complete documentation included.
- PRO-DOS will automatically use the MemoryMaster IIe as a high speed disk drive.
- Three year warranty.

### Low Cost Options.

#### Apple Works Expand™

Although Apple Works is compatible with both a 64K and 128K MemoryMaster IIe, Apple Works only "sees" its first 64K bank. Our Apple Works expand program will make a modification to Apple Works that simply lets it know you've got more memory, giving you 101K work space.

**\$29**

#### Ram Drive IIe™

Sure-fire wait-reduction! Time was, you had to wait for your disk drives. But Ram Drive IIe has changed all that, giving you a large, extremely fast, solid-state disk drive, much faster than floppies or hard disks.

Ram Drive IIe works with either the 64K or 128K MemoryMaster IIe to give you a high speed solid-state disk drive. The Ram Drive IIe software features audio-visual access indicators, easy setup for turnkey operation, and easy menu driven documentation. The program can be modified and is copyable. If you have a 64K MemoryMaster, Ram Drive IIe will act as half a disk drive. If you have a 128K MemoryMaster, Ram Drive IIe will act as a full disk drive. Ram Drive IIe is compatible with APPLESOFT, DOS3.3, PRO-DOS, and PASCAL. Disk also includes a high speed RAM disk copying program. Ram Drive is another disk drive only 20 times faster.

**\$29**

#### CP/M Ram Drive IIe

CP/M Ram Drive IIe is just like the Ram Drive IIe above, only for CP/M. CP/M Ram Drive IIe runs on any Z-80 card that runs standard CP/M i.e. Applied Engineering Z-80 Plus or Microsoft Soft Card. CP/M Ram Drive will dramatically speed up the operation of most CP/M software because CP/M normally goes to disk fairly often. Fast acting software like dBase II and Wordstar become virtually instantaneous when used with CP/M Ram Drive.

**\$29**

#### VC IIe Expander

VC IIe Expander will give owners of VisiCalc IIe and the 128K MemoryMaster a total of 141K work space. This disk will pre-boot VC IIe in 1.5 seconds.

**\$29**

#### Advanced VC IIe Expander

Advanced VC IIe Expander is just like VC IIe Expand only it is designed for Advanced VC IIe. Your work space will be increased to 131K. This disk will pre-boot in 1.5 seconds.

**\$29**

MEMORYMASTER IIe with 128K .....	\$249
Upgradeable MEMORYMASTER IIe with 64K .....	\$169
A. E. Extended 80 Column Card with 64K .....	\$139

Texas Residents Add 5% Sales Tax  
Add \$10.00 If Outside U.S.A.

Send Check or Money Order to:  
APPLIED ENGINEERING  
P.O. Box 798  
Carrollton, TX 75006

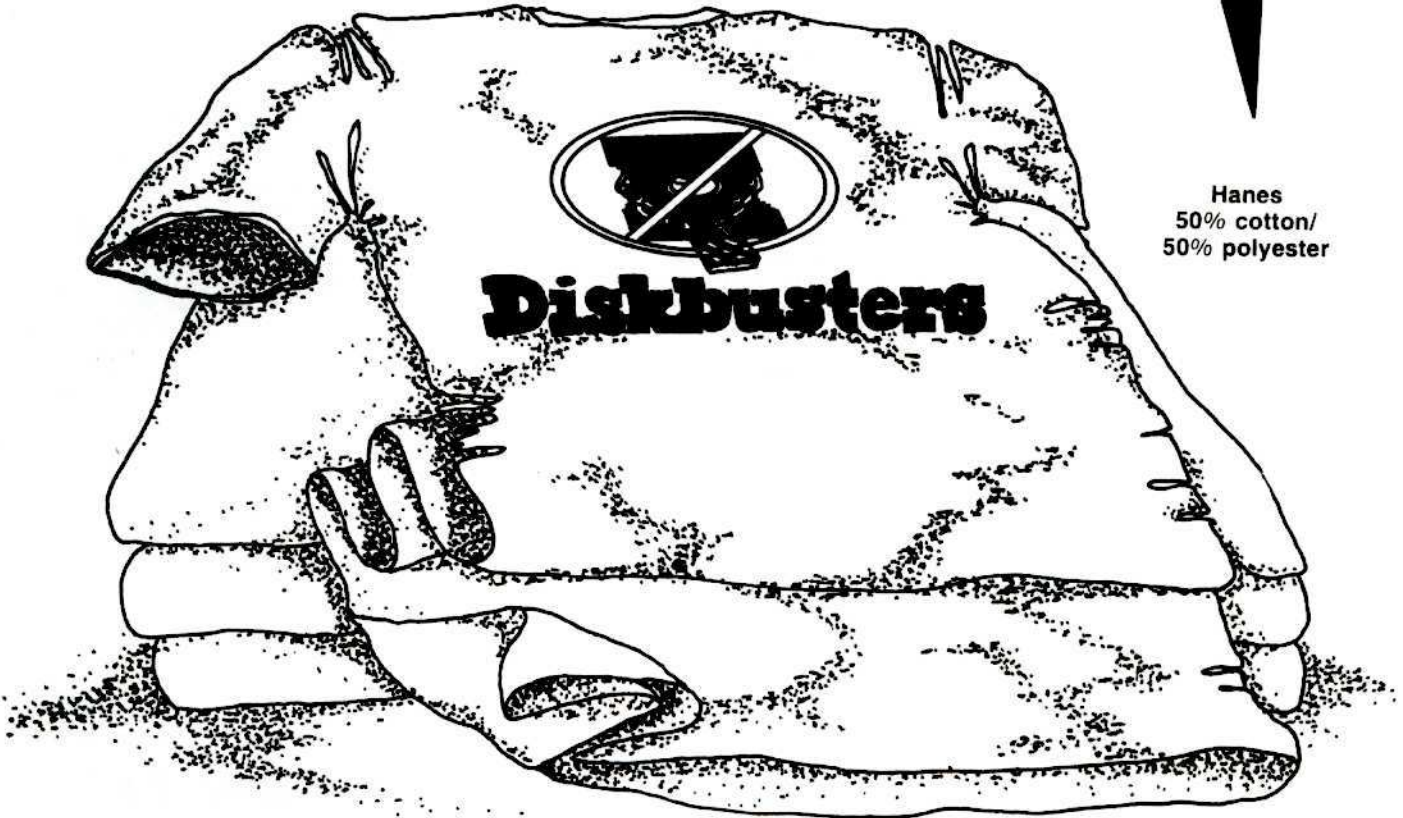
Call (214) 492-2027  
8 a.m. to 11 p.m. 7 days a week  
MasterCard, Visa & C.O.D. Welcome  
No extra charge for credit cards.

Spring  
is  
here!

\$9<sup>only</sup>  
-95

# That means T-SHIRT weather!

What better way to show your friends  
*just* what you think of software copy-protection  
than to wear the ORIGINAL Hardcore COMPUTIST  
DISKBUSTERS t-shirt?!!!



Hanes  
50% cotton/  
50% polyester

Hardcore  
COMPUTIST  
says:

**"We ain't afraid  
of no disk!"**

**YOU  
CAN,  
TOO!**

Rush me \_\_\_\_\_ (total) DISKBUSTERS T-shirts in the sizes indicated  
below. I have enclosed \$9.95 (plus applicable tax & shipping) for each shirt.

ADULT MENS:     Small     Med.     Large     X-Large

Name \_\_\_\_\_

Address \_\_\_\_\_

City \_\_\_\_\_ St \_\_\_\_\_ Zip \_\_\_\_\_

Phone \_\_\_\_\_

VISA/MC \_\_\_\_\_ - \_\_\_\_\_ - \_\_\_\_\_ Exp \_\_\_\_\_

Signature \_\_\_\_\_

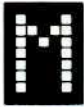
**Send check or money order to:** Hardcore T-shirts, PO Box 110816, Tacoma, WA 98411.  
Washington residents add 7.8% sales tax. Foreign orders add 20% shipping and handling. US funds  
drawn on US bank.

Use your  
VISA/MC,  
personal check  
or money order.

Shirts available  
in adult sizes  
**ONLY!**  
(black & red  
on gray).

**Hardcore  
COMPUTIST**

PO Box 110816 Tacoma, WA 98411



any of the articles published in Hardcore COMPUTIST detail the removal of copy protection schemes from commercial disks or contain information on copy protection and backup methods in general. We also print bit copy parameters, tips for adventure games, advanced playing techniques (APT's) for arcade game fanatics and any other information which may be of use to the serious Apple user.

Hardcore COMPUTIST also contains a center CORE section which focuses on information not directly related to copy-protection. Topics may include, but are not limited to, tutorials, hardware/software product reviews and application and utility programs.

**What Is a Softkey Anyway?** Softkey is a term which we coined to describe a procedure that removes, or at least circumvents, copy protection that may be present on a disk. Once a softkey procedure has been performed, the disk can usually be duplicated by the use of Apple's COPYA program (on the DOS 3.3 System Master Disk).

**Commands and Controls:** In any softkey procedure, the actual keystroke commands which a reader is required to perform are set apart from normal text (typed in bold and indented). An example is:

**PR#6**

Follow this with the RETURN key. The RETURN key must be pressed at the end of every such command unless otherwise specified.

Control characters and shifted characters are indicated in commands as small superscripts.

6<sup>CTRL</sup>P

To complete this command, you must first type the number 6, and then place one finger on the CTRL key and one finger on the P key. Shifted characters have a small SHIFT before them.

**Requirements:** Most of the programs and softkeys which appear in Hardcore COMPUTIST require one of the Apple II series of computers and at least one disk drive with DOS 3.3. Occasionally, some programs and procedures have special requirements: a sector editing program or a "nonautostart" F8 monitor ROM. The prerequisites for deprotection techniques or programs will always be listed at the beginning article under the "Requirements:" heading.

**Software Recommendations:** Although not absolutely necessary, the following categories of utilities are recommended for readers who wish to obtain the most benefit from our articles:

- 1) **Applesoft Program Editor** such as Global Program Line Editor (GPLE).
- 2) **Disk Editor** such as DiskEdit, ZAP from Bag of Tricks or Tricky Dick from The CIA.
- 3) **Disk Search Utility** such as The Inspector, The Tracer from The CIA or The CORE Disk Searcher.
- 4) **Assembler** such as the S-C Macro Assembler or Merlin/Big Mac.
- 5) **Bit Copy Program** such as COPY II+, Locksmith or The Essential Data Duplicator.
- 6) **Text Editor** capable of producing normal sequential text files such as Applewriter II, Magic Window II or Screenwriter II.

You will also find COPYA, FID and MUFFIN on the DOS 3.3 System Master Disk useful.

**Hardware Recommendations:** Many softkey procedures require that the user be able to enter the Apple's system monitor during the execution of a copy-protected program. Check the following list to see what you will need:

**Apple II Plus / Apple IIe / Apple compatibles:** 1) Place an Integer BASIC ROM card in one of the Apple's slots. 2) Use a non-maskable interrupt (NMI) card such as Replay or Wilcard.

**Apple II Plus / Apple compatibles:** 1) Install an F8 ROM with a modified RESET vector on the computer's motherboard. This method was detailed in Ernie Young's article, "Modified ROMS" (Hardcore COMPUTIST No. 6).

**Apple IIe / Apple IIc:** 1) Install a modified CD ROM on the computer's motherboard. Don Lancaster's company (Synergistics, Box 809-AP, Thatcher, AZ

85552) sells the instructions necessary to make this modification but access to an EPROM burner is also required. Making this modification to an Apple IIc will void its warranty, but gaining the ability to RESET into the monitor at will greatly enhances the capacity of the Apple owner to remove a disk's copy protection.

A 16K or larger RAM card, a printer, and a second disk drive are also recommended for Apple II or II+ owners.

**Recommended Literature:** The Apple II and II+ came bundled with an Apple Reference Manual, however this book is not included with the purchase of an Apple IIe. You'll find that this book is necessary reference material. A DOS 3.3 manual is also recommended.

Other helpful books include: *Beneath Apple DOS*, Don Worth and Peter Lechner, Quality Software, \$19.95; *Assembly Language For The Applesoft Programmer* Roy Meyers and C.W. Finley, Addison Wesley, \$16.95; and *What's Where In The Apple*, William Lubert, Micro Ink, \$24.95.

**Keying in Applesoft Programs:** BASIC programs are printed in Hardcore COMPUTIST in a format that is designed to minimize errors for readers who key in these programs. To understand this format, you must first understand the formatted LIST feature of Applesoft.

If you strike these keys:

**10 HOME:REMCLEAR SCREEN**

a program will be stored in the computer's memory. Strangely, this program will *not* have a LIST that is exactly as you typed it. Instead, the LIST will look like this:

**10 HOME : REM CLEAR SCREEN**

Programs don't usually LIST the same as they were keyed in because Applesoft inserts spaces into a program listing before and after every command word or mathematical operator. These spaces usually don't pose a problem except in line numbers which contain REM or DATA command words. The space inserted after these command words can be misleading. For example, if you want a program to have a list like this:

**10 DATA 67,45,54,52**

you would have to omit the space directly after the DATA command word. If you were to key in the space directly after the DATA command word, the LIST of this program would look like this:

**10 DATA 67,45,54,52**

This LIST is different from the LIST you wanted. The number of spaces you key after DATA and REM command words is very important.

All of this brings us to the Hardcore COMPUTIST LISTING format. In a BASIC LISTING, a space that *must* be keyed in is printed as a delta character ( $\delta$ ). All other spaces were put in by Applesoft and it doesn't matter whether you key them in or not.

There is one exception: If you want your checksums (see "Computing Checksums" section) to match up, you must not key in any spaces after a DATA command word unless they are marked by delta characters.

**Keying In Hexdumps:** Machine language programs are printed in Hardcore COMPUTIST as both source code and hexdumps. Only one of these formats need be keyed in to get a machine language program. Hexdumps are the shortest and easiest format to type in. First, you must enter the monitor:

**CALL -151**

Now key in the hexdump exactly as it appears in the magazine ignoring the four digit checksum at the end of each line (a "S" and four digits). If you type something incorrectly, a beep will alert you to retype that line.

When finished, return to BASIC with a:

**E003G**

Remember to BSAVE the program with the correct filename, address and length parameters as given in the article.

**Keying In Source Code:** The Source Code portion of a machine language program is provided only to better explain the program's operation. If you wish to key it in anyway, you will need an assembler. The S-C Assembler is used to generate all source code printed in Hardcore COMPUTIST. Without this assembler, you will have to translate pieces of the source code into something *your* assembler will understand. A table of S-C Assembler directives just for this purpose is printed on page 26 of this magazine. To complete the translation process, you need to understand the directives of your assembler and convert the directives used in the source code listing to directives similar to those used by your assembler.

**Computing Checksums:** Checksums are four digit hexadecimal numbers which verify whether or not you keyed in the program exactly as it was printed. There are two types of checksums: one created by the CHECKBIN program (for machine language programs) and the other created by the CHECKSOFT program (for BASIC programs). If the checksums these programs create on your computer match the checksums accompanying the article, then you keyed in the program correctly. If not, the program is incorrect at the line where the first checksum differs.

1) To compute CHECKSOFT checksums:

**LOAD filename  
BRUN CHECKSOFT**

Get the checksums with

**&**

And correct the program where the checksums differ.

2) To compute CHECKBIN checksums:

**CALL -151  
BLOAD filename**

Load the CHECKBIN program at an out of the way place and hook it up

**BRUN CHECKBIN, AS6000**

Then type the starting address, a period and ending address of the file followed by a CTRL-Y.

**XXXX.XXXX<sup>CTRL</sup>Y**

And correct the lines at which checksums differ.

## How-To's Of Hardcore

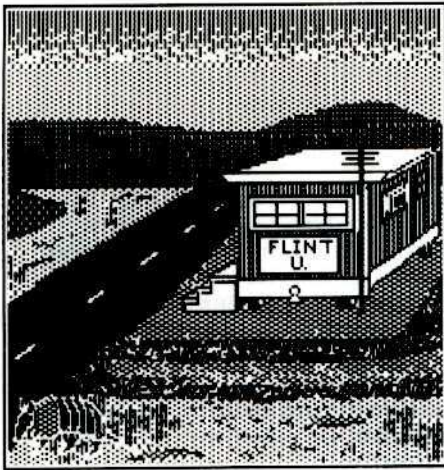
Welcome to Hardcore COMPUTIST, a publication devoted to the serious user of Apple II and Apple II compatible computers. Our magazine contains information you are not likely to find in any of the other major journals dedicated to the Apple market.

Our editorial policy is that we do NOT condone software piracy, but we do believe that honest users are entitled to backup commercial disks they have purchased. In addition to the security of a backup disk, the removal of copy protection gives the user the option of modifying application programs to meet his or her needs.

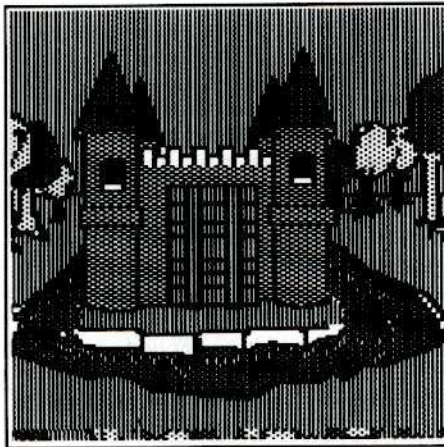
New readers are advised to read this page carefully to avoid making errors when following the softkeys or typing in the programs printed in this issue.

# Hardcore COMPUTIST

Publisher/Editor: Charles R. Haight Technical Editors: Gary Peterson, Ray Darrah  
 Production & Graphics: Lynn Campos-Johnson Circulation: Michelle Frank  
 Business Manager: Valerie Robinson Advertising: (206) 474-5750 Printing: Grange Printing, Inc., Seattle, WA  
 Hardcore COMPUTIST is published monthly, except December, by SoftKey Publishing, 5233 S. Washington, Tacoma, WA 98409  
 Phone: (206) 474-5750



Pg. 16



Pg. 16

This month's cover: A hi-res graphic dump from the popular new parlor game, *Murder By The Dozen*, CBS Software.

Address all advertising inquiries to Hardcore COMPUTIST, Advertising Department, PO Box 110816, Tacoma, WA 98411. Mail manuscripts or requests for Writers Guides to Hardcore COMPUTIST, PO Box 110846-K, Tacoma, WA 98411.

Return postage must accompany all manuscripts, drawings, photos, disks, or tapes if they are to be returned. Unsolicited manuscripts will be returned only if adequate return postage is included.

Entire contents copyright 1985 by SoftKey Publishing. All rights reserved. Copying done for other than personal or internal reference (without express written permission from the publisher) is prohibited.

The editorial staff assumes no liability or responsibility for the products advertised in the magazine. Any opinions expressed by the authors are not necessarily those of Hardcore COMPUTIST magazine or SoftKey Publishing.

Apple usually refers to the Apple II or II Plus Computer, and is a trademark of Apple Computers, Inc.

**SUBSCRIPTIONS:** Rates: U.S. \$25.00 for 12 issues, Canada \$34.00, Mexico \$39.00, Foreign (airmail) \$60.00, Foreign (surface mail) \$40.00. Direct inquiries to: Hardcore COMPUTIST, Subscription Department, PO Box 110846-T, Tacoma, WA 98411. Please include address label with correspondence.

**DOMESTIC DEALER RATES:** Call (206) 474-5750 for more information.

**Change Of Address:** Please allow 4 weeks for change of address to take effect. On postal form 3576 supply your new address and your most recent address label. Issues missed due to non-receipt of change of address may be acquired at the regular back issue rate.

## 9 Softkey For The Scholastic Version of Bank Street Writer

No doubt you've been having trouble trying to deprotect your Scholastic version of the Bank Street Writer with the method outlined in Hardcore COMPUTIST No. 10 (for the Broderbund version, only). Instead, follow this new method specifically designed for the *other* version of this popular word processor.  
 By James Daniels.

## 12 Softkeying Applewriter //e

You won't have to rely on a single protected copy of your Applewriter //e word processor any longer. With this softkey, you'll be able to make all the backups you need. By Peter Edelsten.

## 13 Installing A Free Sector Patch Into Applewriter //e

Using your newly deprotected Applewriter //e disk, you can begin to take advantage of a world of enhancements now available. We'll provide the first one: install the patch described in this article and you'll effectively eliminate the need to guesstimate the amount of free space remaining on a disk.  
 By Bob Bragner.

## 16 The Games of 1984: In Review

Looking for entertainment value? Before you spend any more of your hard-earned dollars on computer games that *aren't quite* what you wanted, look through Hardcore COMPUTIST's, Games of 1984: In Review. You'll find critiques for several of the most recently released top-of-the-line games for the Apple computer. This month we review adventure games; next month, arcade games. By Jeff Hurlburt.

## 19 65C02 Chips Now Available

For those readers who expressed an interest in purchasing one of the chips, this question and answer session provides answers to many of the most commonly asked questions concerning the 65C02 microprocessor.

## 22 Checksoft v2

Still using the original Checksoft program? No more! Here comes Checksoft v2, an upgrade of the slow and tedious program which, in the past, helped you locate the bugs in your typed-in copy of Hardcore COMPUTIST programs. Now, the seemingly untraceable errors at the beginning will no longer ruin your efforts to check the remaining lines of your program. By Ray Darrah.

### CORE SECTION

## 25 Simple Copy Protection

Discover how simple protection schemes are devised. Also, discover how they are circumvented! A guide for the inexperienced computer user. By Rohn Smith.

## 27 RKCopy: Backing-Up SSI's Non-RDOS Disks

Use this program to backup such games as Germany 1985, Baltic 1985 and Reach For The Stars. By M.M. McFadden.

### DEPARTMENTS

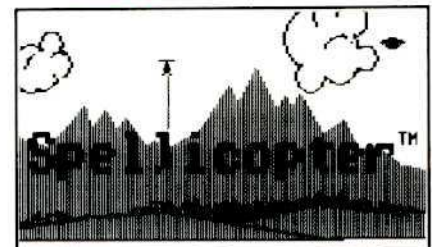
## 4 INPUT

## 7 READERS' SOFTKEY & COPY EXCHANGE

Softkey For  
 BPI Accounting Programs  
 By Roger J. Hofmeister

Deprotecting DesignWare Programs  
 By Clay Harrell

## 21 ADVENTURE TIPS & APT's Was!



# Input

## Stickybears

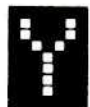


I have used the procedure for copying **Stickybear** disks (Hardcore COMPUTIST No. 15) successfully on Stickybear Opposites in addition to Stickybear BOP. This procedure was greatly appreciated since I was unable to backup any of these disks with the copy programs that I have (Copy II Plus and EDD III).

Please continue to provide helpful techniques such as this.

Stephen J. Ganocy

## Backing Up A2-FS2



Our readers might be interested to know how I was able to produce a working backup of Sublogic's **Flight Simulator II**.

First, looking in Locksmith's 5.0 Level F parameter book it listed Copy Tracks 00-22 with synchronized tracks and automatic nibble count. I did this and immediately got many #5 errors. For Locksmith, this indicates a failure of the nibble count routine. While watching the copy process on the screen, I noticed that some of Flight Simulator's tracks were written at different speeds. The Locksmith display screen shows the number of bytes on the original; some tracks had approximately \$1888-\$188A bytes while others had approximately \$18DA-\$18E0 bytes. With Apple drives running at or near factory settings, it is impossible to put 18E0 bytes onto a track, hence the reason for the #5 errors.

To solve this problem, it was apparent that I had to slow down the speed of the drive which was writing my backup. I have a new Apple duo drive and adjusting the drive speed is easily done. First, select Locksmith's speed check utility. Once it has been loaded, put a blank disk into drive 2 (the drive which will do the writing). Be sure to pick the "recommended" option when Locksmith prompts how you wish to set the drive speed. The recommended speed is slower than the factory speed, but it is still not slow enough to copy FS2.

Just before starting the process, turn the duo drive unit upside down. Yes, upside down! **Careful inspection of the drive case will reveal two holes, one for each drive. Looking inside the hole will reveal an**

**adjustment potentiometer.** To set the drive speed, simply insert a screwdriver into the screw *Do this before the drive light comes on to prevent accidental damage with the screwdriver and the electronics.*

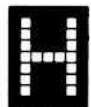
Now, as Locksmith displays your drive speed, adjust the speed so that the "dots" form a line one-half inch to the left of the center line. That should do the job. Now copy the disk using synchronized tracks and automatic nibble count. Your drive is adjusted correctly if you do not get any #5 errors. If you get 1 or 2 #5 errors, simply try to copy those tracks again. If that fails, slow down the drive speed a bit more. After completing this procedure, *don't forget* to reset your drive to its original speed. You can now fly to your heart's content without the fear of "crashing" your disk (pun intended). **NOTE:** Whenever I try to backup one of my original programs, I always use a freshly erased disk using Locksmith's erase utility.

Your magazine has greatly contributed to my understanding of the Apple //e, and you are providing a valuable service to honest users. Thanks. I have recommended your magazine to all my associates. I would also like to stress that any program worth having is worth buying. Although I am not a programmer, I feel strongly that programmers should be paid for their efforts. Also, without documentation, a copied program is useless.

P.S. After writing this letter, I recopied the disk without synchronization but still using an automatic nibble count. This backup also appears to function correctly.

J. C.  
Arlington VA

## BBS Fan



Here's my \$25 for 12 more issues of the world's greatest Apple magazine. I'm proud to say that I have a complete collection including all the updates and the "Best Of". It's the only magazine that I religiously read cover to cover and then save for future reference. And, I refer to the back issues often. I strongly recommend that other subscribers order those back issues that are available. They won't believe how often they will use them.

By the way, the **Index** in Issue No. 15 was a super idea. My compliments. And, if you're still counting votes, I cast mine

to keep Hardcore for Apple II's only. Why spoil a good thing?

In Issue No. 13, the softkey for **Quest** says to change line 9120 in the program QB from Z = Peek (47092) to Z = PEEK (47093). In my version, line 9120 contains Z = PEEK (47094). I tried changing the Peek to several values in the 47093 area, but the program still crashes just before the quest begins. Do you or any readers have any ideas?

New subject: I'm a big fan of **bulletin boards**. My two favorites are Pirates Harbor, (617) 720-3600, and Pirates Bay, (402) 563-1638. I'd like to see Hardcore sponsor or operate a BBS. Subscribers to Hardcore COMPUTIST could get special rates for BBS membership. Non-subscribers would pay higher rates to join, and/or not be granted access required to get the newest softkeys. Hardcore could sponsor a contest, probably monthly, and the four or five best softkeys posted during the previous month would be published in Hardcore with the authors being paid your regular writers' fee.

A single yearly winner might be awarded a special prize worth enough to attract lots of entrants, say a color monitor. One section might contain discount hardware for sale by your advertisers (they would have to pay to post). They might even contribute to, or provide, the yearly prize. There are several possibilities for other sections. Imagine the opportunities created when readers would have on-line feedback to the editorial staff, and the ability to leave E-mail for other readers. I get excited just thinking about it. I believe there would also be a synergistic effect for the magazine...more and better softkeys...and that phenomenal success of the magazine would rub off on the BBS making it quickly successful.

Another new subject: Here are a few COPYA softkeys some readers might find useful. I can't claim these. They were posted on my two favorite BBS' recently.

### 1) Some Milton Bradley software:

**RUN COPYA**

**CTRL C**

**70**

**CALL-151**

**B942: 18**

**3D0G**

**RUN (copy disk)**

### 2) Spellikazam by Designware

**RUN COPYA (copy disk)**

Edit: T0 S3 B64,65  
Was: A5,1E Change to: A9,FF

### 3) Microwave

Boot System Master

CALL-151

B925: 18 60

B988: 18 60

BE48: 18

3D0G

RUN (copy disk)

Edit: T1 SC B51,52,53  
Was: 37, B0, 18 Change to: 20 6F FD

Thanks for publishing a super magazine and providing a great information exchange in the process.

Rich Fabbre  
O'Fallon IL

*Mr. Fabbre: Thanks for your input. We always appreciate hearing suggestions from our readers.*

*Setting up a BBS for Hardcore readers is one topic that has arisen in several of our recent staff meetings, but at this time we just do not have the extra manhours to devote to a project of this sort. We would definitely like to do it in the near future, however.*

## A Big WishList

I would like to use this opportunity to pass along some comments about your fine publication and some suggestions for hardware and software articles I would really like to see in Hardcore.

First, you really could use a cumulative index, arranged alphabetically by software name (for softkeys and feature articles), ditto by the program name for Hardcore programs, author, subject, and subject in Input letters (and any other way which is helpful in locating information that I know I saw somewhere in one of the 21 publications I have received from you in the past 2 years!). After the first index, it would be appreciated if you would follow it annually with updated indexes.

I also have a Do-It-Yourself Hardware Projects wishlist:

1) An infra-red remote keyboard- how to add on a gadget to convert an existing Apple ][ keyboard into a remote unit, like the PC Junior's. Call-A.P.P.L.E. had a

hardware project which used IR data transmission to poll students from handheld multiple choice keyboards. Why not a full Apple keyboard! And do the same for game paddles!

2) I own a numeric keypad from The Keyboard Co. It's handy for Visicalc and other business uses, but a table space waster when I'm not using it. At first I tried to install a 12-15 pin connector to disconnect the thing from its cable, but this freezes up my Apple because the jumper board installed between the Apple's keyboard encoder and the motherboard requires that the numeric keypad always be connected. How stupid! I've heard that a standard calculator keypad can be connected to the Apple's encoder board at the pins which connect to the keyboard. Since the Apple ][ keyboard is just switches, connecting any simple keyboard to the encoder pins should work fine. So, there's a simple project for hardware do-it-yourself neophytes.

3) Also, how about making a dedicated hexadecimal keypad using some keypad available on the surplus market? Several years ago, commercial and homebrew alpha wave feedback "machines" were popular and recently I have seen simple computer games controlled by galvanic skin resistance. How about hardware projects to interface biofeedback through the game I/O socket?!

4) Software/ hardware project- Techniques to alter an old ROM card into an Ultra ROM-like board with 2732 eproms and the capability of putting DOS, GPLE, BIG MAC LC etc. corresident on Eproms!

Now here's my Software Wish Story: When it comes to cracking some of my protected disks, I am great at following instructions in your articles, but mostly lost when I try cracking on my own. I think what I've missed is a comprehensive feature article which carefully details how to go about analyzing any protected disk. It would be really great to see a complete flow-chart of procedures.

I also have a Tutorial Wish List: I use several programs which I have broken under your tutelage- Magic Window II, Visidex, Sensible Speller and DB Master. Having freed copies of Magic Window II allows me to save my text files onto the same disk from which I have booted Magic Window. I find this a wonderful time and effort saver and I do much less disk

shuffling. It would be great to have Sensible Speller on the same disk and be able to select it from the Magic Window menu. I believe that my broken Sensible Speller is a direct disk access loaded program, and that it may require its own DOS. What's required to modify Sensible Speller so it can be copied onto my Magic Window II working disks? Is it realistic to consider cohabiting the 3.3 version of Sensible Speller with Magic Window II on the same disk? In general, direct load programs really confuse me. Is there a simple way to take a broken direct loader program, such as The Spreadsheet 2.0, Sensible Speller, DB Master or Visidex, save it as a binary file and BRUN it directly from a program or the catalog?

These are my suggestions for the Hardcore Most Wanted List:

DB Master v3.0 to deprotect Basic and add a faster sort, DB Master v4.0, General Manager 2.0 (I really miss 80-columns in General Manager on my ][ Plus. Since On-Line won't write a video driver to support 80-columns on the ][ Plus for Visidex, etc., how tough could it be to say patch-in Magic Window II's excellent video driver, or something else?), Soft-Step by Accent Software- a nice Applesoft step, trace type of program debugger, and Time Manager.

Thanks for an always interesting and good magazine!

Ken Conrad  
Seattle, WA

## More Stickybear Info

Using the information presented in Jerry Caldwell's article, "Stickybear BOP Softkey", I have had success in backing-up several other releases in the series. Here is what I did:

1) Used the modified form of COPYA presented in the article.

2) Used the Tracer from the CIA Files to find the sector where the IOB is set up, then searched for a pattern of 8D EC B7 and determined the location on the disk of the protected sector and its destination in memory.

3) RESET into the monitor after booting the game and moved the data from the protected sector to \$9000 where it was safe from a reboot; booted a slave disk and

# Input cont...

B~~S~~A~~V~~E~~d~~ A\$9000,LS100 as stated in the article.

4) Used the Inspector and Watson to write the data recovered from the protected sector back to the COPYAed disk and also performed the sector edit that prevents the program from altering its RWTS.

I have found that this technique will work on four of the disks in the series without any alterations. Here is the data you will need:

Fat City, Stickybear Basket Bounce,  
Stickybear Numbers or Shapes

Sector which reads - Track \$02, Sector \$06  
protected sector.

Protected sector. - Track \$01, Sector \$0F

Destination of  
protected sector. - \$1F00

Sector edit. - Track \$02, Sector \$06  
Byte \$3A to \$60

Your magazine is a great help to Apple owners, particularly those with small accident-prone children.

Albert Snopes  
Jefferson, MS

*Mr. Snopes: Thanks for the information. We have heard that the technique described in Mr. Caldwell's article does not work on some versions of the Stickybear releases (Opposites and Numbers) because a slightly different protection scheme is being used. We would appreciate hearing from any readers who have any information about this alternate method of protection.*

## BSS & DLM

I am writing to request that you put **Bank Street Speller** on your Most Wanted List. I have tried several bit copiers and other methods to backup my copy but without success.

I would also like to make a comment about the Softkey For **DLM Software** in Issue No. 13. I have copied all 6 DLM math programs by switching DOS's and found that there are a number of files which are used repeatedly. Only the file "shapes" is used by different programs so these must be on separate disks. What can be done, then, is to put all 6 math programs on two disks with a little menu type of HELLO program.

Also, the article says that the author does not know if his softkey will work for other

DLM programs, but I have found that just moving DOS in and out (Super IOB and the Swap Controller) also works for **Word Invasion**.

John B. Walkey  
Lethbridge, Alberta  
Canada

## Alternate Master Type Softkey

Like Mr. Rongays, I had a protected version of **Master Type** and was happy to learn that someone had been able to make a COPYable version of it. However, I do not have an F8 monitor ROM so I had to find another way to load in Master Type's RWTS. This is how you can do it:

1) Boot a normal DOS 3.3 diskette and insert the Master Type disk.

2) Enter the monitor with

**CALL -151**

3) Move Boot 0 into RAM and modify it to jump to \$8801

**8600 < C600.C6FFM 86FA:88**

4) Place a jump into the monitor and start the boot

**8801:4C 59 FF**

**8600G**

5) Stop drive and move \$800 to \$8800

**C0E8**

**8800 < 800.8FFM**

6) Set an indirect jump to the modified read routine

**880E:80**

7) Return control after reading the RWTS and start the read

**884A:4C 59 FF**

**8600G**

8) Stop the read and move the RWTS to a safe location

**C0E8**

**1900 < B800.BFFFM**

9) Boot Super IOB disk

**6<sup>CTRL</sup>P**

10) And type

**BSAVE RWTS.MASTER TYPE,  
AS1900,LS800**

Continue at Step 4 in Mr. Rongays'

article and you now have a COPYable version of Master Type.

If you want to make this disk totally DOS 3.3, you must go one step further. First, copy all the files off the broken Master Type to a normal DOS 3.3 disk (I used Copy II + v4.4c to transfer files) and get ready to use your sector editor (I used Tricky Dick). Find all the occurrences of these low byte ASCII codes: CATNDOG, KILLDE, SAVE, YZ123, YZ23. Change them to CATALOG, DELETE, INIT, BLOAD and BRUN, respectively. Then find these high byte ASCII codes: YZ123 and YZ23. Change these to BLOAD and BRUN, respectively. You now have a normal copy of Master Type.

By the way, I was wondering if anyone has modified Muse's, "Super Text" so that it will work under DOS 3.3. I would appreciate any info on this topic.

P.S. You've got a great magazine going. Please don't spoil it by adding other personal computers.

Harry Noel  
Omaha, NE

## Needs Help with The Handlers

I have now been a subscriber for three years. Although I never write letters to the editor, I am now doing so. Please add the package called, "The Handlers" to your Most Wanted List.

The "Back-Up List" dated 6/15/84 has a user submitted softkey that is not correct; at least it does not copy my version, No. 4.2 of The Word Handler.

This set of programs, "The Handlers" by ALS of Sunnyvale, CA includes The Word Handler, The List Handler, and the Spell Handler, all for a semi-reasonable price. Sadly, they are all copy protected; worse, the programs are a bit out-of-date.

Does anyone out there know how to turn the word processor files into a DIF file or any other type that may be compatible with the more common "integrated" type systems?

Thank you for an excellent magazine and for all the answers you just can't find anywhere else.

Michael D. Mullins  
Gold Beach, OR



# Readers' Softkey & Copy Exchange

## Softkey For BPI Accounting Programs By Roger J. Hofmeister

BPI Systems, Inc.  
3423 Guadalupe  
Austin, TX 78705  
(512) 454-2801

### Requirements:

Apple II Plus or equivalent  
One disk drive  
Sector Editor  
Several blank disks



BPI uses a couple of methods to protect its popular accounting programs. First, they have changed the text of the DOS commands. The second line of defense is a routine which verifies the presence of an original BPI disk.

Here are the altered commands:

From	To	From	To
INIT	INIT	APPEND	I CTRL P
LOAD	DLD	RENAME	B CTRL G
SAVE	SAVE	CATALOG	P CTRL G
RUN	B CTRL K	MON	PR#
CHAIN	P CTRL K	NOMON	B CTRL Z
DELETE	I CTRL K	PR#	P CTRL Z
LOCK	LOCK	IN#	I CTRL Z
UNLOCK	UNLOCK	MAXFILES	B CTRL I
CLOSE	I CTRL A	FP	LIST
READ	B CTRL L	INT	CALL
EXEC	P CTRL L	BSAVE	B CTRL E
WRITE	I CTRL L	BLOAD	P CTRL E
POSITION	B CTRL P	BRUN	I CTRL E
OPEN	P CTRL P	VERIFY	BUZZOFF

By looking at this list you should see that if you hit a RESET from the main menu and type "LIST", what you are actually doing is an "FP" command which clears memory. A similar thing happens with a "CALL" command. We can softkey the disk to make it copyable with COPYA, but we'll need to keep these commands or the programs won't run.

### The Procedure

- 1) Initialize three disks (one for each of the program disks). Use faster DOS if you wish. Use the name "BOOTMSG" for the hello program.
- 2) Get out your sector editor and read track

1, sector 7 of the BPI disk. Write this sector to track 1, sector 7 of the newly initialized disks (this is the sector which contains the DOS command names).

3) Boot the System Master disk. Type BRUN FID. Use the wildcard feature "=" to transfer all the programs over to the new disks. When you are finished, exit FID.

At this point you should have all the programs on your new disks but, if you try to run them, they will crash at \$318 (or somewhere close). This is because the program "DRIVER" does a POKE to location \$A415 in DOS which causes the DOS routines to run a disk check which is loaded into page three of memory during the boot process. Since the routine isn't there, the program crashes. All we have to do is fix the POKE and we're done.

4) Insert your new BPI disk and load the driver program

### UNLOCK DRIVER LOAD DRIVER

5) Make the necessary changes depending upon which disk you are fixing (as shown below). Note: These changes may be difficult to make because all the lines are numbered consecutively and they put a lot of information on one line. You may want to use RENUMBER on this file to give yourself some room to play with.

#### Gen Ledger Data Entry Disk

Line 8, change E = 3882 to E = 5111

#### Gen Ledger Posting Disk

Line 9, change C = 1850 \* 20 to C = 2464 \* 20

#### Gen Ledger Maintenance Disk

Line 8, change O = 19905 to O = 26049

#### A/R Data Entry Disk

Line 19, change L = 20440 to L = 32728

#### A/R Reports Disk

Line 8, change S = 20440 to S = 32728

#### For A/R Maintenance Disk

Line 8, change A = 20440 to A = 32728

6) Save the driver program

### SAVE DRIVER LOCK DRIVER

7) Do the same for the other disks

### Other Disks

The technique for the inventory and A/P systems is probably similar, but I don't have those modules. Look for an integer variable that is manipulated a couple of times until it totals 42,005, then change it so that it totals approximately 54,293. This

will cause the POKE to go into ROM where it will have no effect on your programs. I changed the value of the variable rather than removing the POKE because you never know how many times the author put the POKE in there. Changing the variable will take care of any POKE that uses that variable.

### Finalities

Write protect your backup, put your originals away in a safe place and try out your new, hopefully faster, disks. By the way- you can change all of the strange DOS commands to their normal counterparts by using the search and replace feature of GPLE or some similar program. Someday, when I find the time, I'm going to try it. For now, my programs are working fine under a faster DOS so I'm content to leave them alone.

## Deprotecting DesignWare Programs By Clay Harrell

DesignWare, Inc.  
185 Berry St. Bldg. 3 Ste. 158  
San Francisco, CA 94107  
(415) 546-1866

### Requirements:

Apple II, II+, //c, //e  
A sector editor  
Any Designware program



The Designware company publishes some well-designed but, unfortunately, copy protected educational software. The owner of any educational software that is to be used by younger children would be wise to provide a backup for the children to experiment with, not the original. Designware programs are no exception.

Fortunately, Designware has used the same protection on all of its recent releases. I used "Spellicopter" and "States and Traits" for an example, but this procedure should work on all of Designware's recent releases.

If you listen to the Designware programs boot, they do not sound like a normal DOS disk booting. After the initial clackety-clack noise the drive makes when first booting a disk, the drive spins for a few

## Exchange cont...

seconds and then you can hear the drive read off consecutive tracks until a title page appears. The Applesoft prompt never appears and you never hear the drive head swing to the catalog track to find a file. This is a sure bet that the program uses some sort of loader or just the RWTS (Read Write Track Sector) portion of DOS to read in the program.

If you try boot-code tracing the program, you find that track 0, sector 0 (the first sector read in by the disk controller card) is lifted almost verbatim from a DOS 3.3 slave disk. Remember that the disk control card reads track 0, sector 0 into memory at \$800, and then executes this code (called boot1). Boot1 then reads track 0, sectors 0 to 9 into \$B600-BFFF. The code at \$B600-BFFF is called RWTS and it normally reads in the remainder of DOS. RWTS can also be used to read in a program directly without using the rest of DOS. This is called using second stage DOS or RWTS, and is the basis for Designware's protection scheme.

Since we now know that second stage DOS is being used (RWTS), we know that (or you should know that...) after sectors 0 to 9 are read into \$B600-BFFF, the code does an indirect jump through \$8FE. This means that the code jumps to the location in memory which is pointed to by the values in \$8FD and \$8FE. These two locations contain a \$00 and a \$B6, respectively. This tells us that the code jumps to \$B600. But wait a second: add \$100 to this value because location \$8FE gets incremented by the code at \$801. So, this means that after RWTS is read into \$B600-BFFF, the code jumps to \$B700.

Now boot your Designware program. After a few seconds you will hear the drive head move and start loading in the program. Hit RESET, enter the monitor, and examine the code at \$B700. It should look like this:

```

B700- 8E E9 B7 STX $B7E9 :-----
B703- 8E F7 B7 STX $B7F7 : s s
B706- A9 01 LDA #01 : e t
B708- 8D F8 B7 STA $B7F8 : t a
B70B- 8D EA B7 STA $B7EA : u g
B70E- AD E0 B7 LDA $B7E0 : p e
B711- 8D E1 B7 STA $B7E1 :
B714- A9 00 LDA #00 : s d
B716- 8D EC B7 STA $B7EC : e 0
B719- AD E2 B7 LDA $B7E2 : c s
B71C- 8D ED B7 STA $B7ED : o
B71F- AD E3 B7 LDA $B7E3 : n
B722- 8D F1 B7 STA $B7F1 : d
B725- A9 01 LDA #01 :
B727- 8D F4 B7 STA $B7F4 :
B72A- 8A TXA :-----

```

```

B72B- 4A LSR :
B72C- 4A LSR :
B72D- 4A LSR :
B72E- 4A LSR :
B72F- AA TAX :
B730- A9 00 LDA #00 :
B732- 9D F8 04 STA $04F8,X:
B735- 9D 78 04 STA $0478,X:
B738- 20 93 B7 JSR $B793 :read
B73B- A2 FF LDX #$FF :prog.
B73D- 9A TXS :
B73E- 8E EB B7 STX $B7EB :
B741- 4C C8 BF JMP $BFC8 :
B744- 20 89 FE JSR $FEB9 :
B747- 4C 00 08 JMP $0800 :start

```

What the above code does is set-up some parameters for RWTS and then JSR to \$B793 where the actual program is read in using second stage DOS (RWTS). Finally, it jumps to \$800 to start the program.

You may test this procedure by booting your Designware program and, after a few seconds when you hear the drive head move, hit RESET. Now, enter the monitor and type:

### B738G

This will load in the entire program and start the program. So big deal! Right? We found what the program is doing during the load, and how it starts the program. There's really nothing unusual about what we found, and we didn't find any protection code. However, if you use a sector editor to look at this code on the disk, you can discover what is really going on.

When RWTS gets read into \$B600-BFFF, it is read from track 0, sectors 0 to 9. If you look at track 0, sector 5 (the sector that is read into \$B700), you will notice at byte \$38 that the code does not say JSR \$B793 as indicated above. Instead, it says JSR \$BB00. But, if you ever examined the code at \$BB00, it was garbage. To see what is hidden, make a copy of your Designware program with COPYA (YES, it will copy without any errors or mods) and use your sector editor to change bytes \$39 and \$3A from \$00 \$BB to \$59 \$FF of track 0, sector 5. This will cause the program to bomb into the monitor instead of executing the code at \$BB00. Now you can list the code at \$BB00 by typing:

### BB00L

I will not list the code since it is unclear just exactly what it does since we destroy some referenced memory when we jump into the monitor. The point is that

Designware is being deceptive, and their protection lies here.

Fortunately, we can easily defeat their protection by making a two-byte change to track 0, sector 5. All that we must do is change the JSR \$BB00 to a JSR \$B793 on track 0, sector 5, byte \$38, which is what it eventually gets changed to anyway by the program.

It's nice to know that Designware made life easy for us to begin with since this is the sole protection, and the disk copies with COPYA (without any modifications). In review, these are the steps in making a back-up of programs published by Designware.

### The Procedure

1) Boot your DOS 3.3 System Master and run COPYA

#### RUN COPYA

2) Copy the Designware program to a blank disk.

3) Now run your sector editor and make the following change to the copied disk

```

Track 0, sector 5, Byte $39:
Change from $00 $BB
to $59 $B7

```

4) Write the sector back out to disk and you're all done!

## Note To Subscribers:

Hardcore COMPUTIST  
**does not send**  
 subscription renewal  
 notification.  
 This info appears  
 only on your  
 MAILING LABEL.

Please note that the Post Office **does not** inform our subscription department when you file a change of address. You **MUST** send a change of address to Hardcore COMPUTIST on postal form 3576. Issues missed due to non-receipt of change of address can be purchased at the current back issue rate.

# Softkey For The Scholastic Version of Bank Street Writer

By James Daniels

## Requirements:

Apple ][, ][+, or //e (minimum 48K and Applesoft Basic)  
One or two disk drives and DOS 3.3  
One blank disk  
Super IOB (Version 1.2)

**H**ardcore COMPUTIST No. 10 contained a softkey procedure for the Broderbund Software version of The Bank Street Writer; however, this excellent word processor, which is ideally tailored for the beginner, is marketed by two different companies. The first, as mentioned above, is Broderbund Software. The second, and the one at which this softkey procedure is targeted, is Scholastic Incorporated. Although the BSW word processor programs themselves are identical in each of the two versions, the copy protection schemes used to protect the two disks are based on two completely different techniques. The actual protection method used on the Scholastic BSW disk is fully described later in this article and those readers who wish to complete the procedure directly may simply skip the next section which describes the boot process and loading scheme of the original disk and proceed to either the description of the protection scheme or the softkey procedure.

## The Boot Process

As with all Apple disks, the firmware located in the disk controller card ROM (Read-Only Memory) must be able to read Track 0, Sector 0 from the BSW disk and load it into memory from \$800 through \$8FF. The firmware then checks the byte loaded at \$800 and, in a loop, continues loading sequential sectors from Track 0 until the number of sectors loaded is equal to the byte at \$800. On the BSW disk (and most other disks) the byte loaded at \$800 is "01"; therefore, only one sector (Sector 0) is loaded from Track 0. When the disk controller firmware finishes loading sectors, it enters and begins execution of the code at \$801 (the code which was read from Track 0, Sector 0).

At this point the BSW disk boot scheme begins to differ from a normal DOS 3.3 disk. When executed, the code beginning at \$801 first clears the Page 1 hi-res graphics screen memory (\$2000 through \$3FFF) and then, in a loop using a portion of firmware from the disk controller ROM as a subroutine (\$C65C with disk controller card in Slot 6), loads the remaining sectors from Track 0 to certain areas of memory. Table 1 reflects the addresses at which these "PHYSICAL" sectors are loaded.

Table 1

Physical Sector	Logical Sector	Load Address (Hex)
00	00	\$0800 TO \$08FF
01	07	Not Used
02	0E	\$9D00 TO \$9DFF
03	06	\$0900 TO \$09FF
04	0D	\$9E00 TO \$9EFF
05	05	\$0A00 TO \$0AFF
06	0C	\$9F00 TO \$9FFF
07	04	\$0B00 TO \$0BFF
08	0B	\$A000 TO \$A0FF
09	03	\$0C00 TO \$0CFF
0A	0A	\$0200 TO \$02FF
0B	02	\$0D00 TO \$0DFF
0C	09	\$0400 TO \$04FF
0D	01	\$0E00 TO \$0EFF
0E	08	Not Used
0F	0F	\$0F00 TO \$0FFF

"Physical" Sector numbers are the actual numbers of sectors as they *physically* appear on the disk in ascending order (\$00 to \$0F). These are the 4&4 encoded values you will find on a track if you read it with a nibble editor such as The Linguist or Diskview. The DOS 3.3 operating system converts these physical sector numbers to logical sector numbers to allow for faster read/write operations. Since the load scheme of Bank Street Writer deals almost exclusively with physical sectors and most, if not all, sector editors and other disk utilities deal with logical sectors, the middle column of Table 1 has been included to reflect the relationship between the two types of sectors in case you wish to examine the code more closely.

After all the sectors from Track 0 have been loaded, an assembly language "trick" is used to jump to the code at \$400. This trick is accomplished by first "pushing" the bytes \$05, \$04 and \$00 onto the 6502 processor stack and then executing an assembly language "RTS" instruction. This return from subroutine instruction "pulls" the last two bytes (\$00 and \$04) off the stack and loads them into the 6502 processor address buffer causing program execution to continue at a return address of \$400 + \$1 or \$401.

The code on page \$04 represents the heart of the disk protection scheme on the BSW disk. First, the code from \$802 through \$892 is cleared in an attempt to hide the path of

execution as well as the load addresses of the Track 0 sectors. Then, using a small subroutine located from \$470 to \$4B3, the disk drive read/write head is stepped from Track 0 to Track 1. The program then loads all but three sectors from Track 1 into memory by using the disk controller firmware as a subroutine. Table 2 reflects the load addresses of these sectors. Next a jump is made to a subroutine located at \$900 which clears and then fills the page 2 hi-res graphics screen (\$4000 through \$5FFF) with the Bank Street Writer Logo and displays it on your monitor before returning.

Examination of the block of code at \$470, mentioned above, reveals that if execution of this subroutine is begun at \$470 the drive read/write head is stepped up a full track from the preceding track. However, if begun at \$473 the head is only stepped up a half-track. The necessity of this half-track access capability becomes apparent following the loading of the Track 1 sectors. The drive is first stepped up a full track to Track 2 and physical sectors 1 through 5 are read and stored in memory. Then, a half-track step is made to Track 2.5 where physical sectors 6 through 10 are read and stored, and finally another half-track step is made to Track 3 where physical sectors 11 through 15 are read and stored in memory. This type of half-track stepping and reading represents a protection scheme called, "SPIRAL TRACKING" or "TRACK-ARCING" and is the main line of defense of the BSW disk. This scheme is more fully addressed later in this article.

From here the same programming "trick" as before is used to continue execution of the disk boot process of the code at \$501. This is accomplished by pushing a \$00 onto the microprocessor stack to join the \$05 already there and executing an "RTS" instruction. Using the disk head stepper routine from \$470 and the disk controller ROM (\$C65C) firmware as subroutines, the code beginning at \$501, as well as several small routines throughout the \$600 to \$6FF range, complete the loading of the remaining BSW software.

For the most part, the remainder of the track/sector loading from the BSW disk is dependent on the type (Apple ][, ][+ or //e) and memory configuration (RAM cards, Applesoft Basic in ROM, etc) of the computer on which the BSW disk is booted. Discussion of this is beyond the intent of this article in that it has nothing to do with deprotecting the disk. Because a similar dependent load scheme was discussed in depth in the softkey article for the Broderbund version of The Bank Street Writer, it will not be discussed here.

Following the loading of all the BSW software, the code coldstarts or initializes the type of Applesoft Basic in the computer and then displays the main BSW text entry screen on the page 1 hi-res graphics screen (\$2000 through \$3FFF). At this point the BSW disk boot process is complete and the word processor is loaded and ready for text input.

**Hardcore COMPUTIST** welcomes articles of interest to users of the Apple II (or compatible) computers and would like to publish well-written material including:

- Softkeys
- Hardware Modifications
- Advanced Playing Techniques
- DOS modifications
- Utilities
- Product reviews
- Adventure Tips
- Original programs
- Do-it-yourself hardware projects
- General interest articles
- Bit-Copy Parameters

Send your submission on a DOS 3.3 disk using an Apple (or compatible) editing program. Enclose a double-spaced hardcopy manuscript (typewritten or computer printed). Submissions will be returned only if adequate packaging is enclosed.

**Have you  
written  
an ARTICLE or  
PROGRAM  
you'd like to see  
published in  
Hardcore  
COMPUTIST?  
We would like to hear  
from you!**

**Hardcore COMPUTIST** pays on publication. Rate of payment depends on the amount of editing necessary and the length of the article- generally between \$10 for a short softkey, and \$50 per typeset page for a full-length article. For a higher pay rate, enclose the original commercial disk for verification of softkeys. We guarantee the disk's return.

Softkey Publishing buys all rights as well as one-time reprint rights (for upcoming BEST OF Hardcore) on general articles, and exclusive rights on programs. However, alternate arrangements may be made with individual authors, depending on the merit of the contribution.

For a copy of our WRITER'S GUIDE, send a business-sized (22-cent) SASE (self-addressed, stamped envelope) to:

**Hardcore COMPUTIST**  
Writer's Guide  
PO Box 110846-K  
Tacoma, WA 98411

Table 2

Tracks and Sectors Associated  
with Disk Protection

Physical Sector	Track Number			
	1	2	2.5	3
0	\$1000	NL	NL	NL
1	NL	\$8900	NL	NL
2	\$1800	\$8A00	NL	NL
3	\$1100	\$8800	NL	NL
4	\$1900	\$8C00	NL	NL
5	\$1200	\$8D00	NL	NL
6	\$1A00	NL	\$8E00	NL
7	\$1300	NL	\$8F00	NL
8	\$0500	NL	\$9000	NL
9	\$1400	NL	\$9100	NL
A	\$0600	NL	\$9200	NL
B	\$1500	NL	NL	\$9300
C	NL	NL	NL	\$9400
D	\$1600	NL	NL	\$9500
E	NL	NL	NL	\$9600
F	\$1700	NL	NL	\$9700

\*NL = NOT LOADED

### The Protection

As with most protected software available for the Apple today, the protection on the Bank Street Writer disk is set up in layers. These layers are designed to both stop the disk from being copied by normal means and to stop or inhibit a nibble copier from producing a functioning backup.

These goals were accomplished on the BSW disk in three ways. The first layer utilizes a not widely known fact concerning the disk controller firmware. While it is true that, when executed, this firmware must be able to read Sector 0 of Track 0 from any disk, protected or not, it does not mean that Sector 0 must be either listable or copyable by "normal" methods.

If you were to closely examine the firmware from the disk controller ROM you would realize that it does not verify the presence or accuracy of the checksum or epilog marks of the address field of the sector it is reading. Therefore, the checksum which is calculated by exclusively-or-ing (EOR) the disk volume number with the track number and the sector number, all of which are contained in the address field, may purposefully be made incorrect without affecting the boot process of the disk. This is also true of the address field epilog marks. The normal DOS 3.3 marks (\$DE \$AA) are not verified by the firmware and could be changed to make the sector unreadable and uncopyable by normal methods. When your original BSW disk was recorded, the address field checksum of Track 0 Sector 0 was altered and all sector address field epilog marks were changed to \$FF \$FF, thus rendering all sectors both unreadable and uncopyable by DOS 3.3.

The second layer of defense on the BSW disk consists of having all track numbers within the address fields from Track \$0 through Track \$10 set as if they were all Track \$0. Since the protection scheme of the BSW disk utilizes both its own disk drive read/write head stepper routine and the disk

controller ROM firmware to read and load all software from the disk, it is a simple matter to treat each track as Track \$0 and thus, read the disk with no problem. However, this plays havoc with any normal copy program since it would be searching for the normal Track numbers \$0,\$1,\$2,\$3,\$4.....\$10. This also stops many sector editors and other utilities from accessing these tracks and sectors since the program relies on the address field data to locate the correct track and sector.

Both protection layers discussed above will stop the disk from being copied with "COPYA" or any other normal copy program. However, practically any nibble copier would easily defeat them and produce a functional, though still protected, backup. For that reason, the third and final layer of protection on the BSW disk utilizes another idiosyncrasy of the Apple computer to produce a sophisticated protection scheme which, while not impossible, is extremely difficult for a nibble copier to defeat. This restriction of the Apple disk drive centers around the requirement that data recorded on a disk in concentric rings or tracks must be physically separated on the disk by a full track width on either side. This is due to the physical width of the read/write head in the disk drive. If data is physically recorded on a disk side by side with less than a full track separating each side then the read/write head will overlap the adjacent tracks during both read and write operations scrambling the data.

The protection scheme based on this and used on the Scholastic BSW disk is called, "SPIRAL TRACKING" or "TRACK-ARCING". This scheme writes data on adjacent half-tracks but ensures that the data does not physically lie side by side by recording only a few sequential sectors on each half-track then stepping to the next half-track and recording a few more sectors and so on. Table 2 reflects this technique as it was used on Tracks 2, 2.5 and 3 of the BSW disk. Since the sectors are written in a "SPIRAL" pattern they are assured of at least a full track width separation on either side. While a nibble copier can read any of these sectors, it is very unlikely that it will write them to a backup disk in the same physical relationship as they appeared on the original disk which would cause overlapping and scrambling of data. Unfortunately, this type of protection scheme is becoming more and more prevalent in software for the Apple. However, with a little knowledge of your computer and 6502 assembly language and a little patience, it can be defeated as described next.

### The Softkey

While Scholastic Incorporated's, Bank Street Writer disk was originally deprotected with a copy program of my own design, the softkey procedure readily adapts to Ray Darrah's, Super IOB. The controller written

to backup the original disk is divided into three sections. The first section copies all the tracks marked as if they were Track \$0 (Tracks \$0 - \$1 and Tracks \$4 through \$10). The second reads the spiral sectors from Tracks \$2, \$2.5 and \$3 and rewrites them all to Track \$2 of the backup disk. The third portion then completes the copy by duplicating Tracks \$11 through \$28 which differ from normal DOS 3.3 tracks only in that the address field epilog marks have been changed from \$DE \$AA to \$FF \$FF. The controller also acts as a sector editor and alters a few strategic bytes on the backup disk in order to stop the disk from accessing the spiral tracks which will not exist on the copy of BSW.

We are now ready to get started with the softkey. Carefully follow the steps listed below and you will soon have a backup of The Bank Street Writer that loads just as fast and behaves identically to the original disk except that it is completely copyable.

1) Type in the Bank Street Writer controller and save it to a formatted disk with:

**SAVE BSW.CONTROLLER**

2) LOAD SUPER IOB 1.2 and modify line 60 of the program to read as follows:

```
60 LOMEM:8448:HIMEM:9983:DIM
   SN(16):GOTO 10010
```

3) RUN this slightly modified SUPER IOB with the BSW controller installed. Be sure to format the target disk when SUPER IOB asks.

As SUPER IOB produces your deprotected backup disk you will notice that the disk drive used to read the original disk will recalibrate (the clackety-clacking noise) several times. This will occur each time the drive begins reading a range of tracks from the original disk and again when the drive attempts to read Track \$11.

4) You are now finished and have a completely copyable version of Scholastic Incorporated's Bank Street Writer.

## BSW.CONTROLLER

```
1000 REM BSW.CONTROLLER
1010 TK=0:T1=0:T2=0:ST=0:LT=
   29:CD=WR
1020 RESTORE:GOSUB 270:GOSUB 170:
   GOSUB 490
1030 TK=0:CD=0:GOSUB 100:GOSUB
   80:S=-128:GOSUB 130:CD=RD
   :S=T1*2:GOSUB 130
1040 TK=T1:GOSUB 430:IF TK<17
   THEN TK=0
1050 GOSUB 100:ST=ST+1:IF ST<
   DOS THEN 1040
1060 S=2:GOSUB 130:T1=T1+1:IF
   T1=2 THEN 1150
1070 ST=0:IF BF THEN 1090
1080 IF T1<LT THEN 1040
1090 TK=T1:T1=T2:GOSUB 310:
   GOSUB 230:GOSUB 490:ST=0:T1=
   TK:TK=T2
```

```
1100 GOSUB 430:GOSUB 100:ST=ST+1
   :IF ST<DOS THEN 1100
1110 ST=0:TK=TK+1:IF TK=2 THEN
   1210
1120 IF BF=0 AND TK<LT THEN 1100
1130 T2=TK:IF T2<LT THEN 1020
1140 HOME:PRINT "COPY^COMPLETE":
   END
1150 FOR I=0 TO 15:READ SN(I):
   NEXT
1160 SP=1:LS=6
1170 ST=SP:TK=2:GOSUB 430:TK=0
   :ST=SN(SP):GOSUB 100:SP=SP
   +1:IF SP<LS THEN 1170
1180 IF SP=16 THEN 1090
1190 LS=11:IF SP=11 THEN LS=16
1200 S=1:GOSUB 130:GOTO 1170
1210 SP=1:LS=16
1220 ST=SP:GOSUB 430:ST=SN(SP)
   :GOSUB 100:SP=SP+1:IF SP<
   LS THEN 1220
1230 ST=0:T2=4:T1=4:GOTO 1020
1240 DATA 255,255,255,255
1250 DATA 0,7,14,6,13,5,12,4,11
   ,3,10,2,9,1,8,15
1260 DATA 5CHANGES,0,9,44,15,0,9
   ,49,112,0,9,51,96,0,9,52
   ,234,0,9,127,02
```

## Controller Checksums

1000 - \$356B	1140 - \$61C6
1010 - \$ED2E	1150 - \$8F28
1020 - \$F6AC	1160 - \$3CBA
1030 - \$C1F1	1170 - \$0F98
1040 - \$9109	1180 - \$0268
1050 - \$0F23	1190 - \$6C55
1060 - \$51F9	1200 - \$2596
1070 - \$83C0	1210 - \$761E
1080 - \$06EB	1220 - \$1698
1090 - \$13E0	1230 - \$E2CC
1100 - \$22E0	1240 - \$8DD3
1110 - \$EF3C	1250 - \$6697
1120 - \$A5AD	1260 - \$799D
1130 - \$5C86	

## Some Words About The Controller

The following is a line by line description of the basic functions of the BSW.CONTROLLER.

Line 1010- Set Track 28 (\$1C) as the last track to copy.

Line 1020- Restore the pointer to the beginning of the Data items. Ignore address field checksums. Change address field & data field epilog marks from \$DE \$AA to \$FF \$FF for reading.

Line 1030- Recalibrate the disk drive read/write head and then step to the proper track to read.

Line 1040- If track to read is less than 17 (\$11), set track marker to 0.

Line 1050- Read all sectors on the current track and store them in the buffer.

Line 1060- Step the drive to the next track. If on Track 2, jump to the routine which

reads from the "spiral" tracks. Line 1070- If the buffer is full, jump to write routine.

Line 1080- Loop to continue reading the original disk.

Line 1090- If Track 0 is contained in the buffer, edit Sector 9 as described below. Restore DOS to normalize its configuration before writing.

Line 1100- Write all of the sectors from the current track to the target disk.

Line 1110- Increment the track number to be written. If on Track 2, jump to the routine which writes all the spiral sectors to Track 2.

Line 1120- If the buffer still contains data and the last track has not been written, loop back and continue writing.

Line 1130- If the copy process is not complete, loop to read the next range of tracks from the original disk.

Line 1140- Copy finished, so END.

Line 1150- Read logical to physical sector interleaving table and store in array SN.

Lines 1160 through Line 1200- Read sectors from spiral tracks 2, 2.5 and 3.

Line 1190- When finished, jump to the track write routine.

Lines 1210 through Line 1230 - Write all spiral sectors to Track 2 of target disk.

Line 1240- New sector epilog marks.

Line 1250- Logical to physical sector interleaving data.

Line 1260- Data to edit Track 0 Sector 9.

## Sectors Edits Performed By The Controller

While copying Track 0 Sector 9, the Super IOB program and the BSW controller make the following changes to the sector before writing it to the target disk.

ADDRESS	FROM	TO	REASON
BYTE \$2C	\$05	\$0F	Changed to cause the program to read sectors \$01 through \$0F from Track 2 instead of only sectors \$01 through \$05.
BYTE \$31	\$73	\$70	Changed to stop the program from accessing the half-track at 2.5.
BYTE \$33	\$A2	\$60	Changed to make the program exit this code after all spiral sectors have been read.
BYTE \$34	\$0A	\$EA	Changed only to make a disassembly listing of Sector 9 read correctly.
BYTE \$7F	\$20	\$02	Changed to make the program begin looking for the normal track numbers when reading Track \$1 instead of Track \$11.



# SOFTKEYING APPLEWRITER //e

By  
**Peter  
Edelsten**

Apple Writer //e  
Apple Computer Inc.  
20525 Mariani Ave  
Cupertino, CA 95014  
\$149.00

## Requirements:

Apple //e  
COPYA or any other normal copy program  
One blank disk  
Disk Editor

I recently procured two Apple ][e's with hard disks and will be using them to coordinate a rural development project in Tanzania. We are successfully using Applewriter for word processing, but are frustrated because we cannot use Applewriter with the hard disks. The major problem appears to be that

the Applewriter DOS is incompatible with the DOS used by the hard disks. However, before I could even begin to tackle the problem of moving Applewriter //e to a hard disk, I had to first remove the program's copy protection.

The Applewriter diskette can be copied with COPYA, and CATALOGed from normal DOS. However, the copied disk will not work, as the program checks for information, embedded between the DOS sectors on the disk, which is not copied by COPYA. If this check is unsuccessful, the program wipes out memory and jumps to the Applesoft coldstart entry point.

There are two parts to this routine: the check for the embedded codes, and the jump to the routine which wipes memory. If these routines had appeared in several locations on the disk, it could have been very difficult to find them. Fortunately, they were fairly close together in memory and were executed shortly after the main program was loaded in. I found the disk's protection by gradually tracing the disk's boot and inserting jumps to the monitor (JMP FF59) in appropriate places using a sector editor. I also discovered that because Applewriter sets up a routine of its own to handle BRK's, inserting BRKs (00's) would not work.

I found the boot of Applewriter //e to be as follows: after DOS has been loaded in and initialized, it loads a file called OBJ.BOOT which BLOADs and executes one of three

files depending on the model: OBJ.APWRT][F for a 128K Apple //e, OBJ.APWRT][E for a 64K Apple //e, or OBJ.APWRT][D (which simply puts up a message telling you that you do not have an Apple //e). The protection routine is contained in both OBJ.APWRT][F and OBJ.APWRT][E. The offending routine looks like: (this version from OBJ.APWRT][F)

```

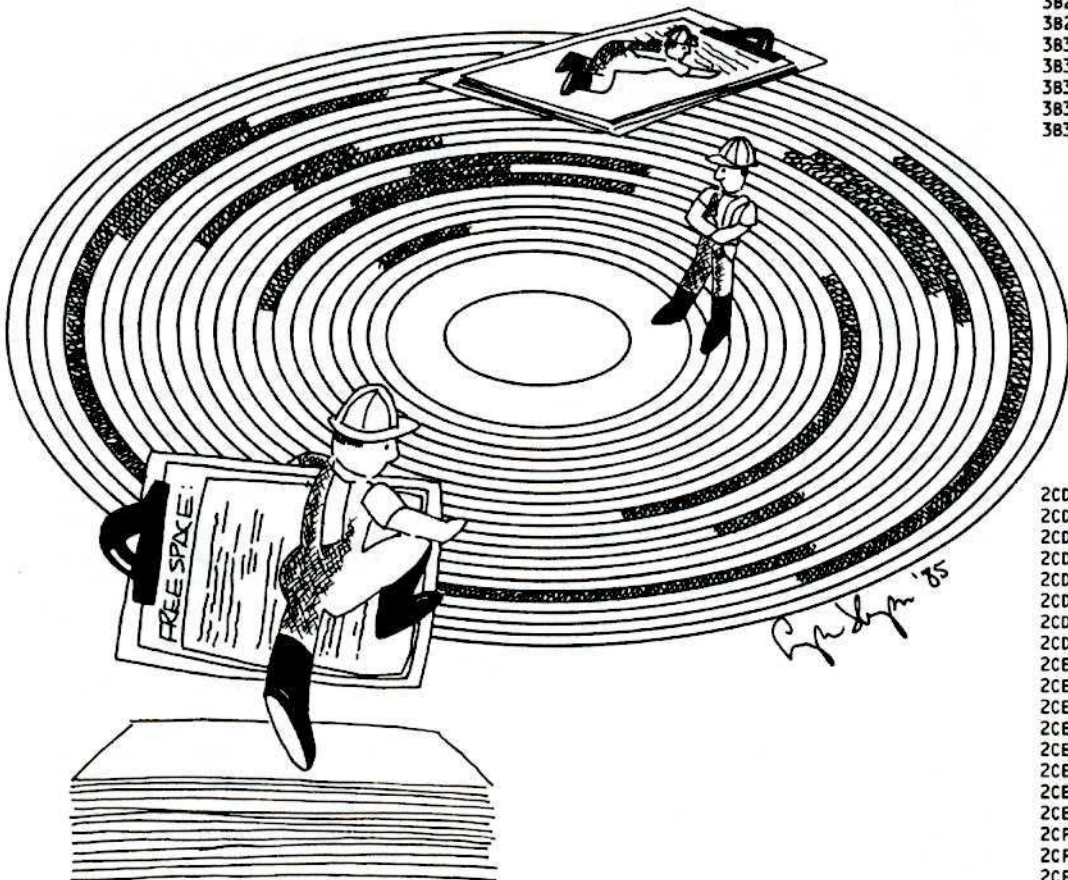
3AF1- 20 08 3B JSR $3B08 ;Read
3AF4- 85 82 STA $82 ;Store
3AF6- 20 08 3B JSR $3B08 ;Read
3AF9- C5 82 CMP $82 ;Same?
3AFB- D0 0A BNE $3B07 ;No, OK!
3AFD- 20 08 3B JSR $3B08 ;Read
3B00- C5 82 CMP $82 ;Same?
3B02- D0 03 BNE $3B07 ;No, OK!
3B04- 4C BF 2C JMP $2CFB ;Nasty!
3B07- 60 RTS ;Good.

3B08- AE E1 02 LDX $02E1 ;slot #
3B0B- BD 8C C0 LDA $C08C,X ;Read,
3B0E- 10 FB BPL $3B0B ;look
3B10- C9 D5 CMP #$D5 ;for
3B12- D0 F4 BNE $3B08 ;$D5.
3B14- EA NOP
3B15- BD 8C C0 LDA $C08C,X ;Read,
3B18- 10 FB BPL $3B15 ;look
3B1A- C9 AA CMP #$AA ;for
3B1C- D0 F2 BNE $3B10 ;$AA.
3B1E- EA NOP
3B1F- BD 8C C0 LDA $C08C,X ;Read,
3B22- 10 FB BPL $3B1F ;look
3B24- C9 96 CMP #$96 ;for
3B26- D0 E8 BNE $3B10 ;$96.
3B28- EA NOP
3B29- EA NOP
3B2A- BD 8C C0 LDA $C08C,X ;Read
3B2D- 10 FB BPL $3B2A ;twice
3B2F- 2A ROL ;and
3B30- 85 80 STA $80 ;store
3B32- BD 8C C0 LDA $C08C,X ;a
3B35- 10 FB BPL $3B32 ;value
3B37- 25 80 AND $80 ;at $80
3B39- 60 RTS
    
```

If this routine does not like what it finds on the disk, the JMP \$2CFB at \$3B04 is taken, where a routine which fills memory with \$00's is moved into the input buffer at \$200 and then executed. The code for the "memory wipe" routine is stored at \$2CD0 (before it is moved to \$200), and looks like the following:

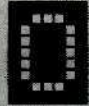
```

2CD0- AD 83 C0 LDA $C083 ;Ramcard
2CD3- AD 83 C0 LDA $C083 ;on.
2CD6- A9 03 LDA #$03 ;Set up
2CD8- 85 01 STA $01 ;pointers
2CDA- A0 00 LDY $00 ;and fill
2CDC- 84 00 STY $00 ;memory
2CDE- 98 TYA ;from
2CDF- 91 00 STA ($00),Y ;$300
2CE1- C8 INY ;through
2CE2- D0 FB BNE $2CDF ;$BFFF
2CE4- E6 01 INC $01 ;and
2CE6- F0 0C BEQ $2CF4 ;$D000
2CE8- A6 01 LDX $01 ;though
2CEA- E0 C0 CPX #$C0 ;$FFF
2CEC- D0 F1 BNE $2CDF ;(in the
2CEE- A2 D0 LDX #$D0 ;Ramcard)
2CF0- 86 01 STX $01 ;with
2CF2- D0 EB BNE $2CDF ;00's.
2CF4- AD 82 C0 LDA $C082 ;Ramcard
2CF7- AD 82 C0 LDA $C082 ;off.
2CFA- 8D 0C STA $C00C ;80 col.
;off.
    
```



# Installing A Free Sector Patch Into Appewriter //e

By Bob Bragner



One useful function that was not provided for Appewriter was a convenient way to determine the amount of free space left on a disk. The manual (page 89) advises you to catalog the disk and then add up the sizes of the files! No thanks. I prefer to let my Apple perform menial tasks such as that.

Listing 1 below shows a Free Sector patch for Version F of Appewriter. The routine has been located at \$2D0F, where the code for option J (Convert Appewriter 1.1 files) of the <sup>CTRL</sup>Q menu used to be.

The patch makes use of three Appewriter internal routines: CONVERT, which takes two values from the A (high) and X (low) registers and converts them into a decimal value leaving them in a buffer at \$16F0.16F4; OUTPUT, which is Appewriter's version of the Monitor's COUT (\$FDED) routine; and BELL, the absurd Appewriter tootle.

Listing 2 patches option J of the <sup>CTRL</sup>Q menu. Listing 3 is a control text file to install the patches.

## How To Do It

**WARNING:** Perform this patch ONLY on your softkeyed copy and only with APWRT][F.

1) Type in the first hexdump at the end of this article and save it with

**B\$AVE OBJ.FRESEC1,A\$2D0F,L\$53**

2) Type in the second hexdump at the end of this article and save it with

**B\$AVE OBJ.FRESEC2,A\$519B,L\$22**

3) Load the Appewriter file

**BLOAD OBJ.APWRT][F,A\$2300**

4) Load in the two patch files

**BLOAD OBJ.FRESEC1  
BLOAD OBJ.FRESEC2**

5) Save the newly patched Appewriter file

**UNLOCK OBJ.APWRT][F  
B\$AVE OBJ.APWRT][F,  
A\$2300,L\$30D1  
LOCK OBJ.APWRT][F**

## How To Use It

Now whenever you type <sup>CTRL</sup>Q followed

*Continued on next page*

```
2CFD- 20 84 FE JSR $FE84 ;Setnorm.
```

```
2D00- 20 2F FB JSR $FB2F ;Text on.
2D03- 20 93 FE JSR $FE93 ;DOS
2D06- 20 89 FE JSR $FE89 ;off.
2D09- 20 58 FC JSR $FC58 ;Home.
2D0C- 4C 00 E0 JMP $E000 ;BASIC.
```

This routine which clears memory is only executed from one place in the program, and now all that must be done to remove the copy-protection is to NOP the three bytes at \$3B04-6. The code which jumps to the memory wipe routine in the OBJ.APWRT][E version is JMP \$2BEC at \$39AD. These modifications can be easily made with any sector editor. Here is the procedure I used to produce the protected version of Appewriter //e.

## Making the Copy

- 1) Copy the disk with COPYA or any other normal copy program.
- 2) Make the following sector edits

Track	Sector	Byte	From	To
\$04	\$0C	\$B1	\$4C	\$EA
\$04	\$0C	\$B2	\$EC	\$EA
\$04	\$0C	\$B3	\$2B	\$EA
\$07	\$0C	\$08	\$4C	\$EA
\$07	\$0C	\$09	\$BF	\$EA
\$07	\$0C	\$09	\$2C	\$EA

If your disk differs from mine, you should use a disk search utility and look for JMP xxxx RTS (4C XX XX 60) sequences on the disk and replace the 3 bytes preceding the 60's with EA's.

Now onward to disassembly of the Appewriter //e DOS.

## FRESEC1 Source Code

```

-----*
*                                     EXTERNAL EQUATES                                     *
*-----*
16F0- DEC.WORK .EQ $16F0   WHERE DECIMAL ASCII IS STORED AFTER CALLING CONVERT
24AC- BELL     .EQ $24AC   AW//E ROUTINE THAT BEEPS
4951- CONVERT  .EQ $4951   AW//E ROUTINE THAT CONVERTS HEX (A,X) INTO DECIMAL
49FF- OUTPUT   .EQ $49FF   AW//E EQUIVELANT OF COUT ($FDF0)
EBF2- BITMAP   .EQ $B3F2+$3800 WHERE THE LAST ACCESSED DISKS VTOC IS

*-----*
*                                     ORIGINS AND STUFF                                     *
*-----*

.OR $2D0F
.TF OBJ.FRESEC1

*-----*
*                                     INITIALIZE REGISTERS                                 *
*-----*

2D0F: A0 C8   INIT      LDY #C8      # BYTES IN TRACK BITMAP
2D11: A2 00                          LDX #0       ZERO THE HEX LO BYTE
2D13: 8E 52 2D          STX HEX.MSB  AND THE HEX HI BYTE

*-----*
*                                     CALCULATE THE NUMBER OF FREE SECTORS                *
*-----*

2D16: B9 F2 EB   NXT.BYTE LDA BITMAP,Y GET A VTOC BIT MAP BYTE
2D19: F0 0B     END.TST  BEQ NXT.ADDR NO MORE FREE SECTORS, GET NEXT BYTE
2D1B: 0A          NXT.SECT ASL          GET SECTOR STATUS IN C
2D1C: 90 FB     BCC END.TST NOT FREE, GET NEXT ONE
2D1E: E8          INX          SECTOR IS FREE, INC HEX LSB
2D1F: D0 FA     BNE NXT.SECT NO PAGE CROSSING, CONTINUE
2D21: EE 52 2D  INC HEX.MSB FOR EVERY 256 FREE SECTORS FOUND
2D24: D0 F5     BNE NXT.SECT ..ALWAYS

2D26: 88          NXT.ADDR DEY          GET NEXT OFFSET
2D27: D0 ED     BNE NXT.BYTE IF NOT DONE, CONTINUE

*-----*
*                                     TURN NUMBER OF FREE SECTORS INTO DECIMAL            *
*-----*

```

*Continued on next page*

Continued from previous page

by J, the message, "FREE SECTORS:xyz" will appear near the bottom of the display. "xyz" shows the amount of space still available on the last accessed disk. If, for any reason, you have changed disks without informing Appewriter of the fact, the value returned by FRESEC will almost certainly be irrelevant. Any disk access at all such as CATALOG, SAVE, LOAD, etc., automatically updates DOS' (and thus, FRESEC's) information about the current disk.

Note the amount of free space on the disk, and press any key to return to the word processor.

My special thanks to Don Lancaster, whose, "Tearing into Appewriter IIe" (now called "AWIle Toolkit") gave me a huge shove in the right direction.

### FRESEC1 Hexdump

```

2D0F: A0                      $93DF
2D10: C8 A2 00 8E 52 2D B9 F2 $5D81
2D18: EB F0 0B 0A 90 FB E8 D0  $44CD
2D20: FA EE 52 2D D0 F5 88 D0  $F07E
2D28: ED AD 52 2D 20 51 49 A0  $849C
2D30: 0F B9 52 2D 20 FF 49 88  $016B
2D38: D0 F7 BD F0 16 20 FF 49  $0B10
2D40: CA 10 F7 20 AC 24 8D 10  $6524
2D48: C0 2C 00 C0 10 FB 8D 10  $DCBC
2D50: C0 60 00 BA D3 D2 CF D4  $5690

2D58: C3 C5 D3 A0 C5 C5 D2 C6  $B11A
2D60: 8D 8D                      $1033

```

### FRESEC2 Hexdump

```

519B: CA AE A0 C3 EF          $CAAF2
51A0: F5 EE F4 A0 E6 F2 E5 E5 $50F2
51A8: A0 F3 E5 E3 F4 EF F2 F3 $1737
51B0: A0 EF EE A0 E4 E9 F3 EB  $8A4C
51B8: A0 A0 A0 A0 8D          $4584

```



You can get  
the programs  
that appear in  
Hardcore  
COMPUTIST  
on disk!

(See page 31  
for information)

Continued from previous page

```

2D29: AD 52 2D          LDA HEX.MSB HI BYTE IN A REG (LO BYTE ALREADY IN X)
2D2C: 20 51 49        JSR CONVERT CALL THE CONVERT ROUTINE IN AW//E
*-----*
*                   PRINT THE FREE SECTORS MESSAGE                   *
*-----*

2D2F: A0 0F          LDY #END-MSG
2D31: B9 52 2D    PRNT.MSG LDA MSG-1,Y GET A BYTE OF THE MESSAGE
2D34: 20 FF 49    JSR OUTPUT PRINT IT
2D37: 88          DEY
2D38: D0 F7        BNE PRNT.MSG NOT END OF MESSAGE, KEEP GOING

*-----*
*                   PRINT THE DECIMAL VALUE                           *
*-----*

2D3A: BD F0 16    PRNT.DEC LDA DEC.WORK,X PRINT THE NUMBER FOUND
2D3D: 20 FF 49    JSR OUTPUT
2D40: CA          DEX GET NEXT INDEX
2D41: 10 F7        BPL PRNT.DEC NOT DONE, CONTINUE

*-----*
*                   BEEP AND WAIT FOR A KEY                           *
*-----*

2D43: 20 AC 24        JSR BELL
2D46: 8D 10 C0       STA $C010 CLEAR ANY KEYS
2D49: 2C 00 C0    WAIT BIT $C000 TEST FOR A KEYPRESS
2D4C: 10 FB        BPL WAIT NOT YET, KEEP TESTING
2D4E: 8D 10 C0       STA $C010 CLEAR KEY PRESSED
2D51: 60          RTS BACK TO APPEWRITER

*-----*
*                   STORAGE OF MSB OF HEX DIGIT OF NUMBER OF FREE SECTORS *
*-----*

2D52- HEX.MSB .BS 1 MSB OF HEX NUMBER OF SECTORS FREE

*-----*
*                   FREE SECTORS MESSAGE                               *
*-----*

2D53: BA D3 D2
2D56: CF D4 C3
2D59: C5 D3 A0
2D5C: C5 C5 D2
2D5F: C6 MSG .AS -/:SROTCE EERF/ 'FREE SECTORS' BACKWARDS
2D60: 8D 8D .HS 8D8D RETURN, RETURN
END

```



### FRESEC2 Source Code

```

.OR $519B
.TF OBJ.FRESEC2

519B: CA AE A0
519E: C3 EF F5
51A1: EE F4 A0
51A4: E6 F2 E5
51A7: E5 A0 F3
51AA: E5 E3 F4
51AD: EF F2 F3
51B0: A0 FF FE
51B3: A0 F4 F9
51B6: G3 FB A0
51B9: A0 A0 A0 .AS -"J. Count free sectors on disk "
51BC: 8D .HS 8D

```





# The Games Of 1984: In Review

By Jeff Hurlburt

This past year has been one of mixed blessing for those of us who like to spend time at our Apple's keyboard and joystick blasting aliens, mutants and robots to their makers, casting spells against evil enchantresses, fighting World War III, making megabucks on Wall Street or any of the other myriad of activities that are available through the wonder of computer games. On one hand we lost at least two major sources of information about new computer games (Softalk and St. Game) and one major game publisher (Sirius Software). On the more positive side of things, there probably have been more games for the Apple II released this past year than in any previous and, to top it off, the overall quality of new releases has been very good. Unfortunately, as a result of all this, it has become increasingly difficult for the value-conscious computer-gamer to decide where to spend his or her dollar (pound, franc, lira or yen as the case may be).

The purpose of the reviews presented in this issue of Hardcore COMPUTIST and the next is to assist you in making the best possible use of your computer gaming dollar. This month the reviews cover recently released adventure/fantasy games; next month, arcade-type games.

## The Reviews

Each review begins with a "game type" descriptor (such as "Picture-Text Adventure") and the number of players for which the game is designed. Compatibility, hardware requirements, options, and recommended peripherals, etc. are listed next. Lacking convenient access to an Apple IIc, I was unable to test games for IIc compatibility. Thus, only products so designated by the publisher are labeled as IIc compatible in the reviews. Game list price and publisher address and phone number are also provided in the event that you are unable to find the game you want in a local software shop.

## The Ratings

I've played each game listed in this review, and in many cases, watched others play it as well. The results of this observation appear towards the end of each review, following the game description.

Each game is rated in five areas and "overall" on a ten point scale:

10 = SUPERIOR 07 = GOOD  
05 = FAIR 03 = POOR  
"NA" means "NOT APPLICABLE"

"Graphics" (GRFX) is the first area rated. It rates quality of artwork, clarity, impact, smoothness, speed, and realism. Good "Support Materials" (S.M.) include clear, thorough directions for play. In some cases attractiveness, tutorial value, or effectiveness in creating 'atmosphere' may also be important.

"Playability" (PLAY) is determined by the amount of extraneous activity that the player must perform to play the game. Good parsing, rapid "save" and "restore" functions, efficient menus, smooth controls, and readily available "Help" screens are features which enhance playability.

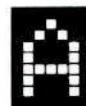
"Difficulty" (DIFF) is self-explanatory for single-player games. For others, it evaluates the difficulty in reaching a fairly high level of play. High "Interest" (INTR) games effectively attract and hold the player's attention. Typically, these are the adventures you can't wait to continue and the arcades you play, and replay, for hours at a time. The "Overall" (GAME) rating amounts to a summary of player reaction(s) during tryouts. For educational games, a second rating (/ED) of educational value is included.

As I mentioned earlier, the overall quality of computer games released this past year has been the best ever and the evaluations reflect this generally favorable climate. Even products which fare poorly here will appeal to some readers, and many of the more superior games will undoubtedly soon rank among your favorites.

## The Dallas Quest

(Picture-Text Adventure)

Compatible: 48K Apple II+, IIc, IIc  
Requirements: One disk drive  
KB input



famous detective, you have been retained by Sue Ellen to obtain the lost map which identifies the location of a juicy oil find in South America (it seems she has ideas of becoming financially independent of J.R.). Unfortunately J.R. overhears the conversation and determines to get the map himself. If you happen to stand in his way--well, J.R. has been known to play dirty when the chips are down.



The Dallas Quest takes you from the rooms and grounds of Southfork mansion to South American jungles complete with suspicious natives, lions, and a very helpful monkey. Graphics are well done and the parsing, though simple, is adequate. Humor and several neat little puzzles make The Dallas Quest an enjoyable evening's entertainment.

Available from: DataSoft, Inc., 19808 Nordoff, Chatsworth, CA 91311, (213) 701-5161. Cost: \$34.95.

GRFX	S.M.	PLAY	DIFF	INTR	GAME
07	05	09	05	07	07

## Enchanter (Text Adventure)

Compatible: 48K Apple II+, IIc  
Requirements: One disk drive  
KB input



lighted villages and the weakening of beneficial magic are but the more obvious signs of the evil Krill's growing power. Knowing that a full enchanter sent against the warlock would instantly attract his total forces, the Circle of Enchanters have summoned you. Only a rank novice, explains the venerable Belboz, might pass unnoticed through the evil one's domain. Even while searching the warlock's usurped castle for the means of his overthrow, your

aura will hardly distinguish you among the horde of half-human occupants.

Although **Enchanter** is set in the region about a mysterious flat-topped mountain, most of the action takes place in the warlock's castle. You are most concerned with acquiring spell scrolls and other useful artifacts and avoiding premature contact with Krill.

Each scroll confers new abilities which assist in unraveling the numerous puzzles (including one of the cleverest of maze contrivances). A rainbow-shelled turtle, sleepwalking ghost, demon worshippers, and a bothersome yet helpful adventurer highlight an amazing cast of characters.

Witty and very challenging, **Enchanter** is a superbly crafted adventure best shared by at least two players. Excellent parsing and fine 'color' documentation quickly involve you in one of the more entertaining, longer-playing gaming experiences available.

Available from: Infocom, Inc., 55 Wheeler St., Cambridge, MA 02138, (617) 492-1031. Cost: \$49.95.

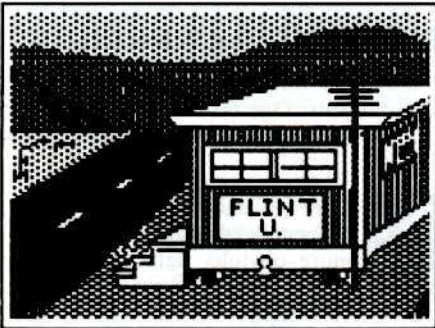
GRFX	S.M.	PLAY	DIFF	INTR	GAME
NA	09	08	08	09	09

## Expedition

### Amazon (Map Adventure)

**Compatible:** 48K Apple II+, //c  
**Requirements:** One disk drive  
 KB input

**D**etermined to prove that "ancient astronauts" were the source of Inca civilization, you and three colleagues set out to find and explore the lost city of Ka, located (you hope) somewhere in the Amazon jungle. At long last, it seems your dreams of exotic adventure and excitement are to be finally fulfilled.



**Expedition Amazon** employs ten 18x13 block, single-screen maps which, being unexplored, are blank. Moving onto a block maps it (turns it on) and, if you explore any region fully before leaving it, the map is yours for future use. If not, you lose it and

the screen will again show nothing should you return.

Assorted critters, traps, and hostile natives all take their toll, and it is necessary to make occasional visits downriver to Iquitos. There you may restock precious supplies and sell any artifacts your party has uncovered. Should a party member be lost, you leave Iquitos for Texas another.

Even at the lowest of nine difficulty levels, this is a very challenging game. Characters do become tougher and more skilled with experience but, unlike many such adventure games, they are never really "safe".

Although documentation skillfully sets the scenario, it leaves out some key elements of play, not the least of which is that pitfalls can be avoided if you respond quickly.

You will have to overlook Penguin's cheepo approach to packaging and actually play the game to see that **Expedition Amazon** is an exceptional adventuring value.

Available from: Penguin Software, 830 Fourth Avenue, PO Box 311, Geneva, IL 60134, (312) 232-1984. Cost: \$34.95.

GRFX	S.M.	PLAY	DIFF	INTR	GAME
07	05	07	09	08	08

## Infidel (Text Adventure)

**Compatible:** 48K Apple II+, //c  
**Requirements:** One disk drive  
 KB input

**Y**ou've conned the daughter of a deceased archeologist into making you head of an expedition to follow up on her father's potentially valuable find: a small cube, and the coordinates of its unearthing-- clues to the location of an unknown Egyptian queen's tomb. Now, after weeks of fruitless poking about, you awake with a mickey-induced hangover to a deserted camp. Nearby is a note penned by the chief of your admiring crew to the effect that you are on your own and, "May the jackals feed well upon your bones."

The first of Infocom's, "Tales of Adventure", **Infidel** is singularly difficult to 'get into'. Even given mirages, the Egyptian desert has to be one of the more uncomfortable, boring, and generally discouraging places in which to look for anything. When perseverance and the direct approach finally pay off, the relief of the tomb's coolness almost surpasses the excitement of your breakthrough. Suddenly there are more clues and portent-laden goodies than you can shake an ank at: the key, you soon discover, is to learn to decipher the numerous hieroglyphic

inscriptions.

Two heads are always better (and more fun) than one, but relentlessly logical arrangement makes this adventure more amenable to single player solution than most. Intriguing puzzles and Infocom's usual attention to detail in support materials make **Infidel** good for several evenings of first class adventuring fun.

Available from: Infocom, Inc., 55 Wheeler Street, Cambridge, MA 02138, (617) 492-1031. Cost: \$49.95.

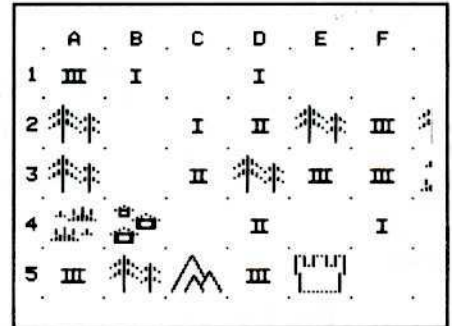
GRFX	S.M.	PLAY	DIFF	INTR	GAME
NA	09	08	07	08	08

## Lordlings

### Of Yore (Strategy-Combat Simulation for 1 to 4 players)

**Compatible:** 48K Apple II+, //c  
**Requirements:** One disk drive  
 KB input (printer recommended)

**B**orrowing heavily from such entrenched boardgame favorites as **Diplomacy** and **Risk** (and perhaps a bit from the computer classic **Santa Paravia & Fiumaccio**), **Lordlings of Yore** is set in the time of medieval legends. Magic works, but so do economics, good government, and armies. There are always four players with otherwise open positions being assumed by the computer.



As a shire lord, your objective is to take over the holdings of your rivals (each is nuisance and, besides, you need the added income). To this end, you build and deploy armies, make (and break) secret alliances, hire spies, and cultivate the good will of your subjects. Then too, the court necromancer is ever ready to cast a cogent spell (for a price, of course).

The game comes with copyable forms for recording positions (unnecessary if you select the printer option) and an illustrated instruction booklet. A unique blend, **Lordlings of Yore** is ideally suited to the hardcore strategy gamer who relishes involved conflicts which may span days

instead of hours.

Available from: Soflcore Corp., 8714 Wellesley Manor, San Antonio, TX 78240, (512) 691-2800. Cost: \$39.95.

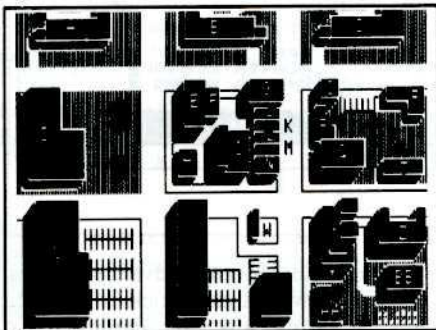
GRFX	S.M.	PLAY	DIFF	INTR	GAME
06	06	05	08	07	07

## Murder By The Dozen (Parlor Game for 1 to 4 players)

Compatibility: 48K Apple ][+, //e  
Requirements: One disk drive  
KB input

**Murder by the Dozen** belongs to the growing list of multi-player contests which challenge player intelligence more than specific game skills. In any one of twelve murder mystery scenarios, you strive to be the first to identify the murderer and, depending on scenario criteria, supply the proof to back up your claims, locate the missing money, etc. As you move about the city, question witnesses and look for clues, you quickly catch a scent and zero in on your prey. The program determines player turns according to time used, so make trips across town sparingly or your fellow sleuths may outwit you.

Clues and "no clues" are obtained at each location by selecting from among several search or questioning options. Clue numbers then appear on the screen and you must look up the text in the Clues Booklet.



Other game supplies include an answer booklet and note sheets complete with city map. In tryouts with several friends, **Murder** proved a consistent hit. This one has to rate as one of the genuine parlor game finds of the year.

Available from: CBS Software, One Fawcett Pl., Greenwich, CT 06836, (203) 622-2525. Cost: \$39.95.

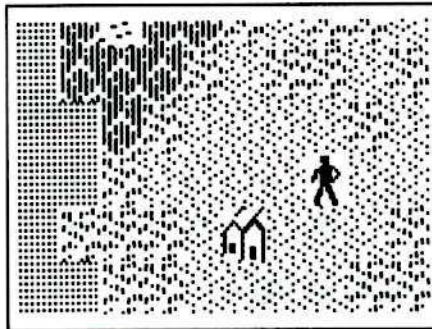
GRFX	S.M.	PLAY	DIFF	INTR	GAME
07	08	06	07	09	09

## Questron (Map Adventure)

Compatible: 48K Apple ][+, //e  
Requirements: One disk drive  
KB or Joystick



Wakened by insurrection and blighted by the machinations of the rogue wizard Mantor, **Questron** is a land on the brink of ruin. At just such a time, according to old legends, a mere peasant will rise to smash the evil at its source.



**Questron** looks and, in many respects, plays like a souped-up version of Ultima I. On the two super-sized continents ("Questron" and "The Land of Evil") you will find towns, castles, and cathedrals. Each location is different and each provides opportunities for trade, obtaining information, gambling, etc. The three dungeons and associated monsters are much like those of Ultima I, except the monsters move much faster. **Questron** leaves few loose ends, so you must penetrate each location in order to win.

Throughout play, your major concern remains character development. Strength, stamina, intelligence, etc. as well as hit points must be increased if you are to survive the perils of **Questron**. You will also be able to obtain more potent weapons and spells, better armor, and faster, safer transportation (**Questron's** equivalent of Ultima's aircar is the giant eagle). As to the question of a suitable challenge for the fully developed hero (who squashes surface monsters like bugs), fear not! When at last you break through the greatest of the dungeons and thence into Mantor's castle (itself a full-size map), your powers are tested to the limit.

Of the game's several innovations, the most notable is the use of on-screen menus. Besides permitting joystick control, the need to refer to game documentation is virtually eliminated. **Questron's** continent maps are rendered in an entirely new (vs. Ultima) hi-res character set. While aesthetic merit is debatable, the loss of detail is not. Hopefully, SSI will remedy this in any sequels. Finally, I was glad to see that, after several evenings spent in the service of the

empire, success was rewarded with due pomp and ceremony.

Of the several experienced players who tried out the game, all compare **Questron** favorably with Ultima III and at least one claims it is better. Certainly **Questron** is a fine entertainment value. As with the best of adventure games, it's one you'll be a bit sorry to win.

Available from: Strategic Simulations, Inc., 883 Stierlin Rd., Bldg. A-200, Mountain View, CA 94043, (800) 227-1617. Cost: \$49.95.

GRFX	S.M.	PLAY	DIFF	INTR	GAME
06	07	08	08	08	08

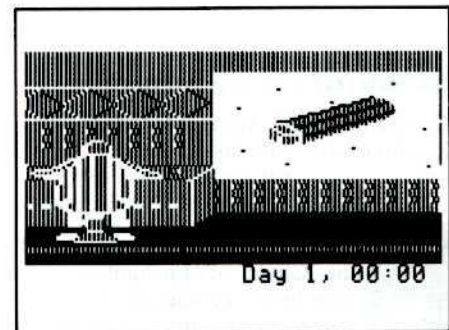
## Rendezvous With RAMA (Picture-Text Adventure)

Compatible: 64K Apple ][+, //e  
Requirements: One disk drive  
KB input



head of you, if your ship's sensors are to be believed, is the first genuine alien artifact encountered by humanity, a 50 km. long hulk code-named, "Rama". As captain of survey ship, Endeavor, your mission is to board and, if possible, probe the secrets of Man's first "Contact".

**Rendezvous With Rama** is among Telarium's first releases based upon popular print fiction. It is also one of the few to employ a 2:3 (scene:text) hi-res display, leaving ample room for the sometimes verbose descriptions. Alas, despite the promise of four diskette sides, **Rama** is not 'Infocom with pictures'. Incongruously, the game couples wordy prose and a standard-grade parser which can't even recognize most of the items mentioned.



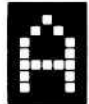
**Rama** does feature above average sound effects, efficient game save/restore functions, and several worthwhile puzzles. Add an interesting theme (plus imagination) and your rendezvous with Rama will be an entertaining venture.

Available from: Telarium, 1 Kendal Square, Cambridge, MA, 02139, (617) 494-1200. Cost: \$39.95.

GRFX	S.M.	PLAY	DIFF	INTR	GAME
07	05	07	06	07	07

## Seastalker (Text Adventure)

**Compatible:** 48K Apple ][+, //e  
**Requirements:** One disk drive  
KB input



As the inventor and director of an oceanic explorations group, you've just received an urgent call for assistance. Mysteriously, an enormous creature has appeared in the bay and attacked the Aquadome, your undersea research station.

**Seastalker's** scenario first involves using Scimitar, an experimental mini-sub, to reach the dome. Here you must foil sabotage attempts and outfit Scimitar for combat against all enemies.

Infocom bills **Seastalker** as "junior level" adventure for "ages 9 and up". Besides hint cards (readable only via a red filter "decoder"), designers supply the most structured scenario I've encountered in a game of this type. Often this results in the loss of initiative, a good deal of waiting for the program to catch up, and confusion as to who is really in charge. Having played through several adventures along with children, it's hard to believe that such an approach is necessary or even helpful.

Weak by Infocom standards, **Seastalker** is hardly a disaster. Indeed I've enjoyed the game since accepting that play is not so much a matter of solving problems as piecing together a script.

Available from: Infocom, Inc., 55 Wheeler Street, Cambridge, MA 02138, (617) 492-1031. Cost: \$39.95.

GRFX	S.M.	PLAY	DIFF	INTR	GAME
NA	08	07	06	06	07

## Sorcerer (Text Adventure)

**Compatible:** 48K Apple ][+, //e  
**Requirements:** One disk drive  
KB input



Belboz, the much esteemed dean of the Circle of Enchanters, has vanished! Considering that he is scheduled to address the 458th Enchanting Expo it seems clear that foul workings are afoot, and you alone must rescue your mentor.

Nominally the sequel to **Enchanter**, **Sorcerer** takes you from the cozy village with its quaint mail service to a curiously deserted landscape, and then into the depths

of the Great Underground Empire. With each new setting the demands on ingenuity and magical skill escalate, as do the available resources. In one of the more convoluted puzzles you are ever likely to encounter in any adventure, even time itself must be twisted.

Lavishly furnished with support goodies, **Sorcerer** is the richest and most challenging of Infocom's releases and I often found myself mulling for hours over some perplexing puzzle. Fortunately, the game's versatile parser makes it relatively easy to try out solutions.

Enormously fun, **Sorcerer** is simply the best game of its kind.

Available from: Infocom, Inc., 55 Wheeler Street, Cambridge, MA 02138, (617) 492-1031. Cost: \$49.95.

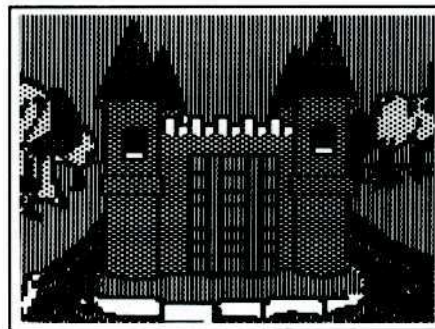
GRFX	S.M.	PLAY	DIFF	INTR	GAME
NA	09	08	09	09	10

## Sorcerer Of Claymorgue Castle (Picture-Text Adventure)

**Compatible:** 48K Apple ][+, //e  
**Requirements:** One disk drive  
KB input



True, you are not unskilled in magic, and the ancient fortress scarcely daunts an experienced adventurer. Yet, at this time of greatest challenge, some evil power has stolen the memory of the spells you carry.



In **Sorcerer of Claymorgue Castle**, #13 in the Scott Adams series, your quest is to recover the thirteen Stars of Power. **Claymorgue Castle** is rated "difficult" by its publishers and indeed it is, though not entirely for the best of reasons. Interesting puzzles, diabolical traps, and quality graphics more than balance the continued reliance on an antiquated two-word parser. Unfortunately, learning the use of your spells is bound to involve errors, most of which are fatal (along with practically every other mistake you make in this game). Coupled with the most cumbersome of

game restore routines, the resulting game comes just about as close to unplayability as a bootable diskette can get.

If you want to try this one and don't mind risking the forty quatlons, OK. Otherwise, don't bother.

Available from: Adventure International, PO Box 3435, Longwood, FL 32750, (305) 862-6917. Cost: \$39.95.

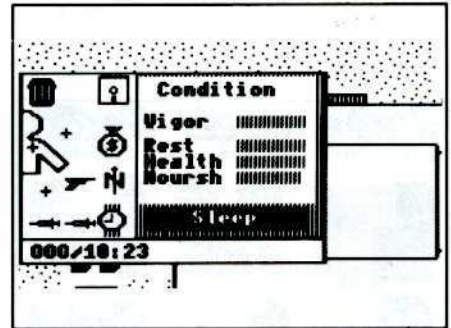
GRFX	S.M.	PLAY	DIFF	INTR	GAME
08	03	02	09	05	03

## Sundog (Map Adventure)

**Compatible:** 48K Apple ][+, //e  
**Requirements:** One disk drive  
Joystick



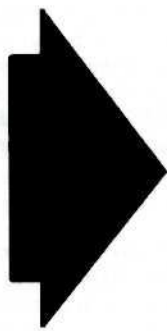
The untimely demise of your Uncle Brock has left you heir to the obligations of a ruinously tight contract and nominal owner of the cargo ship, **Sundog**. Now the half million credits advanced by the Society of the New Faith (a windfall for a crafty trader like Brock) weighs you like a millstone. Not only must you deliver needed materials to the society's new colony, but transport cryogenically preserved disciples ("cryogens") there as well. The latter's location, alas, did not survive your uncle's passing. Failure means virtual bondsman status but, if you succeed, the freedom of the galaxy is yours.



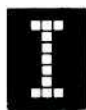
**Sundog** incorporates advanced hi-res screen windowing and joystick-only input. Thus, entering any of **Sundog's** six engineering bays (you use the joystick like a pointer) calls up the relevant status/choices window neatly superimposed on the ship display. Similarly, entering a building in any of some fifty cities produces a hi-res insert showing your character inside. Often, as when engaged in bargaining for supplies, withdrawing funds at the bank, etc. there will be two or more levels of windows with stick-selectable options.

Besides pirates and other deadly hazards a trader normally faces, you will confront

# The Chips Are In!



## 65C02 Chips Now Available



In Issue No. 16 we ran an article which we hoped would determine the number of readers interested in purchasing one of the 65C02 microprocessors we were to make them available. The response to that short note has been overwhelming.

In fact, so many people have called and written that we are not able to contact everyone, who has expressed an interest, by mail (as we had originally planned to do). For those who would like to place their order, the 65C02's (2MHz version from Western Design Center) are now available (please see the order form on page 30). With your order you will receive installation instructions and a data sheet.

Judging from some of the letters we have received, there appears to be some confusion as to the advantage a 65C02 offers over a standard 6502. *Here are some common questions we have received about the 65C02:* ▼

**Q** : If I install a 65C02 microprocessor in my Apple II Plus (or Apple compatible) will I be able to run software, like Appleworks, on my computer?

**A** : No, a 65C02 will not allow you to run such software. The main problem is that this software is designed to use the Apple IIe or IIc 80-column video and/or memory circuitry which is not available on a II, II Plus or compatible. The IIe and IIc also have extra routines in ROMs which are mapped into the \$C100-\$CFFF area of memory. These routines are not present in the older generation of Apples or in any of the Apple compatibles.

**Q** : Will a 65C02 let me address more than 64K of memory?

**A** : The 65C02, just like the 6502, can only address 64K of memory at one time. The Apple IIe and IIc employ bank-switching to allow them to address 128K of RAM.

At this time, the only 6502-compatible microprocessor that can address more than 64K is Western Design's 65SC816. This 16-bit CPU is code-compatible, but not pin-for-pin compatible with the 6502. Its close relative, the 65SC802, can only address the standard 64K of memory, but is both code-compatible and pin-for-pin compatible with the 6502.

Apple Computer's on-again, off-again Apple IIx computer is rumored to be designed around the 65SC816 microprocessor. Western Design has not actually gone into production of this chip yet because they are still working the bugs out of it. They are promising full-scale production of the 65SC16 and 65SC802 by April or May (almost a year and a half after it was originally promised).

**Q** : Does a 65C02 run faster than a 6502?

**A** : Both yes and no. The microprocessor that is plugged into the Apple's

powerful foes who stand to gain much should you default, so you will need to pack a laser and learn how to use it. Even a simple expedition to a nearby planet can involve several arcade type sequences.

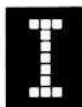
As might be expected of an adventure spanning 21 planets in 12 star systems, **Sundog** is not a single evening's entertainment. Happily, the menu inserts require little explanation, and an exceptionally well done manual fills any gaps. Challenging, colorful, and at the leading edge in adventure game design, **Sundog** ranks among the top entertainment values available.

Available from: FTL Games, 7907 Ostrow Street, Ste. F, San Diego, CA 92111, (617) 279-5711. Cost: \$39.95.

GRFX	S.M.	PLAY	DIFF	INTR	GAME
09	08	08	08	08	09

### Xyphus (Map Adventure)

**Compatible:** 48K Apple II+, IIe  
**Requirements:** One disk drive  
KB input



In some past (or future?) Earth, a mighty conqueror has defeated the forces of chaos until only the monster-infested land of Arroya

opposes his rule. You and your three comrades have volunteered to penetrate this dark region and dispatch Xyphus, arch demon and source of evil power.

Penguin's first swords-and-sorcery, Ultima-type venture features six successive scenarios complete with maps, trading posts, castles, and over 60 monster types. Characters may be fighters or magicians in one of three races: human, dwarf, elf.

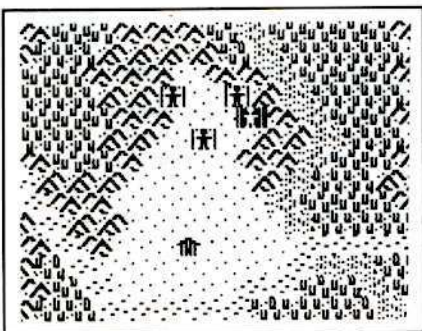
All movement and combat occurs on the 64x70 scrolling map. Hi-res blocks are 2x1 text characters in size so, with the half-block offset of alternating rows, movement is in six directions. All trading and other such transactions are handled in text.

In **Xyphus** your main activities include searching for exotic weapons, recovering assorted scrolls, and unearthing the precious xyphoid amulets (without which your spells are useless). Zapping monsters yields experience as well as gold for weapons and spells. A brief passage introduces each scenario outlining the situation and criteria for success.

Although I enjoyed **Xyphus**, it's hard to escape the feeling that it was plucked a bit early from the Penguin game tree. Opportunity for character development is minimal and the inability to move a party as a single unit (especially with variable movement factors) can drive you right up the wall.

Available from: Penguin Software, 830 Fourth Avenue, PO Box 311, Geneva, IL 60134, (312) 232-1984. Cost: \$34.95.

GRFX	S.M.	PLAY	DIFF	INTR	GAME
05	07	04	08	07	07



motherboard must run at a clock speed of 1 megahertz. A 6502 and a 65C02 will, therefore, run at the same clock speed. However, certain instructions on the 65C02 require fewer clock cycles than their counterparts on the 6502. As such, there may be a small, although probably not a noticeable, increase in the speed of execution of certain programs. Code written to take advantage of the 65C02's additional machine language instruction set, would run faster than equivalent code written for a 6502.

**Q**: What is the "enhanced instruction set" of the 65C02 and will it add new commands to Applesoft?

**A**: The 65C02 has a total of 178 functional op codes, versus 151 on the 6502. These additional opcodes come from 8 new machine language instructions (or 10 new instructions, depending upon what you consider to be a new instruction) and the extension of new addressing modes to existing instructions. Tables 1 and 2 show the new instructions and additional instruction addressing modes available on the 65C02.

Other advantages of the 65C02 are its low power consumption, the default of undefined op codes to 1 cycle NOP's (rather than having unpredictable results as when executed on a 6502) and the correction of the indirect JMP bug.

Sorry, but the addition of a 65C02 microprocessor to your computer will not enhance Applesoft BASIC by giving it any new commands. You will just have to learn assembly language in order to take advantage of the 65C02. Maybe installing a 65C02 in your computer will provide you with a good excuse to learn assembly language.

**Q**: Can I expect to have any hardware or software compatibility problems if I install a 65C02 in my computer?

**A**: As far as we know, there are no programs written for the Apple that will not execute on a computer with a 65C02 because the new microprocessor has no problem executing standard 6502 code. It is, however, possible to write timing-dependent 6502 code that cannot execute properly on a 65C02 because of the differences in the number of cycles that certain instructions require. Although we have not seen any code of this sort, no doubt someone's obscure copy-protection method relies upon it. Also, because the indirect JMP bug has been fixed, an instruction of the form, JMP (\$XFFF), does not execute the same on a 65C02 as a 6502.

There are some rumors of 65C02 hardware compatibility problems (we have not actually seen any), namely Apple ][ or ][ Plus's with 65C02's and certain 80-column video cards. We have tested the 65C02 in Apple ][ Plus's and Franklin Aces with both Videx and

Franklin 80-column cards and have yet to experience any problems, but we want to make our readers aware of this potential problem. Apple ][e owners should have no problems as long their 80-column is in the auxiliary slot.

**Q**: What are the advantages in upgrading to a 65C02?

**A**: Unless you program in assembly language or plan to learn assembly language, installing a 65C02 in your Apple or compatible will do you little good (unless you like to play one-up-manship with your fellow user group members). For assembly language programmers, the main advantage will be the ability to write more compact, faster running code.

By the time you read this, Apple Computer will probably have released an upgrade kit for the ][e (price about \$70) that contains a 65C02 and three new ROMs (new character ROM and new CD and EF ROMs). This upgrade will make the ][e more-or-less compatible with the ][c by giving the ][e mouse icons and better interrupt handling. Other improvements are said to include a fix for the jagged 80-column scrolling, the

ability of Applesoft and the monitor to accept commands in upper or lower case, and putting the mini-assembler back into place. The changes will eventually be included in all new ][e's that are sold. Since all the ROMs in the ][e are socketed and completely compatible with standard EPROMs, manufacturers of EPROM burners will probably soon be noticing an uptick in their sales figures.

We have purchased the 65C02's that we will be offering from Western Design Center because we understand that the chips from this source eliminate the need to substitute 74F257's for the 74LS257's at locations B6 and B7 on the motherboard. As such, we will not be offering the 74F257 chips for sale at this time. We have tried the 65C02's in several Apple ][ Plus's and have yet to run across a problem, although that is no guarantee that someone out there will not have one. If you find that you absolutely need the 74F257's, you can order them from Jameco Electronics for \$1.79 each (Jameco Electronics, 1355 Shoreway Road, Belmont, CA 94002, phone (415) 592-8097). If you want more information on why this substitution is sometimes needed (mainly for

C  
O  
R  
E

### New Instructions

Instruction	Op Code	Description
BRA	80	Branch Always (Relative)
PHX	DA	Push X on Stack
PHY	5A	Push Y on Stack
PLX	FA	Pull X from Stack
PLY	7A	Pull Y from Stack
STZ	9C	Store a Zero in memory (Absolute)
STZ	9E	Store a Zero in memory (Absolute,X)
STZ	64	Store a Zero in memory (Zero page)
STZ	74	Store a Zero in memory (Zero page,X)
TRB	1C	Test and reset bits with Accum. (Absolute)
TRB	14	Test and reset bits with Accum. (Zero page)
TSB	0C	Test and set bits with Accum. (Absolute)
TSB	04	Test and set bits with Accum. (Zero page)

### Additional Instruction Addressing Modes

Instruction	Op Code	Description
ADC	72	Add memory to Accum. w/Carry (Zero Page)
AND	32	AND memory with Accum. (Zero Page)
BIT	3C	Test memory bits with Accum. (Zero Page,X)
BIT	34	Test memory bits with Accum. (Zero Page,X)
CMP	D2	Compare memory with Accum. (Zero Page)
EOR	52	Exclusive OR with Accum. (Zero Page)
JMP	7C	JUMP (Absolute (Indirect,X))
LDA	B2	Load Accum. with memory (Zero Page Indir.)
ORA	12	OR memory with Accum. (Zero Page)
SBC	F2	Subtract from Accum. w/borrow (Zero Page)
STA	92	Store Accum. in memory (Zero Page)

65C02's manufactured by Rockwell International), see the article on page 15 of the December 1984 issue of Apple Assembly Lines.

We would like to thank everyone who has expressed their interest in the 65C02. If there are other chips that you would like to see us offer (the 65SC802 and 65SC816, when they become available, EPROM's, RAM's, etc.), drop us a line and let us know. If there is sufficient interest in a particular chip, we will look into offering them for sale, also.



## Bugs in Hardcore COMPUTIST No.'s 16 & 17

### Issue No. 16, pg. 10

The XFER Boot Source code (bottom of pg. 10) contains errors which are corrected by substituting the following lines for those originally printed:

```
0306:8D 5E C0 STA ANN3.OFF DBL HIRES OFF!
0310:8D EE 03 STA XFER.ADR+1 STORE MSB
```

### Issue No. 17, Contents pg. and page 13

The name of the co-author of, "The Print Shop: A Softkey" was inadvertently misspelled. We would like to credit the article to Mr. William Hinger and Mr. Albert Stocker. Our sincere apologies.

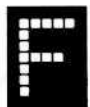
### pg. 15

The space (originally indicated by a delta character) at the end of Line 5000 of the Print Shop Controller should be omitted as follows:

```
5000 DATA 5*CHANGES
```

## APT's

### Championship Loderunner



First, start the game in the usual manner and then commit suicide until you have one man left. Save the game, then load your saved game. Commit suicide one more time to end the game. Now check your saved game. It should show that you have no men. Load this game again and commit suicide. You should now have 254 men!

Contributed by S. Scott.

## ADVENTURE TIPS



Hardcore COMPUTIST is always looking for adventure hints to any of the popular adventure/fantasy games sold for the Apple II, II+ or IIe. We prefer that these hints not be dead give-aways to dilemmas encountered in the game, but instead contain just enough information to help the stumped adventurer on his way.

Send your ADVENTURE TIPS on a 3 x 5 postcard to the address below, and watch for them to appear in this column in upcoming issues:

**Hardcore COMPUTIST**  
Att. Adventure Tips  
PO Box 110846-K  
Tacoma, WA 98411.

And DON'T FORGET to include the name *and manufacturer* of the adventure game to which your hint pertains.

### \* Pirate Adventure Scott Adams Adventures

Mongoose?? Are you sure??

### \* Planetfall Infocom, Inc.

Pronunciation's the key here. Read the signs just as they look.

\* Contributed by Kate Shuster

### # Enchanter Infocom, Inc.

The cell with the dungeon is old and crumbling, especially the wall. Could there be loose bricks?

Rat holes are good places to hide things, such as scrolls.

Got a shredded scroll that you can't read?

Check the forest for an answer.

The "Long Road" is very, very long.

A turtle's shell can deflect spears.

\* Contributed by Mike Kao

### ‡ Mask Of The Sun Ultrasoft

A very long road will only lead to trouble. Make sure to take your medicine before doing any heavy lifting.

Don't trust men that sit on tree stumps.

‡ Contributed by Jeff Beard

## ADVENTURE TIPS

### Wizard and the Princess Sierra On-Line

Try throwing a rock at the snake to get past, but watch out for scorpions.

Still having trouble? Get the stick from the desert to prod him out of his burrow. To get across the chasm you'll need a little magic. Look carefully at the note (two halves) for the clue.

That parrot isn't just a bird-brain! Look for a tasty tidbit to give to him.

The gnome is sure a pesty devil. If your stuff is still missing, free the snake back in the desert. He'll have a clue for you.

### Death In The Caribbean Microlab

In the dark? If you want to get past the trapdoor, you'll need some light.

Check out the organ for reading material. Getting down a cliff isn't just child's play. Go back and get the swing rope.

Moving furniture can be tricky, but a wagon isn't hard at all. Just tie the rope to it and throw it over the cliff's edge.

### Zork II Infocom, Inc.

Chilled by the wall of ice? A dragon's breath just might do the trick. Hit him with your sword a few times and he'll follow you wherever you want.

Be polite to the princess. She may reward you with an important clue.

Light the newspaper to provide a source of hot air for the balloon.

Now you can't find the brakes on your balloon? Try reading the instructions (or just douse the fire).

### Cranston Manor Sierra On-Line

No key is necessary to open the gate. Just pry around in the junkyard until you find a likely tool.

Can't figure out the basement door? The shape of the lock is the key to your dilemma.

I hope you're not afraid of heights because the only way out of the bridal suite is to go over the edge.

Your adversary may be *bullish* in the light of day, but put him in the dark and he's nothing to be concerned about.

# CHECKSOFT

V2

By  
Ray  
Darrah

(No. 1, page 11)

The very first program printed in the very first issue of Hardcore COMPUTIST was an assembly language program called CHECKSOFT, written by Robb Canfield. CHECKSOFT was, and still is, the program that generates the series of four-digit (or is it hexit?) hexadecimal numbers that accompany all Applesoft BASIC programs that are printed in our magazine. These

C  
O  
R  
E

"checksums", one for each line of the BASIC program, are designed to help our readers track down the typing errors that occur when they manually enter the programs into their computers. Checksoft updates its checksum value for every byte in each line of a BASIC program and displays the current number when it has finished with that line number. Since the checksum is cumulative, all the previous checksums generated affect the checksum values which follow it. This makes it much easier to spot the lines of a program which contain typing errors.

## The Problem

Unfortunately, the original Checksoft program wasn't very friendly. For example, if you discovered an incorrect checksum in line 500 of a program and you attempted to fix it, you would have to view all the previous checksums before seeing if your correction worked.

A second, and much more irritating, problem with Checksoft occurs if you just cannot get a correct checksum for a certain line. Due to the cumulative nature of the checksum values, this single incorrect line will prevent you from using the checksums to find out if the remainder of your program is correct or not.

## The Cure

After extensive modifications to the original Checksoft program, I have created Checksoft v2. In Version 2, the checksums are calculated in the same manner as did the original program; therefore, the same checksums will result when you use either of the two checking methods.

## Using Checksoft v2

Both versions of checksoft are accessed via

Applesoft's Ampersand command (&). To generate the checksum, you first type in the program to be checked and then activate Checksoft by typing

```
BRUN CHECKSOFT
&
```

The major improvement of Checksoft v2 is that two parameters can be passed to the program through the ampersand command. The new syntax of Checksoft v2 is:

```
&linenum,seed
```

Linenum, the comma, and seed are optional. If you just type an ampersand, the checksums will be generated in the same fashion as the original Checksoft.

Linenum is a number corresponding to the starting line of the checksums you wish to display. For example, if you type:

```
&500
```

**"Checksoft...is the program that generates the series of four-digit hexadecimal numbers that accompany all Applesoft BASIC programs that are printed in Hardcore COMPUTIST. The major improvement of CHECKSOFT v2 is that two parameters can be passed to the program through the ampersand command."**

then the first checksum you would see would be that for line 500. The linenum parameter is handy if you want to quickly see if your correction worked or if you just want to see if the final checksum value of a program is correct.

Seed is a four digit (one, two and three digits will not work) hexadecimal number that sets the checksum to that value before proceeding at the line number specified by linenum. For instance: suppose that you are tired of trying to get the checksum for line 250 to match up with the value that is printed in the magazine and would like to check the remainder of the program for accuracy. Let's say that the magazine indicates line 250 should have a checksum value of \$20CD. To check the program from line 260 onward you would type

```
&260,20CD
```

therefore bypassing whatever error happens to be in line 250.

## Typing It In Using Checksoft v2

Use the procedure outlined in the How-To's of Hardcore (adjacent to the Contents page) to key in the following hexdump and save it with:

```
* BSAVE CHECKSOFT,AS2E7,LSE9
```

The source code for Checksoft v2 begins on this page.

### Checksoft Source Code

```
*****
*
*          CHECKSOFT V2
*          BY
*          ROBB S. CANFIELD
*          AND RAY DARRAH
*
*          COPYRIGHT 1985
*          SOFTKEY PUBLISHING
*
*****

*
* LOCATIONS USED
*
ED24- LINPRNT .EQ $ED24 PRINT DECIMAL VALUE OF X AND A REG
FD8E- CROUT .EQ $FD8E PRINT A CARRIAGE RETURN
FD8E- COUT .EQ $FD8E PRINT A REG IN ASCII
F941- PRINTX .EQ $F941 PRINT A AND X REG AS A TWO HEXBYTE NUMBER (A WORD)
0067- BEG .EQ $67 BEGINNING OF APPLESOFT PROGRAM
```

Continued on next page



To use Checksoft v2, type in or LOAD the Applesoft program you want to check, type BRUN CHECKSOFT and start the checksum process by entering an ampersand character followed by its optional parameters.

One thing to note about the checksums that are printed in Hardcore COMPUTIST is that they are generated by the standard configuration of the program, which ignores any characters that appear following REM statements. This means that you must type in all REM statements that appear in a program, but you do not have to type in any of the programmer's comments which follow the REM. It is possible to configure Checksoft so that it will calculate the checksums using the comments following REM's or even have it exclude line numbers from the checksum calculation, but we do not recommend that readers do this since we always use Checksoft in its standard configuration to generate the checksums which appear in the magazine.

### Checksoft v2 Hexdump

```

02E7: A9          $6533
02E8: 4C 8D F5 03 A9 F7 8D F6 $16BD
02F0: 03 A9 02 8D F7 03 60 20 $0C74
02F8: 0C DA A2 14 86 0A A6 67 $D055
0300: 8E A5 03 86 0B A6 68 8E $D840
0308: A6 03 86 0C AA F0 14 20 $C61F
0310: 1A D6 A5 9B 8D A5 03 A5 $4598
0318: 9C 8D A6 03 20 A8 03 E6 $AF33
0320: B8 A2 00 86 FF E0 02 90 $8B3A
0328: 21 E0 04 B0 03 48 90 0D $2168

0330: A8 F0 1D A4 FF 30 13 C9 $9091
0338: B2 D0 02 85 FF 6A 45 0B $69B5
0340: 2A 45 0C 85 0B 45 0C 6A $7217
0348: 85 0C 20 9C 03 E8 D0 D5 $DCB5
0350: 68 A8 68 AA 9B C5 51 D0 $65B9
0358: 02 E4 50 90 2F 20 24 ED $98BC
0360: A0 04 B9 CB 03 20 ED FD $4594
0368: 8B 10 F7 A5 0B A6 0C 20 $6643
0370: 41 F9 20 8E FD C6 0A D0 $3E0F
0378: 13 A9 14 85 0A 20 8E FD $4A66

0380: AD 00 C0 10 FB 8D 10 C0 $A07E
0388: C9 83 F0 0D A2 00 20 9C $3D1C
0390: 03 D0 90 E8 20 9C 03 D0 $2B8C
0398: 8A 4C 8E FD EE A5 03 D0 $4941
03A0: 03 EE A6 03 AD FF FF 60 $A6A7
03A8: A2 00 86 0B 86 0C 20 B2 $6647
03B0: 03 E8 20 BB 03 0A 0A 0A $E6D0
03B8: 0A 95 0B 20 B1 00 49 30 $6363
03C0: C9 0A 90 02 E9 67 15 0B $879A
03C8: 95 0B 60 A4 A0 AD A0 A0 $CD7A

```



### CHECKSOFT SOURCE CODE (Continued from previous page)

```

0024- HTAB .EQ $24 HORIZONTAL POSITION OF CURSOR
C010- CLEAR .EQ $C010 CLEAR KEYBOARD STROBE
C000- READ .EQ $C000 LOAD A REG WITH KEY PRESSED
000A- COUNTER .EQ $0A COUNTER FOR NUMBER OF LINES
000B- CHKSUM .EQ $0B CHECKSUM BYTES $0B & $0C
03F5- AMPER .EQ $3F5 AMPERSAND JUMP VECTOR
0050- LINNUM .EQ $50 WHERE BASIC STORES LINE NUMBER
009B- LOWTR .EQ $9B POINTER TO BASIC FOUND LINE
D61A- FNDLIN .EQ $D61A BASIC ROUTINE THE FINDS SPECIFIED LINE NUMBER
00FF- REM.FLG .EQ $FF BIT 7 = CURRENTLY IGNORING A REM?

.OR $2E7 PLACE PART ON PAGE 2 AND REST ON PAGE 3
.TF CHECKSOFT

```

```

*
* INITIALIZATION FOR AMPERSAND
*

```

```

02E7: A9 4C INIT LDA #$4C SET UP AMPERSAND JUMP VECTOR
02E9: 8D F5 03 STA AMPER
02EC: A9 F7 LDA #SETUP
02EE: 8D F6 03 STA AMPER+1
02F1: A9 02 LDA /SETUP
02F3: 8D F7 03 STA AMPER+2
02F6: 60 RTS

```

```

*
* PROGRAM STARTS HERE
*

```

```

02F7: 20 0C DA SETUP JSR $DA0C GET STARTING LINE NUMBER
02FA: A2 14 LDX #20 # OF LINES UNTIL STOPPED
02FC: 86 0A STX COUNTER
02FE: A6 67 LDX BEG GET BEGINNING OF APPLESOFT PROGRAM
0300: 8E A5 03 STX GET+1
0303: 86 0B STX CHKSUM USE THIS BYTE FOR CHECKSUM
0305: A6 68 LDX BEG+1
0307: 8E A6 03 STX GET+2
030A: 86 0C STX CHKSUM+1
030C: AA TAX NO SEED VALUE?
030D: F0 14 BEQ LOOP1 YUP

030F: 20 1A D6 JSR FNDLIN MAKE LOWTR POINT TO FIRST LINE
0312: A5 9B LDA LOWTR FIX GETTER
0314: 8D A5 03 STA GET+1
0317: A5 9C LDA LOWTR+1
0319: 8D A6 03 STA GET+2
031C: 20 A8 03 JSR GET.SEED PUT THE SEED VALUE IN
031F: E6 B8 INC $B8 NO SYNTAX ERROR!
0321: A2 00 LDX #0 SET COUNTER TO ZERO
0323: 86 FF STX REM.FLG CLEAR REM FLAG
0325: E0 02 CPX #2 ON OFFSET?
0327: 90 21 BCC SUMIT3 YES, SO SKIP END OF LINE CHECK
0329: E0 04 CPX #4 ON LINE #?
032B: B0 03 BCS SUMIT NO, SO DON'T SAVE LINE #
032D: 48 PHA SAVE LINE #
032E: 90 0D BCC SUMIT2 SKIP END OF LINE CHECK. DEFAULT = USE LINE #
0330: A8 TAY END OF LINE?
0331: F0 1D BEQ PRINTCHK YES, PRINT CHCKSM & CHECK FOR END OF PROG.
0333: A4 FF LDY REM.FLG IGNORE REM?
0335: 30 13 BMI SUMIT3 YES
0337: C9 B2 CMP #$B2 TOKEN FOR REM?
0339: D0 02 BNE SUMIT2
033B: 85 FF STA REM.FLG SET REM FLAG. DEFAULT IS TO IGNORE THEM
033D: 6A ROR
033E: 45 0B EOR CHKSUM ;GENERATE A CHECKSUM
0340: 2A ROL
0341: 45 0C EOR CHKSUM+1
0343: 85 0B STA CHKSUM
0345: 45 0C EOR CHKSUM+1
0347: 6A ROR
0348: 85 0C STA CHKSUM+1
034A: 20 9C 03 SUMIT3 JSR INCGET INCREMENT COUNTERS
034D: E8 INX
034E: D0 D5 BNE LOOP2 CONTINUE BY GETTING ANOTHER BYTE
0350: 68 PRINTCHK PLA GET HIGH BYTE OF LINE #

```

Continued on next page

# Most Wanted List

If you have been trying to backup a program, and have only ended up pulling your hair out as a result of the ordeal, let us know about it. We will include it in our Most Wanted List. Send requests to:

**Hardcore COMPUTIST  
Wanted List  
PO Box 110846-K  
Tacoma, WA 98411**

If you know how to de-protect, unlock or modify any of the programs below, we encourage you to help other Hardcore COMPUTIST readers and earn some EXTRA MONEY at the same time. Send the information to us in article form on a DOS 3.3 diskette.

1. **Apple Business Graphics**  
*Apple Computer*
2. **Flight Simulator II**  
*Sub Logic*
3. **Apple LOGO II**  
*Apple Computer*
4. **DB Master 4.0+**  
*Stoneware, Inc.*
5. **Bookends**  
*Sensible Software*
6. **Visiblend**  
*Micro Lab*
7. **Dollars And Sense**  
*Monogram*
8. **Lifesaver**  
*Micro Lab*
9. **Catalyst**  
*Quark, Inc.*
10. **Gutenberg Jr. & Sr.**  
*Micromation LTD*
11. **Prime Plotter**  
*Primesoft Corp.*
12. **SSI Wargame Series**  
*Strategic Simulations, Inc.*
13. **Robot Odyssey**  
*The Learning Company*
14. **Zardax**  
*Computer Solutions*
15. **The Listers**  
*Silicon Valley Systems*
16. **Milliken Math Series (NEW)**  
*Milliken Publishing*
17. **College Entrance Exam Prep**  
*Borg Warner*
18. **Bank Street Speller**  
*Broderbund*
19. **Karateka**  
*Broderbund*
20. **Microzine Series**  
*Scholastic*
21. **Report Card**  
*Sensible Software*

Continued from previous page

```

0351: A8          TAY          SAVE IT TEMPORARILY
0352: 68          PLA          GET LOW BYTE OF LINE #
0353: AA          TAX          SET UP FOR LINE PRINT
0354: 98          TYA          GET HIGH BYTE BACK
0355: C5 51       CMP LINNUM+1 PRINTING CHECKSUMS
0357: D0 02       BNE NOMAT.MSB YET? IF GREATER THAN LINNUM
0359: E4 50       CPX LINNUM   LSB TOO
035B: 90 2F       NOMAT.MSB BCC NOSTOP

035D: 20 24 ED     JSR LINPRINT PRINT DECIMAL LINE #
0360: A0 04       LDY #4      # OF CHARS. TO PRINT MINUS 1 (-1)
0362: B9 CB 03     LDA MSG1,Y  PRINT1
0365: 20 ED FD     JSR COUT    PRINT AREG IN ASCII
0368: 88          DEY         DONE WITH PRINT?
0369: 10 F7       BPL PRINT1  NO. SO GET NEXT CHARACTER
036B: A5 0B       LDA CHKSUM  PRINT CHECKSUM VALUE
036D: A6 0C       LDX CHKSUM+1
036F: 20 41 F9     JSR PRINTAX
0372: 20 8E FD     JSR CROUT
0375: C6 0A       DEC COUNTER DONE WITH 20 LINES
0377: D0 13       BNE NOSTOP  NO, SO CONTINUE
0379: A9 14       LDA #20     YES, SO GET KEYPRESS
037B: 85 0A       STA COUNTER AND RESET COUNTER
037D: 20 8E FD     JSR CROUT
0380: AD 00 C0     LDA READ    GET A KEYPRESS
0383: 10 FB       BPL KEY1    LOOP UNTIL A KEY IS PRESSED
0385: 8D 10 C0     STA CLEAR   CLEAR KEYBOARD STROBE
0388: C9 83       CMP #583    WAS IT CTRL C?
038A: F0 0D       BEQ END     YES, SO STOP PROGRAM
038C: A2 00       LDX #0      OFFSET ZERO
038E: 20 9C 03     JSR INCGET  GET NEXT BYTE
0391: D0 90       BNE LOOP1   IF NO ZERO WAS ENCOUNTERED CONT.
0393: E8          INX         NEXT POS
0394: 20 9C 03     JSR INCGET  AND BYTE
0397: D0 8A       BNE LOOP1   IF NO ZERO
0399: 4C 8E FD     JMP CROUT   DONE WITH PROGRAM

```

\*  
\* SUBROUTINES CALLED FROM MAIN PROGRAM  
\*

```

039C: EE A5 03     INCGET     INC GET+1    ADD ONE TO THE VALUE IN GET
039F: D0 03       BNE GET    IS THERE A CARRY OVER
03A1: EE A6 03     INC GET+2  YES INCREMENT HIGH BYTE
03A4: AD FF FF     GET        LDA $FFFF    GETS A VALUE FROM APPLESOFT
03A7: 60          RTS

03A8: A2 00       GET.SEED   LDX #0        FIRST IS LSB
03AA: 86 0B       STX CHKSUM
03AC: 86 0C       STX CHKSUM+1
03AE: 20 B2 03     JSR GETNUM GET IT
03B1: EB          INX
03B2: 20 BB 03     GETNUM     JSR GETDIG    GET A DIGIT
03B5: 0A          ASL        MSN FIRST
03B6: 0A          ASL
03B7: 0A          ASL
03B8: 0A          ASL
03B9: 95 0B       STA CHKSUM,X
03BB: 20 B1 00     GETDIG     JSR $B1      GET NEXT CHARACTER
03BE: 49 30       EOR #$30   MAKE REAL
03C0: C9 0A       CMP #8A    DIGIT YET?
03C2: 90 02       BCC ORING  YUP!
03C4: E9 67       SBC #$67   NOW IT IS REAL
03C6: 15 0B       ORING      ORA CHKSUM,X
03C8: 95 0B       STA CHKSUM,X
03CA: 60          RTS

```

\*  
\* PRINT MESSAGES  
\* THE MESSAGE IS STORED BACKWARDS  
\*

```

03CB: A4 A0 AD     MSG1      .AS -" - "
03CE: A0 A0

```



# Simple Copy Protection

By Rohn Smith

## Requirements:

- Apple II series of computer
- Standard DOS 3.3 disk
- Sector editor that can access track \$23
- Disk containing files to be protected

There is one location on a normal 3.3 disk that gets overlooked by the standard Apple copy programs (and most people using nibble copiers) is track \$23. Apple DOS does not use this track, even though it is accessible on Apple drives. This makes track \$23 a great place to put hide something. What could you put there? Why not the VTOC (Volume Table Of Contents) and CATALOG, since no DOS operation will work without first accessing these (the VTOC tells DOS where to find the beginning of the catalog and which sectors on the disk are free for storage).

Before the VTOC and CATALOG can be moved to track \$23, a disk with this track initialized will have to be made. Use the following patch so that DOS will INIT all \$24 tracks instead of the usual \$23, and you then can make use of track \$23 (track \$23 is the \$24th track on the disk- the first track is \$00).

1) Boot up with a DOS 3.3 disk

**PR#6**

2) Remove the disk you booted with and insert a blank disk in its place.

3) Enter the Apple's monitor by typing

**CALL -151**

4) The following patch to the INIT command handler will initialize track \$23

**BEFE:24**

5) Initialize the disk by typing

**INIT HELLO**

6) Next use FID (on the DOS 3.3 Master Disk) or another file transfer program to transfer all the files you want to protect over to the disk you just initialized.

7) Use a sector editor to make the following change to the disk to be protected, so that DOS can find the VTOC on track \$23

Track	Sector	Byte	From	To
\$01	\$08	\$01	\$11	\$23

Now you must move track \$11 (VTOC and directory) to track \$23, make some changes to the relocated VTOC and directory sectors and fill track \$11 with zeroes. Now, when CATALOGed from normal DOS, the disk will appear empty. You can choose one of two ways to accomplish this.

*\*\* see N629, page 5 (letter) further uses*

The first method requires a sector editor that can access track \$23 (such as Tricky Dick). With it, you must move track \$11 to track \$23, sector by sector, making some changes to the sectors along the way. Empty sectors also have to be written to track \$11. This method is tedious, but it gets the job done. To use this method, proceed to "Sector Edit Method".

## The VTOC Mover

The second method requires that you type in an assembly language program which will relocate the VTOC to track \$23, make the necessary alterations and also blank out track \$11. Essentially, it will do everything that you would have done manually with a sector editor.

To use this program, type in the hexdump on page 26 and save it to disk (not the protected disk!) with

**BSAVE VTOCMOVER,AS300,LS93**

This simple program basically just puts the necessary values into the IOB (Input/Output Block) and uses a routine in the RWTS, at \$B793, to read or write a range of pages (sectors). The first instruction of this program is an RTS (\$60) which prevents the program from doing anything to a disk that it is BRUN from. The program also checks to make sure that a disk's VTOC has not

already been relocated. If it were run on a disk more than once, that disk would be left with no VTOC or directory. That is probably a bit more protection than anyone would want.

First BLOAD VTOCMOVER, insert the disk you wish to protect into the drive and then type CALL 769 (or 301G from the monitor). The disk drive will come on momentarily and that will be it. Try to CATALOG the disk. All that you should see is a DISK VOLUME 254 message. To see the real catalog, do a POKE -21503,35 (AC01:23 from the monitor) before typing CATALOG. Do a POKE -21503,17 to restore this location so you can CATALOG disks with their VTOC in the normal spot.

## Sector Edit Method

Probably, only people with an aversion to typing and/or programming will make use of the Sector Edit method. Just remember that you need a sector editor that can access track \$23.

1) With your sector editor, read in track \$11, sector \$0 and make the following changes

Track	Sector	Byte	From	To
\$11	\$00	\$01	\$11	\$23
\$11	\$00	\$30	\$23	\$24
\$11	\$00	\$7C	\$00	\$FF
\$11	\$00	\$7D	\$00	\$FF

## VTOC Mover Source Code

```

*-----*
* PROGRAM TO MOVE A DISK'S VTOC FROM TRACK $11 TO TRACK $23 FOR *
* PROTECTION. ALSO ZERO'S OUT TRACK $11 AND FIXES THE MOVED VTOC. *
*-----*
B7E8- IOB .EQ B87E8 ; INPUT/OUTPUT BLOCK.
B793- RW.PAGES .EQ B8793 ; ROUTINE TO R/W SEVERAL PAGES.
B7E1- NUM.PAGES .EQ B87E1 ; NUMBER OF PAGES TO R/W.
1000- BUFF .EQ 10000 ; BUFFER ON PAGES $10-$1F.
00FE- PTR .EQ $FE ; TEMPORARY POINTER.
*
*
* .OR $300
* .TF VTOCMOVER
*
*-----*
* READ TRACK $11 *
*-----*
0300: 60 RTS ; TO BE SAFE.
0301: A9 11 READ11 LDA #$11 ; READ TRACK
0303: 8D EC B7 STA IOB+4 ; $11.
0306: A9 01 LDA #$01 ; CODE FOR READ.
0308: 8D F4 B7 STA IOB+$C ; STORE IT IN IOB.
030B: 20 79 03 JSR SETIOB ; SET UP THE IOB.
030E: 20 93 B7 JSR RW.PAGES ; GO READ TRACK $11.
0311: AD 01 10 LDA BUFF+1 ; MAKE SURE THAT
0314: C9 11 CMP #$11 ; VTOC WAS ON
0316: D0 60 BNE DONE ; TRACK $11.
*
*-----*
* FIX THE DIRECTORY LINKS *
*-----*
0318: A9 00 FIXPTRS LDA #BUFF ; FIX THE
031A: 85 FE STA PTR ; LINKS
031C: A9 1F LDA /BUFF+$F00 ; IN THE

```

Continued on next page

Continued from previous page

```

031E: 85 FF          STA PTR+1    ; DIRECTORY
0320: A2 0E          LDX #0E     ; SECTORS
0322: A0 01          LDY #01     ; SO THAT
0324: A9 23          LDA #23     ; THEY ARE
0326: 91 FE          STA (PTR),Y ; CORRECT.
0328: C6 FF          DEC PTR+1
032A: CA            DEX
032B: D0 F9          BNE .1      ; DONE?
032D: A9 24          LDA #24     ; FIX # OF
032F: 8D 34 10       STA BUFF+$34 ; TRACKS.
0332: A9 FF          LDA #FF     ; FREE UP
0334: 8D 7C 10       STA BUFF+$7C ; TRACK $11
0337: 8D 7D 10       STA BUFF+$7D ; IN VTOC.
033A: A9 23          LDA #23     ; FIX CATALOG
033C: 8D 01 10       STA BUFF+$01 ; POINTER.
*
*-----*
*          WRITE TRACK $23
*-----*
*
033F: A9 02          WRITE23 LDA #02     ; CODE FOR WRITE.
0341: 8D F4 B7       STA IOB+$C   ; STORE IT IN IOB.
0344: A9 23          LDA #23     ; WRITE TRACK
0346: 8D EC B7       STA IOB+$4   ; $23.
0349: 20 79 03       JSR SETIOB  ; SET UP THE IOB.
034C: 20 93 B7       JSR RW.PAGES ; GO WRITE TRACK $23.
*
*-----*
*          ZERO OUT TRACK $11
*-----*
*
034F: A9 1F          ZERO11  LDA /BUFF+$F00 ; FILL THE
0351: 85 FF          STA PTR+1    ; R/W BUFFER
0353: A9 00          LDA #BUFF    ; WITH 00'S
0355: 85 FE          STA PTR      ; BEFORE
0357: A2 10          LDX #10     ; WRITING IT
0359: A8            TAY         ; TO TRACK
035A: 91 FE          STA (PTR),Y ; $11.
035C: C8            INY
035D: D0 FB          BNE .1
035F: C6 FF          DEC PTR+1
0361: CA            DEX
0362: D0 F6          BNE .1
*
*-----*
*          WRITE TRACK $11
*-----*
*
0364: A9 02          WRITE11 LDA #02     ; CODE FOR WRITE.
0366: 8D F4 B7       STA IOB+$C   ; STORE IT IN IOB.
0369: A9 11          LDA #11     ; WRITE TO
036B: 8D EC B7       STA IOB+4   ; TRACK $11.
036E: 20 79 03       JSR SETIOB  ; SET UP THE IOB.
0371: 20 93 B7       JSR RW.PAGES ; GO WRITE IT.
0374: A9 00          LDA #00     ; FIX THIS
0376: 85 48          STA $48     ; LOCATION.
0378: 60            DONE     RTS      ; ALL DONE
*
*-----*
*          SET UP IOB PARAMETERS
*-----*
*
0379: A9 00          SETIOB  LDA #00     ; READ/WRITE
037B: 8D EB B7       STA IOB+3   ; ANY VOL. #
037E: A9 0F          LDA #F      ; START AT
0380: 8D ED B7       STA IOB+5   ; SECTOR $F
0383: A9 00          LDA #BUFF    ; LOW BYTE
0385: 8D F0 B7       STA IOB+8   ; OF R/W BUFFER.
0388: A9 1F          LDA /BUFF+$F00 ; TOP PAGE
038A: 8D F1 B7       STA IOB+9   ; OF BUFFER.
038D: A9 10          LDA #10     ; READ/WRITE
038F: 8D E1 B7       STA NUM.PAGES ; 16 PAGES.
0392: 60            RTS

```

2) Write the new VTOC to track \$23, sector \$0. 3) Read in track \$11, sector \$1 and write it out to track \$23, sector \$1. 4) For the rest of the directory sectors (sectors \$2-\$F), you will have to read in the sector from track \$11, change byte \$01 from an \$11 to a \$23 (thereby changing the link, or pointer, to the next directory sector) and then write that sector out to track \$23. For instance: read track \$11, sector \$2, change byte \$01 from a \$11 to a \$23 and then write the sector to track \$23, sector \$2. Repeat this process for sectors \$3 through \$F. No change was needed to sector \$1 because it is the last sector in the directory and it has no links to the next directory sector. 5) Zero out track \$11 by filling the buffer of the sector editor with all 00's and then writing it to each sector of track \$11.

### Final Thoughts

Now, if you boot the disk, it will work just as a normal DOS 3.3 disk. If someone uses COPYA or if any whole disk copy program is used to copy the disk, on boot-up or catalog only a "FILE NOT FOUND" or "DISK VOLUME 254" message will appear. Trying to catalog the disk with a disk utility will only result in a "NO FILES" message. This is because the standard DOS looks for the VTOC on track \$11, sector \$0F and, of course, it is not there. Finally, if someone uses a bit copy program but doesn't set it to copy track \$23 (and who does?), it will not work.

This is really an elementary protection scheme, but it will defeat a lot of would-be copiers because of their lack of knowledge of the way DOS works. Which brings me to the whole point of this: to help you gain a better understanding of what goes on inside of DOS and to increase your knowledge and enjoyment of your Apple.

### VTOC Mover Hexdump

0300:	24 2C 41 2C 31 29 D1 D0	\$936E
0308:	22 46 22 29 C4 82 00 3A	\$A69B
0310:	09 28 00 51 D0 41 3A 42	\$2117
0318:	41 00 32 35 32 3A 80 39	\$FD5B
0320:	30 3A 4E 31 24 D0 E9 28	\$F3DD
0328:	4E 40 24 2C E3 28 4E 4D	\$E042
0330:	24 29 C9 51 29 3A 80 39	\$A5A9
0338:	30 00 5B 09 2D 00 BA 3A	\$9A58
0340:	BA E7 28 34 20 22 42 4C	\$EB17
0348:	4F 41 44 20 43 48 45 43	\$897D
0350:	48 42 49 4E 20 28 4D 57	\$1F79
0358:	29 22 00 A8 09 32 00 9F	\$D06E
0360:	3A BA 3A BA 22 04 4E 4F	\$EBCB
0368:	4D 4F 4E 43 22 3A BA 22	\$CA86
0370:	04 4D 4F 4E 49 4F 22 3A	\$AC51
0378:	BA 22 04 4F 50 45 4E 22	\$3CF0
0380:	41 24 3A BA 22 04 44 45	\$7194
0388:	4C 45 54 45 22 41 24 3A	\$8D71
0390:	BA 22 04 4F	\$D543

# RKCopy:

## Backing-Up SSI's Non-RDOS Disks

By M. M. McFadden

### Requirements:

- Apple ][ Plus or equivalent
- One disk drive
- A nibble copier
- A Roger Keating program

Most of the newer programs written by Roger Keating (Germany 1985, RDF 1985, Baltic 1985, Reach for the Stars etc.) have been published by Strategic Simulations, Inc. Instead of using the infamous RDOS 2.1 on these programs, SSI has used a strange format which almost completely foils any effort to deprotect them. On the other hand, the disks can be easily duplicated by just about any of the newer bit copy programs but, for users with only one drive, this can be extremely inconvenient. This special DOS (which I call RK.DOS) uses only 10 sectors per track and uses the "4+4" encoding technique, (see Hardcore COMPUTIST No. 13, page 25 for a description of this format).

After a bit of searching, I discovered that by setting certain zero-page locations and then CALLING \$3D0, I could read a range of tracks and sectors into memory, or write the memory to the disk. This prompted me to write the program presented below: RKCopy.

RKCopy is written mainly in Applesoft, but requires a short machine-language routine to load RK.DOS into memory. RKCopy can copy a disk formatted with RKDOS in three passes on a 48K machine and, because the tracks will be in sync with one another, disk access will be faster than if the disk were copied with a nibble copier.

Unfortunately, RKCopy cannot initialize disks. The initialize code is not part of RK.DOS and it is stored in different locations on different disks. So, formatting a disk is accomplished by going to the program's I/O menu (usually by hitting **CTRL B**) and selecting the "format disk" option. RKCopy also cannot copy track 0.

The problem is that, in order to boot, the disk must have one sector that is recognizable by DOS 3.3. So, track 0, sector 0 is DOS 3.3 format, while the rest of the track is RKDOS format! Therefore, the only reasonable way to copy this split-format track is with a nibble-copier. Copy ][ Plus did the trick for me, but any other decent bit copier should work, too.

The machine language routine (OBJ.RKCopy) performs a boot code trace on the original disk in order to capture RK.DOS. That is, it will boot the disk. But, instead of allowing the program to execute, it will return control to RKCopy as soon as RK.DOS is loaded.

*Continued on next page*

### RKCopy Source Code

```
.OR $2000
.TF OBJ.RKCOPY

00FB- MOVE.FRM .EQ $FB
00FD- MOVE.TO .EQ $FD
00FF- TEMP .EQ $FF          TEMPORARY STORAGE
0067- BEG.BAS .EQ $67
00AF- END.BAS .EQ $AF

2000: A9 00          LDA #0          FIX LSBS FOR
2002: 85 FB          STA MOVE.FRM MEMORY MOVE ROUTINE
2004: 85 FD          STA MOVE.TO

*-----*
*          MOVE BASIC PROGRAM TO $2100          *
*-----*

2006: A5 00          LDA END.BAS+1 CALCULATE NUMBER OF PAGES TO MOVE
2008: 38          SEC          = END.BAS - BEG.BAS + 1
2009: E5 68          SBC BEG.BAS+1
200B: AA          TAX          PUT ANSWER IN X
200C: E8          INX
200D: A5 68          LDA BEG.BAS+1 MOVE FROM CURRENT BASIC START
200F: A0 21          LDY #$21          MOVE TO $2100
2011: 86 FF          STX TEMP          SAVE # OF PAGES
2013: 20 54 20       JSR MOVE.MEM MOVE PROGRAM

*-----*
*          MOVE BOOT CODE ($C600) TO $1600      *
*-----*

2016: A9 C6          LDA #$C6          FROM $C600 SLOT 6
2018: A0 16          LDY #$16          TO $1600
201A: A2 01          LDX #$01          ONLY ONE PAGE
201C: 20 54 20       JSR MOVE.MEM MOVE IT
```

*Continued on next page*

### Personal Use Of RK.DOS

To use RK.DOS for your own applications, set \$00 and \$01 to the buffer range that you want to read in or write out. Then, set \$04 and \$05 to the track and sector to start on. When you CALL \$3D0, RK.DOS will read a range of tracks and sectors until the buffer is filled. The following tables should be helpful.

### RK.DOS Special Locations

Loc.	Description (RKCopy var. name)
\$00	Buffer start (BS)
\$01	Buffer end (BE)
\$03	Command: 1=read, 2=write (CMD)
\$04	Track (TRK)
\$05	Sector (SEC)
\$3D0	RWTS entry point (IO)
\$3D3	Slot (SLT)
\$3D4	Drive number (DRV)

### RKCopy Variables

Var.	Description
BA	Buffer start
BB	Buffer end
DV	Number of drives
S	==16336 (used for click)
SC	Current sector
ST	Step number
TA, TB, TC	used to determine next track, sector, and buffer
TK	Current track
X	Multi-purpose variable
AS	Used for input
CO\$(n)	Holds instructions for step n
IN\$	Prints a prompt
ST\$(n)	Holds name of step n

Continued from previous page

```

*-----*
*               MODIFY BOOT0                *
*-----*

201F: A9 2C      LDA #PROG1  MODIFY CONTROLLER ROM
2021: 8D F9 16   STA $16F9  TO JUMP TO PROG1 WHEN FINISHED
2024: A9 20      LDA /PROG1  DO MSB
2026: 8D FA 16   STA $16FA
2029: 4C 00 16   JMP $1600  EXECUTE BOOT 0 (WILL RETURN TO PROG1)

*-----*
*               MODIFY BOOT1                *
*-----*

202C: A9 39      PROG1  LDA #PROG2  MAKE BOOT1
202E: 8D 8D 08   STA $088D  JMP TO PROG2
2031: A9 20      LDA /PROG2  WHEN FINISHED
2033: 8D 8E 08   STA $088E
2036: 4C 01 08   JMP $0801  EXECUTE BOOT 1

*-----*
*               LOAD MAIN RKDOS INTO MEMORY *
*-----*

2039: A9 4C      PROG2  LDA #$4C   CODE FOR JUMP
203B: 8D DA 10   STA $10DA  MAKE BOOT2 GO TO
203E: A9 48      LDA #PROG3  PROG3 WHEN DONE
2040: 8D DB 10   STA $10DB  LOADING RKDOS
2043: A9 20      LDA /PROG3  INTO MEMORY
2045: 8D DC 10   STA $10DC
2048: 4C 00 10   JMP $1000  EXECUTE MODIFIED CODE

*-----*
*               EXIT PROGRAM                *
*-----*

2048: AD E8 C0   PROG3  LDA $C0E8  TURN OFF DRIVE
204E: A9 21      LDA #$21   MOVE BASIC PROGRAM FROM $2100
2050: A4 68      LDY BEG.BAS+1 BACK TO START OF BASIC PROGRAM
2052: A6 FF      LDX TEMP  GET SAVED #PAGES, FALL THROUGH TO MEM.MOVE

*-----*
*               MEMORY MOVE ROUTINE        *
*               ON ENTRY:                  *
*               A=FROM MSB, Y=TO MSB, X=# OF PAGES *
*-----*

2054: 85 FC      MOVE.MEM STA MOVE.FRM+1 STORE FROM MSB (LSB MUST BE ZERO)
2056: 84 FE      STY MOVE.TO+1 STORE TO MSB (LSB MUST BE ZERO)
2058: A0 00      LDY #0    START WITH OFFSET ZEROED
205A: B1 FB      MOV1BYTE LDA (MOVE.FRM),Y GET A BYTE
205C: 91 FD      STA (MOVE.TO),Y STORE IT AT DESTINATION
205E: C8        INY      DONE WITH PAGE?
205F: D0 F9      BNE MOV1BYTE NOPE, KEEP MOVING
2061: E6 FC      INC MOVE.FRM+1 NEXT FROM MSB
2063: E6 FE      INC MOVE.TO+1 NEXT TO MSB
2065: CA        DEX     DONE WITH MOVE?
2066: D0 F2      BNE MOV1BYTE NOPE, KEEP GOING
2068: 60        RTS     RETURN TO CALLER

```



### Keying It In

First of all, enter the hexdump at the end of this article (using the procedure outlined in the How-To's on page 2) and save it with

**BSAVE OBJ.RKCOPY, AS2000, LS69**

Second, type in the BASIC program found at the end of this article (again using the procedure detailed on page 2) and save it with

**SAVE RKCOPY**

### Using The Program

Initialize a disk with the RK.DOS format before running RKCOPY. You can usually accomplish this by hitting <sup>CTRL</sup>B from "Global Mode" of your protected disk. This will get you to the utility options menu. In most cases, hitting "I" will invoke the format routine. When the disk is initialized, boot normal DOS again and

**RUN RKCOPY**

and follow the instructions. Don't forget to nibble copy track 0 when you're through. Happy war-gaming.

### RKCopy BASIC Program

```

10 TEXT : NORMAL : HOME
20 VTAB 3 : HTAB 14 : PRINT "****"
   RKCOPY****"
30 PRINT CHR$(4) "BLOADOBJ.RKCOPY"
40 VTAB 6 : PRINT "THIS^PROGRAM^
   WILL^COPY^REACH^FOR^THE" :
   PRINT "STARS",^
   "GERMANY^1985",^"RDF^1985",^
   AND"
50 PRINT "ALMOST^ANYTHING^WRITTEN^
   RECENTLY^BY" : PRINT "ROGER^"

```

```

KEATING"
60 PRINT : PRINT "***NOTE: ARKCOPY
  CANNOT FORMAT DISKS." : PRINT
  TAB(4) "USE THE INITIALIZE
  DISK OPTION"
70 PRINT TAB(4) "FROM THE GAME'S
  I/O MENU INSTEAD."
80 PRINT : PRINT "PLEASE INSERT A
  ORIGINAL DISK IN DRIVE 1:" ;
90 PR# 0 : IN# 0 : POKE - 16368 , 0 : GET
  AS
100 POKE - 16368 , 0 : CALL 8192
110 CLEAR : GOSUB 360
120 HOME : ST = 1 : GOSUB 320
130 GET AS : IF AS <> "S" AND AS <> "D"
  THEN PRINT CHR$(7) ; : GOTO 130
140 DV = 1 : IF AS = "D" THEN DV = 2
150 VTAB 2 : HTAB 32 : PRINT "DRIVES:"
  ; DV
160 GOSUB 320 : GET AS : SC = 0 : TK = 1
170 IF DV = 1 THEN VTAB 14 : PRINT :
  PRINT "INSERT SOURCE DISK
  AND HIT <SPACE>" ; : CALL - 868 :
  POKE - 16368 , 0 : GET AS : IF AS <>
  "A" THEN 170
180 POKE BS , BA : POKE BE , BB : POKE
  CMD , 1 : POKE TRK , TK : POKE SEC
  , SC : POKE DRV , 1 : CALL IO : X =
  PEEK (S) + PEEK (S)
190 TA = PEEK (4) : TB = PEEK (5)
200 IF DV = 1 THEN VTAB 14 : PRINT :
  PRINT "INSERT DESTINATION
  DISK AND HIT <SPACE>" ; : POKE
  - 16368 , 0 : GET AS : IF AS <> "A"
  THEN 200
210 POKE BS , BA : POKE BE , BB : POKE
  CMD , 2 : POKE TRK , TK : POKE SEC
  , SC : POKE DRV , DV : CALL IO
220 TB = TB + 1 : IF TB = 10 THEN TB = 0
  : TA = TA + 1
230 TK = TA : SC = TB : TC = (TA * 10) +
  TB
240 BB = BA + (349 - TC) : IF BB - BA < 0
  THEN 270
250 IF BB > 185 THEN BB = 185
260 X = FRE (0) : GOTO 170
270 PRINT CHR$(7) : VTAB 10 : PRINT :
  CALL - 868 : GOSUB 320
280 GET AS : HOME : VTAB 10 : PRINT
  TAB(7) "HIT <SPACE> TO COPY A
  ANOTHER" : PRINT : PRINT TAB(11
  ) "OR <RET> TO REBOOT:"
290 PRINT : HTAB 19 : PRINT IN$ ;
300 GET AS : IF AS = CHR$(13) THEN
  PR# 6 : PRINT
310 GOTO 110
320 X = FRE (0) : IN$ = "<" + CHR$(95)
  + ">" + CHR$(8) + CHR$(8) : IF
  ST = 1 THEN 340
330 VTAB 2 * (ST - 1) : HTAB 1 : PRINT
  "STEP" ST - 1 "A:" ; ST$(ST - 1)
340 VTAB 2 * ST : PRINT "STEP" ST "A
  :A" ; : INVERSE : PRINT ST$(ST) :
  NORMAL
350 VTAB 18 : CALL - 958 : PRINT
  CO$(ST) : VTAB 22 : HTAB 19 :
  PRINT IN$ ; : ST = ST + 1 : RETURN

```

# Are you a NEW SUBSCRIBER?

**BACK ISSUES of  
Hardcore COMPUTIST and ★ CORE are  
PACKED with information that you  
won't want to miss.**

**Hardcore COMPUTIST 17:** Softkeys for The Print Shop, Crossword Magic, The Standing Stones, Beer Run, Skyfox, and Random House Disks / A Tutorial For Disk Inspection and the Use Of Super IOB / The Graphic Grabber For The Print Shop / The Lone Catalog Arranger Part Two / S-C Macro Assembler Directives (Reprint)

**Hardcore COMPUTIST 16:** Softkeys for Rescue Raiders, Sheila, Basic Building Blocks, Artscl Programs, Crossfire, Sensible Speller for ProDOS and Sideways / Secret Weapon: RAMcard / The Controller Writer / A Fix For The Beyond Castle Wolfenstein Softkey / The Lone Catalog Arranger Part 1

**Hardcore COMPUTIST 14:** Super IOB v1.2 Update / Putting Locksmith 5.0 Fast Copy Into a Normal Binary File / Softkeys for Seadragon, Rocky's Boots, Knoware, PFS Software, Computer Preparation SAT & Mathemagic / Batman Decoder Ring / REVIEW: Boulder Dash / A Fix For DiskEdit.

**Hardcore COMPUTIST 13:** Softkeys for Laf Pak, Beyond Castle Wolfenstein, Transylvania and The Quest, Electronic Arts, Snooper Troops (Case 2), DLM Software, Learning With Leeper, & TellStar / CSaver: The Advanced Way to Store Super IOB Controllers / Adding New Commands to DOS 3.3 / Fixing A ProDOS 1.0.1 BSAVE Bug / REVIEW: Enhancing Your Apple / Locksmith 5.0 and the Locksmith Programming Language.

**Hardcore COMPUTIST 11:** Copy II Plus 4.4C Update / PARS for Essential Data Duplicator / Ultimaker III / Mapping of Ultima III / Ultima II...The Rest of the Picture / Softkeys for Sensible Speller, Ultima III, Softporn Adventure, The Einstein Compiler v5.3, & Mask of the Sun.

**Hardcore COMPUTIST 10:** Controller Saver / Softkeys for The Arcade Machine, Bankstreet Writer, Minit Man, Sensible Speller IV, Essential Data Duplicator III, Krell LOGO, & Canyon Climber, ApplEar / REVIEWS: The 65SC802 & 65SC816 Chips and Dino Eggs / Examining Protected Applesoft Basic Programs / Crunchlist II / Pams for DB Master v4.

**Hardcore COMPUTIST 4:** Ultima II Character Editor / Softkeys for Ultima II, Witness, Prisoner II, & Pest Patrol / Adventure Tips for Ultima II & III / Copy II Plus PARS Update.

**Hardcore COMPUTIST 1:** Softkeys for Data Reporter, Multiplan & Zork / PARS for Copy II Plus / No More Bugs / APT's for Choplifter & Cannonball Blitz / Reviews: Replay, Crackshot, Snapshot & Wildcard copy cards.

**CORE 3 Games:** Constructing Your Own Joystick / Compiling Games / GAME REVIEWS: Over 30 of the latest and best / Pick Of The Pack: All-time TOP 20 games / Destructive Forces / EAMON / Graphics Magician and GraFORTH / and Dragon Dungeon.

**CORE 2 Utilities:** Dynamic Menu / High Res: Scroll Demo / GOTO Label: Replace / Line Find / Quick Copy: Copy.

**CORE 1 Graphics:** Memory Map / Text Graphics: Marquee, Boxes, Jagged Scroller / Low Res: Color Character Chart / High Res: Screen Cruncher, The UFO Factory / Color / Vector Graphics: Shimmering Shapes, A Shape Table Mini-Editor / Block Graphics: Arcade Quality Graphics for BASIC Programmers / Animation.

(\* CORE is no longer published as an independent quarterly magazine. Back issues not listed are no longer available.)

<b>Send me the back issues indicated below:</b>		HC 10.....	\$3.50	<input type="checkbox"/>	
HC 17.....	\$3.50	<input type="checkbox"/>	HC 4.....	\$3.50	<input type="checkbox"/>
HC 16.....	\$3.50	<input type="checkbox"/>	HC 1.....	\$3.50	<input type="checkbox"/>
HC 14.....	\$3.50	<input type="checkbox"/>	CORE 1 Graphics.....	\$3.50	<input type="checkbox"/>
HC 13.....	\$3.50	<input type="checkbox"/>	CORE 2 Utilities.....	\$3.50	<input type="checkbox"/>
HC 11.....	\$3.50	<input type="checkbox"/>	CORE 3 Games.....	\$3.50	<input type="checkbox"/>

Name \_\_\_\_\_ ID# \_\_\_\_\_

Address \_\_\_\_\_

City \_\_\_\_\_ St \_\_\_\_\_ Zip \_\_\_\_\_

Phone \_\_\_\_\_

VISA/MC \_\_\_\_\_ Exp \_\_\_\_\_

Signature \_\_\_\_\_

**Send check or money order to:** Hardcore COMPUTIST, PO Box 110846-B, Tacoma, WA 98411. Most orders shipped UPS. Please use street address. Washington residents add 7.8% sales tax. Foreign orders add 20% shipping and handling. US funds drawn on US bank.

Void after June 30, 1985

Continued from previous page

```
* 360 QS = CHR$(34) : ST$(1) = "DRIVE^
SELECTION" : COS(1) = "ENTER^
AN^" + QS + "S" + QS + "A^FOR^
SINGLE^DRIVE^COPY^OR^A^" + QS +
"D" + QS + "A^FOR^DUAL" + "1^AND^
THE^DESTINATION^DISK^IS^IN^
DRIVE^2^ (OR^IN^YOUR^HAND^IF^
YOU^HAVE^ONE^DRIVE)"
370 ST$(2) = "COPYING" : COS(2) =
"MAKE^SURE^THE^SOURCE^DISK^
IS^IN^DRIVE^4^YOU^RE^GOING^TO^
A^HAVE^TO^COPY^TRACK^0^B^
YOURSELF^WITH^A^NIBBLE^
COPIER^ (SORRY)"
380 ST$(3) = "FINAL^TOUCH" : COS(3)
= "TRACK^0^HAS^BEEN^MODIFIED^
TO^WORK^WITH^THE^3.3^
BOOT^PROGRAM."
390 S = -16336
400 BS = 0 : BE = 1 : CMD = 3 : TRK = 4 : SEC
= 5 : BA = 17 : BB = 185 : DRV = 980
: SLT = 979 : IO = 976 : RETURN
```

**RKCopy Hexdump**

2000:	A9 00 85 FB 85 FD A5 B0	\$249D
2008:	38 E5 68 AA E8 A5 68 A0	\$22C4
2010:	21 86 FF 20 54 20 A9 C6	\$BDF7
2018:	A0 16 A2 01 20 54 20 A9	\$9040
2020:	2C 8D F9 16 A9 20 8D FA	\$853B
2028:	16 4C 00 16 A9 39 8D 8D	\$501A
2030:	08 A9 20 8D 8E 08 4C 01	\$887A
2038:	08 A9 4C 8D DA 10 A9 4B	\$B61A
2040:	8D DB 10 A9 20 8D DC 10	\$B7E0
2048:	4C 00 10 AD E8 C0 A9 21	\$F3EF
2050:	A4 68 A6 FF 85 FC 84 FE	\$FFFF
2058:	A0 00 B1 FB 91 FD C8 D0	\$1B8B
2060:	F9 E6 FC E6 FE CA D0 F2	\$AEBA
2068:	60	\$861E

**RKCopy BASIC Checksums**

10	- \$A7A5	210	- \$94DD
20	- \$9EA6	220	- \$D552
30	- \$4E9C	230	- \$AF4A
40	- \$3F00	240	- \$2704
50	- \$7660	250	- \$5848
60	- \$AEB6	260	- \$AD1B
70	- \$827A	270	- \$1630
80	- \$A96F	280	- \$414F
90	- \$63F7	290	- \$074E
100	- \$5029	300	- \$7F50
110	- \$6338	310	- \$47AE
120	- \$A394	320	- \$097B
130	- \$F6F6	330	- \$674F
140	- \$4D2F	340	- \$8FB8
150	- \$0A46	350	- \$DE4A
160	- \$3431	360	- \$1A86
170	- \$BE8C	370	- \$CF12
180	- \$62CA	380	- \$C181
190	- \$0CFB	390	- \$408E
200	- \$C241	400	- \$D0CB



# Get 25 SS/SD 5 1/4" diskettes

(ANSI Spec 100% guaranteed, with reinforced hubs. Tyvek sleeves and write-protect tabs included.)

For Only **\$18.75**

**That's only 75¢ per disk!!**  
(Sold in multiples of 25 ONLY.)

Please send me \_\_\_\_\_ sets (25 disks per set) of 5 1/4" SS/SD diskettes. I have enclosed \$18.75 plus shipping and handling for each set.

Name \_\_\_\_\_  
 Address \_\_\_\_\_  
 City \_\_\_\_\_ St. \_\_\_\_\_ Zip \_\_\_\_\_  
 Country \_\_\_\_\_  
 VISA/MC \_\_\_\_\_ Exp \_\_\_\_\_  
 Signature \_\_\_\_\_

Send your order to: Disk Offer, SoftKey Publishing, PO Box 110816, Tacoma, WA 98411. Please enclose \$3 shipping and handling for the first set plus \$1 per additional set. Foreign orders add 20% shipping and handling. US funds drawn on US bank. Washington state residents add 7.8% sales tax. Orders shipped via UPS. Please use street address.

## ENHANCE your Apple! with a

# 65C02

Upgrade to this pin-for-pin compatible CMOS version of the 6502 microprocessor.

**Advantages of the 65C02:**

- ▶ 27 NEW Machine Language Opcodes
- ▶ 8 Completely NEW Machine Language Instructions
- ▶ Low Power CMOS Design
- ▶ Invalid Opcodes Default to 1 Cycle NOP's
- ▶ The JMP(XXFF) Bug Has Been Fixed

Only **\$9.95**

These chips are the 2MHZ versions manufactured by Western Design Center of Mesa, AZ. Used in the Apple IIc, the 65C02 microprocessor is also included in the Apple IIe upgrade package recently released by Apple Computer (for more information on the 65C02, see the article on page 19 of this issue).

Each order will include a data sheet and installation instructions.

**PLACE YOUR ORDERS NOW!**  
Offer expires May 31

Please send \_\_\_\_\_ 65C02 chip(s) to the address below. I have enclosed \$9.95 plus shipping and handling for each chip.

Name \_\_\_\_\_  
 Address \_\_\_\_\_  
 City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_  
 VISA/MC \_\_\_\_\_ Exp \_\_\_\_\_  
 Signature \_\_\_\_\_

Domestic orders please enclose \$1.00 shipping and handling for the first chip + .25 per additional chip; foreign orders add 20%. US funds drawn on US bank. Washington state residents add 7.8% sales tax. Please allow 6 weeks for delivery. Send order to: Chip Offer, Hardcore COMPUTIST, PO Box 110846-C, Tacoma, WA 98411.



# STILL TYPING IN PROGRAMS THAT APPEAR IN HARDCORE COMPUTIST?

## Order the LIBRARY DISK, instead!

Each month a Library Disk with all the programs that appeared in the previous issue of Hardcore COMPUTIST is prepared for SMART READERS like you who have better things to do with their time than type in program listings.

**Disk No. 17** for Hardcore COMPUTIST 17 includes Controllers for Skyfox, Zaxxon, The Print Shop and Crossword Magic, Super Controller, Graphic Grabber Source Code and BASIC Program & Lone Catalog Arranger Part Two Source Code.

**Disk No. 16** for Hardcore COMPUTIST 16 includes Controllers for Sheila, AceCalc (for Artsci programs) and Beyond Castle Wolfenstein (revised version), Controller Writer & Lone Catalog Arranger Part One BASIC Program.

**Disk No. 15** for Hardcore COMPUTIST 15 includes Master Type Controller & Boot From Drive 2.

**Disk No. 14** for Hardcore COMPUTIST 14 includes SAT Controller, Sea Dragon Controller, Super IOB v1.2 with Standard Controller and Swap Controller, Batman Decoder Ring & Rocky's Boots Controller.

**Disk No. 10** for Hardcore COMPUTIST 13 includes Penguin Controller, Snooper Troops Controller, CSaver, New DOS 3.3 Command Listings & Electronic Arts Controller

**Disk No. 9** for Hardcore COMPUTIST 12 includes The Armonitor, Lion's Share Controller, CORE Disk Searcher & Psychedelic Symphony

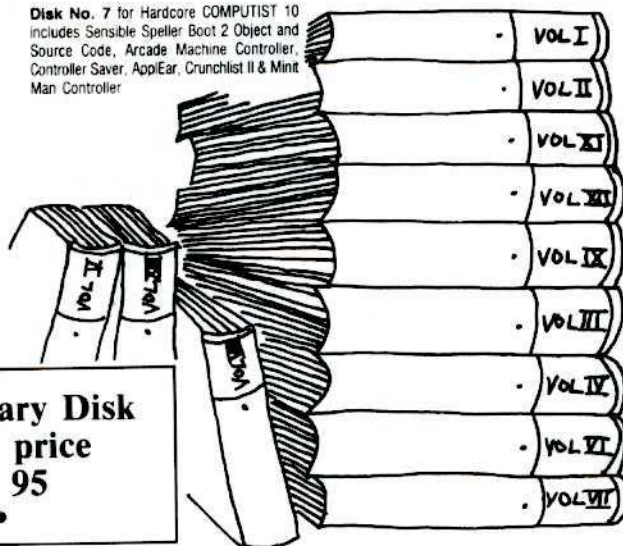
**Disk No. 8** for Hardcore COMPUTIST 11 includes Ultimaker, Ultimapper, Ultima III Controller & Sensible Speller.Loader.Mod Source Code

**Disk No. 7** for Hardcore COMPUTIST 10 includes Sensible Speller Boot 2 Object and Source Code, Arcade Machine Controller, Controller Saver, AppiEar, Crunchlist II & Minit Man Controller

**Disk No. 3** for CORE Games issue includes Destructive Forces & Dragon Dungeon

**Disk No. 2** includes programs for CORE Utilities issue and Hardcore COMPUTIST 4

**Disk No. 1** includes programs for CORE Graphics issue and Hardcore COMPUTIST 1



Get each Library Disk  
for the low price  
of **\$9.95**

Please send me the Library Disk(s) I have indicated below. I have enclosed \$9.95 (plus applicable tax & shipping) for each disk.

**NOTE:** Library Disks need accompanying magazine for documentation. Library Disks for sold-out issues are available upon request.

**Send check or money order to:** Hardcore COMPUTIST, PO Box 110846-B, Tacoma, WA 98411. Most orders shipped UPS. Please use street address. Washington residents add 7.8% sales tax. Foreign orders add 20% shipping and handling. US funds drawn on US bank.

- |                                      |                                     |
|--------------------------------------|-------------------------------------|
| <input type="checkbox"/> Disk No. 17 | <input type="checkbox"/> Disk No. 8 |
| <input type="checkbox"/> Disk No. 16 | <input type="checkbox"/> Disk No. 7 |
| <input type="checkbox"/> Disk No. 15 | <input type="checkbox"/> Disk No. 3 |
| <input type="checkbox"/> Disk No. 14 | <input type="checkbox"/> Disk No. 2 |
| <input type="checkbox"/> Disk No. 10 | <input type="checkbox"/> Disk No. 1 |
| <input type="checkbox"/> Disk No. 9  |                                     |

### SPECIAL!

Order FIVE Library Disks  
of your choice  
for only \$40.00!

No. \_ / \_ / \_ / \_ / \_

Name \_\_\_\_\_ ID# \_\_\_\_\_

Address \_\_\_\_\_

City \_\_\_\_\_ St \_\_\_\_\_ Zip \_\_\_\_\_

Phone \_\_\_\_\_

VISA/MC \_\_\_\_\_ Exp \_\_\_\_\_

Signature \_\_\_\_\_

### What does Hardcore COMPUTIST offer its readers?

More of what they need to know about backing up Apple software than ANY OTHER magazine on the market.

### Why should YOU subscribe?

Because any *serious* Apple computer user knows he can get MORE out of the software he buys using the advanced deprotection techniques found in each issue of Hardcore COMPUTIST.

- \* MORE backups!
- \* MORE modifications!

**DON'T USE your valuable software without FIRST backing-it-up.**

INSTEAD, rely on Hardcore COMPUTIST to provide you with up-to-date deprotection methods for all the new games and programs available for the Apple II line of computers.

# Hardcore COMPUTIST

### Annual Rates: Please check one

- |   |        |
|---|--------|
| <input type="checkbox"/> U.S. ....                    | \$25   |
| <input type="checkbox"/> Canada, U.S. 1st Class ..... | \$34   |
| <input type="checkbox"/> Mexico .....                 | \$39   |
| <input type="checkbox"/> Foreign Airmail .....        | \$60   |
| <input type="checkbox"/> Foreign surface mail .....   | \$40   |
| <input type="checkbox"/> SAMPLE, U.S. ....            | \$3.50 |
| <input type="checkbox"/> SAMPLE, Foreign .....        | \$4.50 |

( ) Yes, start my subscription now.  
( ) I would like to RENEW my subscription. (Please paste PRESENT MAILING LABEL below)

Name \_\_\_\_\_  
Address \_\_\_\_\_  
City \_\_\_\_\_ St \_\_\_\_\_ Zip \_\_\_\_\_  
Phone \_\_\_\_\_  
VISA/MC \_\_\_\_\_ Exp \_\_\_\_\_

Signature \_\_\_\_\_  
Send check or money order (US funds drawn on US bank) to: Hardcore COMPUTIST, PO Box 110846-T, Tacoma, WA 98411

# WHO LIKES THE CIA?



HERE ARE JUST A  
FEW OF THE MANY

- "an essential part of the Apple-user's repertoire" - APPLE USER
- "a valuable buy ... manual is practically worth having on its own" - WASHINGTON APPLE PI
- "the folks at Golden Delicious should be commended ... worth waiting for" - HARDCORE COMPUTIST
- "multifaceted" - NIBBLE
- "the most comprehensive disk accessor I have ever come across" - A. B., VERNON, CANADA
- "its ability to unlock other programs will greatly help me" - DR. B.P., SAN FRANCISCO, CALIFORNIA
- "an excellent set of programs ... just great - and good value too" - E. A. S., MILTON KEYNES, ENGLAND
- "very, very educational ... great manual ... it is FANTASTIC!!!!!!" - J.C., TUCSON, ARIZONA
- "a very enlightening piece of software/book ... top of my list for good buys" - H.S., BLAINE, MINNESOTA
- "I like yours the BEST" - R.R., CHICAGO

Why all the excitement about the CIA (confidential information advisors)? Probably because it is the ONLY set of utilities (5 in all) which enable even a beginner to investigate, edit, locate, list, trace, rescue, translate, patch, repair, verify, examine, protect, unprotect, analyse, encrypt, and decrypt programs on normal AND protected disks. You also get the "CIA Files", a 65000+ word book which contains detailed instructions for using the C.I.A. plus easy-to-follow, hand-holding tutorials about patching, repair, formatting, encoding, protection, and numerous other disk topics. You'll find plenty of material here which has never before appeared in print. PROGRAMS NOT COPY PROTECTED

To put the 5 C.I.A. utilities, plus book, on the trail of your Apple II+/IIe (R) disks, send \$65.00 by check or money order to:

GOLDEN DELICIOUS SOFTWARE LTD.  
350 Fifth Avenue, Suite 3308, Dept H, New York, New York 10118

## DEALERS!

Overstocked on *computer magazines*  
that seem to have  
**MORE ADS** than articles?

Put Hardcore **COMPUTIST**  
on your shelves, instead!

### Here's how:

Send in the coupon below and we'll put your dealer info in the mail. DON'T WAIT! Apple computer users everywhere salute you!

Yes! Please send me information on how I can obtain copies of Hardcore COMPUTIST at substantial savings.

Dealer Name \_\_\_\_\_

Address \_\_\_\_\_

City \_\_\_\_\_ St \_\_\_\_\_ Zip \_\_\_\_\_

Send to: Hardcore COMPUTIST, PO Box 110816,  
Tacoma, WA 98411 or CALL (206) 474-5750



Now-the ultimate back-up system!

# EDD III® and TRAK STAR™

COMPLETE PACKAGE:

# 159<sup>95</sup>\*

Includes • TRAK STAR • 2-Drive Adaptor  
• EDD III • Trak Star Patching Software

PRICED SEPARATELY:

## TRAK STAR 99<sup>95</sup>

2-Drive Adapter (required for 2-drive systems) \$12

Documentation: \$3

\* Please add \$3 for shipping

Refundable with the

& handling. Foreign airmail &

purchase of TRAK STAR

handling, add \$8

## Save copying time with nibble programs

- Works with nibble copy programs to display tracks and half tracks that the program accesses.
- Operates with any Apple®-compatible program.
- Save time by copying only the tracks being used.
- Displays up to 80 tracks and half-tracks; compatible with high density drives.
- If copied program doesn't run, Trak Star displays track to be recopied.
- Includes patching software for Trak Star.
- Compact size permits placement on top of disk drive.
- Does not use a slot in the Apple® computer.
- For Apple II, II+, IIe and compatibles.

Personal checks,  
M.O., VISA &  
Mastercard  
accepted.



Apple is a registered  
trademark of Apple  
Computer, Inc. EDD III  
is a trademark of  
Utilico Microware

## Midwest Microsystems

To order, phone:  
913 676-7242

9071 Metcalf / Suite 124  
Overland Park, KS 66212

*By Hackers  
For Hackers*

- ELITE BOARD DOWNLOADS
- CRACKING TIPS
- PHREAKING SECTION
- GAME CHEATS
- PARMS
- PROGRAMS
- INTERVIEWS
- ADVENTURE TRIPS
- HACKING TIPS
- MYSTERY SECTIONS

Published on both sides of  
an Apple diskette -  
4 times a year.

# The BOOT-LEGGER MAGAZINE

**Subscribe Now!**

Send 25 Bucks for a 1-Year Subscription  
THE BOOT LEGGER, 3310 Holland Loop  
Road, Cave Junction, Oregon 97523.  
Overseas Subscriptions \$50.  
Canadian \$30 U.S. Currency.

FOR AD INFO. & QUESTIONS  
CALL BOOTLEG AT (503) 592-4461

# SIMPLY SOFTWARE

DISCOUNT SPECIALS ON APPLE SOFTWARE • ALL PRICES 30% OFF

<p>LIST \$20.00 Hypnosis PixIt Shape Library #1</p> <p>LIST \$24.95 Code Breaker Print Shop Graphics Library</p> <p>LIST \$29.95 Apple Panic Apple Spice Arcade Album Figures and Formulas Freddy's Puzzling Adventures</p>	<p>SPECIAL \$14.00 PixIt Shape Library #2 Terrapin Utilities</p> <p>SPECIAL \$17.47 Fastgammon Super Quality Epson Hi-Res Dump</p> <p>SPECIAL \$20.97 Hall of Alchemie How to Program in Basic Lunar Leeper Multiple Choice Filer Pole Position</p>	<p>LIST \$34.95 Addition Sequences Division Sequences NATO Commander Pathwords Percents Sequences Romper Room's I Love My Alphabet</p> <p>LIST \$39.95 Agent U.S.A. Brain Game Conan Copy II + Donald Duck's Playground Goofy's Word Factory Imperium Galactium Mickey's Space Adventure Mines of Qyntarr Willy Byte in the Digital Demension</p>	<p>SPECIAL \$24.47 Sea Dragon Shuttle Intercept Solo Flight Spy vs. Spy Subtraction Sequences Transylvania Turbo Tutor</p> <p>SPECIAL \$27.97 Operation Frog Run for It Sticky Bear Math Sticky Bear Reading Story Maker Fact n' Fic Tool Kit Trivia Fever Vidtex Welcome Aboard Winnie the Pooh</p>	<p>LIST \$40.00 ARCO Computer Prep. for SAT Bank Street Filer</p> <p>LIST \$99.95 Beginning Algebra Doubletime Printer</p> <p>LIST \$125.00 Elite Controller PFS: File</p>	<p>SPECIAL \$48.97 Bank Street Mailer Interior Design Landscape Design</p> <p>SPECIAL \$69.97 Locksmith 5.0 Professional Architectural Design</p> <p>SPECIAL \$87.50 PFS: Graph PFS: Report PFS: Write</p>	<table border="0"> <thead> <tr> <th>LIST</th> <th>SPECIAL</th> </tr> </thead> <tbody> <tr><td>\$ 35.00</td><td>\$ 24.50</td></tr> <tr><td>\$ 89.95</td><td>\$ 71.96</td></tr> <tr><td>\$ 44.95</td><td>\$ 31.47</td></tr> <tr><td>\$369.00</td><td>\$258.30</td></tr> <tr><td>\$ 44.95</td><td>\$ 31.47</td></tr> <tr><td>\$ 30.00</td><td>\$ 21.00</td></tr> <tr><td>\$ 32.95</td><td>\$ 23.07</td></tr> <tr><td>\$ 55.00</td><td>\$ 38.50</td></tr> <tr><td>\$425.00</td><td>\$297.50</td></tr> <tr><td>\$149.00</td><td>\$104.30</td></tr> <tr><td>\$ 26.95</td><td>\$ 18.87</td></tr> <tr><td>\$ 59.95</td><td>\$ 41.97</td></tr> <tr><td>\$250.00</td><td>\$175.00</td></tr> <tr><td>\$ 28.00</td><td>\$ 19.60</td></tr> <tr><td>\$319.00</td><td>\$223.30</td></tr> <tr><td>\$175.00</td><td>\$122.50</td></tr> <tr><td>\$ 54.95</td><td>\$ 38.47</td></tr> <tr><td>\$ 69.00</td><td>\$ 48.30</td></tr> <tr><td>\$299.00</td><td>\$209.30</td></tr> <tr><td>\$ 79.95</td><td>\$ 55.97</td></tr> </tbody> </table>	LIST	SPECIAL	\$ 35.00	\$ 24.50	\$ 89.95	\$ 71.96	\$ 44.95	\$ 31.47	\$369.00	\$258.30	\$ 44.95	\$ 31.47	\$ 30.00	\$ 21.00	\$ 32.95	\$ 23.07	\$ 55.00	\$ 38.50	\$425.00	\$297.50	\$149.00	\$104.30	\$ 26.95	\$ 18.87	\$ 59.95	\$ 41.97	\$250.00	\$175.00	\$ 28.00	\$ 19.60	\$319.00	\$223.30	\$175.00	\$122.50	\$ 54.95	\$ 38.47	\$ 69.00	\$ 48.30	\$299.00	\$209.30	\$ 79.95	\$ 55.97
LIST	SPECIAL																																															
\$ 35.00	\$ 24.50																																															
\$ 89.95	\$ 71.96																																															
\$ 44.95	\$ 31.47																																															
\$369.00	\$258.30																																															
\$ 44.95	\$ 31.47																																															
\$ 30.00	\$ 21.00																																															
\$ 32.95	\$ 23.07																																															
\$ 55.00	\$ 38.50																																															
\$425.00	\$297.50																																															
\$149.00	\$104.30																																															
\$ 26.95	\$ 18.87																																															
\$ 59.95	\$ 41.97																																															
\$250.00	\$175.00																																															
\$ 28.00	\$ 19.60																																															
\$319.00	\$223.30																																															
\$175.00	\$122.50																																															
\$ 54.95	\$ 38.47																																															
\$ 69.00	\$ 48.30																																															
\$299.00	\$209.30																																															
\$ 79.95	\$ 55.97																																															
<p>ASCII Express Pro DOS Below the Root BreakThru in the Ardennes Build Using Communication Edge Complete Graphics System Computer Preparation for the SAT Cross Talk Data Factory 6.0 Elite I Disk Elite I Plus w/Controller Elite II Disk Elite II Plus w/Controller Elite III Disk Elite III Plus w/Controller Flashcalc Forecast Knoware Letter Wizard w/Spelling Checker Mach II Joystick</p>	<p>LIST \$129.95 \$ 26.95 \$59.95 \$ 30.00 \$195.00 \$ 79.95 \$ 79.95 \$195.00 \$199.00 \$299.00 \$399.00 \$499.00 \$599.00 \$599.00 \$699.00 \$ 99.00 \$ 59.95 \$ 95.00 \$ 74.95 \$ 44.95</p> <p>SPECIAL \$103.96 \$ 19.87 \$ 41.97 \$ 21.00 \$136.50 \$ 55.97 \$ 63.96 \$136.50 \$139.30 \$209.30 \$279.30 \$349.30 \$419.30 \$419.30 \$489.30 \$ 69.30 \$ 41.97 \$ 66.50 \$ 52.47 \$ 31.46</p>	<p>LIST \$40.00 Boulder Dash Miner 2049er II</p> <p>LIST \$49.95 Black Death Blazing Paddles PixIt</p>	<p>SPECIAL \$28.00 Pitstop II Robots of Dawn</p> <p>SPECIAL \$34.97 Return of Werdna Rock's Boots Wilderness</p>	<p>Math SAT I Mathmagic Mission Algebra MPC Parallel Interface w/64K Buffer Paddle Stick PixIt Animation Libraries #1 Prisoner 2 SC Assembler II Softcard II Superbase Swiss Family Robinson Take 1 Tax Preparer 1985 Telengard Titan Accelerator IIe Board Total Telecommunication w/Pro EZ Software Turbo Pascal V Calc Preboot for Ultraterm Volksmodem 1200 Writing Wizard</p>	<p>\$ 35.00 \$ 89.95 \$ 44.95 \$369.00 \$ 44.95 \$ 30.00 \$ 32.95 \$ 55.00 \$425.00 \$149.00 \$ 26.95 \$ 59.95 \$250.00 \$ 28.00 \$319.00 \$175.00 \$ 54.95 \$ 69.00 \$299.00 \$ 79.95</p>	<p>\$ 24.50 \$ 71.96 \$ 31.47 \$258.30 \$ 31.47 \$ 21.00 \$ 23.07 \$ 38.50 \$297.50 \$104.30 \$ 18.87 \$ 41.97 \$175.00 \$ 19.60 \$223.30 \$122.50 \$ 38.47 \$ 48.30 \$209.30 \$ 55.97</p>																																										

PLEASE make check or M.O. payable to: **Simply Software Inc. • P.O. Box 36068 • Kansas City, Missouri 64111**

Add \$3.00 shipping, Missouri residents add 5 5/8% sales tax. Allow 4-6 weeks for delivery.

# DISCOUNTS!

## 5 1/4 DISKETTES & STORAGE

- SS/DD, BOX OF 10 ..... \$12.00
- SS/DD, 10 BOXES ..... \$115.00
- DOUBLE-NOTCHED DS/DD, EACH ..... \$1.35
- DOUBLE-NOTCHED DS/DD, 100 ..... \$125.00
- HARD PLASTIC STAND-UP DISKETTE LIBRARY CASES ..... \$2.75 EACH  
4 FOR \$10.00  
(specify color choices beige, black, blue, green, grey, red, yellow)
- SMOKED PLASTIC JUMBO-SIZE FLIP-TOP 75 DISKETTE FILE CASES ..... \$16.00 \*
- 70-DISKETTE FILE CASES ..... \$12.00 \*
- 140-DISKETTE LOCKING WOOD FILE CABINET ..... \$33.00

## PRINTERS

- PANASONIC P1090 ..... \$239.00
- PANASONIC 1091 ..... \$299.00
- EPSON RX-80 F/T ..... \$289.00
- EPSON RX-100 ..... \$389.00
- EPSON FX-80 ..... \$379.00
- OKI-DATA MICROLINE 92A DOT MATRIX ..... \$369.00
- SILVER REED 400 LETTER QUALITY ..... \$269.00
- STARWRITER A-10 25CPS LETTER QUALITY ..... \$495.00
- TOSHIBA 1340 DOT MATRIX AND LETTER QUALITY COMBINED ..... \$795.00

## PRINTER INTERFACES AND ACCESSORIES

- STANDARD PARALLEL INTERFACE CARD ..... \$49.00
- GRAPHICS PARALLEL INTERFACE CARD ..... \$75.00
- FINGERPRINT PUSH-BUTTON GRAPHICS CARD ..... \$119.00
- MICROFAZER GENERAL PRINT BUFFER ..... \$149.00
- PRINTER STAND ..... \$14.00
- SWITCH BOX, 3 PARALLEL PORTS ..... \$129.00
- SWITCH BOX, 3 SERIAL PORTS ..... \$79.00

## FLOPPY DISK DRIVES

- FOURTH DIMENSION (FULL OR SLIMLINE) ..... \$179.00
- DISTAR ..... \$159.00
- LASER ..... \$139.00 \*
- IIc DRIVES ..... \$158.00 \*
- DISK CONTROLLER ..... \$59.00
- DOUBLE-SIDED DRIVE ..... \$199.00
- 650K RANA DRIVE ..... \$439.00

## HARD DISK DRIVES

- 5 MEGABYTE WITH CONTROLLER AND SOFTWARE ..... \$749.00
- 10 MEGABYTE ..... \$1175.00

## MONITORS

- GORILLA 12-INCH GREEN ..... \$84.00
- USI 12-INCH GREEN ..... \$94.00 \*
- USI 12-INCH AMBER ..... \$99.00 \*
- INTRA 14-INCH COMPOSITE COLOR/80 COLUMN ..... \$239.00

## MODEMS

- ZOOM TELEPHONICS 300 BAUD ..... \$109.00
- CENTAURI 300 BAUD ..... \$179.00
- PRO-MODEM 1200 ..... \$349.00
- PRO-MODEM 1200A INTERNAL ..... \$279.00
- SINGALMAN MARK XII ..... \$239.00

## GRAPHICS DEVICES

- POWER PAD & STARTER KIT ..... \$99.00

## VIDEO & DISPLAY EQUIPMENT

- DIGITIZER ..... \$299.00
- B & W CAMERA ..... \$195.00
- COLOR PROCESSOR ..... \$99.00
- COLOR PROCESSOR/ENHANCER STABILIZER/SYNTHESIZER ..... \$279.00

## GENERAL ITEMS

- 6 OUTLET POWER STRIP ..... \$19.00
- SURGE PROTECTOR ..... \$11.00
- RF MODULATOR ..... \$49.00
- COMPUTER STAND ..... \$24.00

## GAME I/O DEVICES

- CH MACH II JOYSTICK ..... \$37.00
- CH MACH III JOYSTICK ..... \$45.00

\* DENOTES NEW PRICE OR ITEM

## SLOT EXPANSION

- 16 RAM CARD ..... \$49.00
- 64K RAM & 80 COLUMN CARD FOR IIc ..... \$109.00
- MEMORY MASTER IIc 64K + RAM & 80 COLUMN CARD ..... \$145.00
- MEMORY MASTER IIc 128K RAM & 80 COLUMN CARD ..... \$175.00 \*
- MICROTEK II + 128 K VISICALC AND MEMORY EXPANSION ..... \$219.00
- MODEM ELIMINATOR CABLE ..... \$21.00
- SERI-ALL SERIAL INTERFACE CARD ..... \$119.00
- 80-COLUMN CARD (VIEWMASTER) WITH SOFT-SWITCH ..... \$129.00
- CENTAURI APS Z-80 CARD ..... \$59.00
- Z-80 PLUS CARD (CPM FOR APPLE) ..... \$115.00
- FAST Z-80 CARD APPLICARD ..... \$175.00
- TIMEMASTER II CLOCK/CALENDAR CARD ..... \$109.00
- QUICK-LOADER PROM BOARD ..... \$149.00
- ANALOG/DIGITAL BOARD ..... \$99.00
- SUPER I/O BOARD ..... \$49.00
- MULTIPLE-SLOT EXPANSION CHASSIS ..... \$149.00
- SINGLE SLOT EXTENDER ..... \$29.00

## SPECIAL PERIPHERALS

- COOLING FAN WITH SURGE PROTECTOR ..... \$39.00
- TITAN KEYBOARD ..... \$159.00
- LIFETIME EXTERNAL POWER SUPPLY ..... \$179.00
- SHIFT KEY MOD KIT ..... \$8.00
- SCREEN SWITCHER/ DRIVE STEPPER ..... \$74.00

## APPLE SOFTWARE

- WORD STAR ..... \$195.00
- MAIL MERGE/SPELL STAR/ STAR INDEX ..... \$125.00

LONG DISTANCE  
CALL TOLL FREE  
WITH TOUCH TONE PHONE  
FROM ANY CITY!  
DIAL 950-1088  
WAIT FOR TONE  
DIAL 363-1313



ORDERS &  
CALL BACK  
MESSAGES  
(ANSWERING  
MACHINE)  
(202) 362-9176

VISIT OUR NEW STORE  
LOCATION  
8231 WOODMONT AVE.  
(AT BATTERY LANE)  
IN BETHESDA

## STORE HOURS:

12-8 M-TH  
12-6 FRI  
11-5 SAT

SPECIAL  
APPLE IIc  
PRINTER  
CABLES  
\$19.00\*

UPS shipping,  
\$4.00 per order  
plus \$6.00  
per printer  
or monitor

(202) 363-1313  
**ASSOCIATES**

8231 WOODMONT AVE., BETHESDA, MARYLAND