

For The Serious User Of A-II Computers

# COMPUTIST

Issue No. 49

November 1987

USA \$3.75

Canada & Mexico \$7.00

All Others \$8.75

## Softkeys:

Addition Logician  
 Animate  
 Arcade Boot Camp  
 Arctic Fox  
 Bard's Tale II  
 Cat'n Mouse  
 Counting Critters  
 Dam Busters  
 Destroyer  
 Draw Plus v1.0  
 Dr. Ruth's Computer Game Of  
 Good Sex  
 Echo 1.0  
 E.D.D. 4  
 Gamemaker  
 Hard Ball  
 Infiltrator  
 Locksmith 6.0 Fastcopy †  
 Magic Slate  
 Math Critters  
 Millionaire  
 Mind Mirror  
 One On One  
 Paintworks Plus v1.0, 1.1  
 PHM Pegasus  
 Portal  
 Quotient Quest  
 Reader Rabbit  
 Saunder's Chemistry CAI  
 Science Toolkit  
 Shanghai  
 Spell Breaker  
 Strip Poker  
 Super Bunny  
 Super Sunday  
 Swordthrust series  
 Term Paper Writer  
 Top Fuel Eliminator  
 Typing! †  
 Up-n-Down  
 Willy Byte  
 Writer's Choice - Elite v1.0  
 Writing A Character Sketch  
 Writing A Narrative

## Features:

### Steve Marwin's

- Eliminate some ProDOS erroneous error messages
- Date/time without a clock card

### Philip Goetz's

- Sector Surgery
- Generating Applesoft programs on-the-fly



COMPUTIST  
 PO Box 110846-T  
 Tacoma, WA 98411

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

# Welcome to COMPUTIST & Apple RDEX

COMPUTIST, a publication devoted to the serious computer user (especially, Apple II and its compatibles), contains information you are not likely to find in any other major computer journal.

New readers are advised to read this page carefully to avoid frustration when attempting to follow a softkey or when entering the programs printed in this issue.

## ☛...What is a softkey, anyway?

Softkey is a term which we coined to describe a procedure that removes, or at least circumvents, any copy-protection on a particular disk. Once a softkey procedure has been performed, the resulting disk can usually be copied by the normal copy programs (for example, Apple II COMPUTISTS can use COPYA, on their DOS 3.3 System Master Disk).

## ☛...Commands and control keys

In any article appearing in COMPUTIST, commands which a reader is required to perform are set apart by being in boldface and on a separate line:

### PR# 6

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

Control characters are specially boxed:

6 **Ⓜ**

Press **6**. Next, place one finger on **Ⓜ** and press **P**. Remember to enter this command line by pressing **RETURN**.

## ☛...Special requirements

Special prerequisites for COMPUTIST articles, programs and softkeys are usually listed at the start under "■ Requirements:".

## ☛...Software recommendations

For Apple COMPUTISTS:

- ✓ Applesoft program editor such as *Global Program Line Editor (GPLE)*.
- ✓ Sector editor such as *DiskEdit* (in the COMPUTIST Starter Kit) or ZAP from *Bag of Tricks*.
- ✓ Disk-search utility such as *The Inspector*, the *CIA* or the *Core Disk Searcher* (COMPUTIST Starter Kit).
- ✓ Assembler such as the *S-C Assembler* from S-C software or *Merlin/Big Mac*.
- ✓ Bit-copy program such as *Copy II Plus*, *Locksmith* or *EDD* (Essential Data Duplicator).
- ✓ Text editor (that produces normal sequential text files) such as *Applewriter II*, *Magic Window II* or *Screenwriter II*.
- ✓ COPYA, FID and MUFFIN from the *DOS 3.3 System Master* disk are also useful.

## ☛...Super IOB and Controllers

This powerful deprotection utility (COMPUTIST Starter Kit) and its various Controllers are used in many softkeys. (It is also on each Super IOB Collection disk.)

## ☛...Reset into the Monitor

Softkeys occasionally require the user to stop the execution of a copy-protected program and directly enter the Apple's system monitor. Check the following list to see what hardware you will need to obtain this ability.

**Apple II+, //e, compatibles:** 1) Place an Integer BASIC ROM card in one of the Apple slots. 2) Use a non-maskable interrupt (NMI) card such as *Replay* or *Wildcard*.

**Apple II+, compatibles:** 1) Install an F8 ROM with a modified reset-vector on the computer's motherboard as detailed in the *Modified ROM's* article (COMPUTIST6 or *Book Of Softkeys III*) or the *Dual ROM's* article (COMPUTIST19).

**Apple //e, //c:** Install a modified CD ROM on the computer's motherboard. Cutting Edge Ent. (Box 43234 Ren Cen Station-HC; Detroit, MI 48243) sells a hardware device that will give you this important ability but it will void an Apple //e warranty.

## Recommended literature:

The *Apple II Reference Manual* and DOS 3.3 manual are musts for any serious Apple user. Other helpful books include: *Beneath Apple DOS*, Don Worth and Pieter Lechner, Quality Software; *Assembly Language For The Applesoft Programmer*, Roy Meyers and C.W. Finley, Addison Wesley; and *What's Where In The Apple*, William Lubert, Micro Ink.

## ☛...Keying in Applesoft programs:

BASIC programs are printed in COMPUTIST in a format that is designed to minimize errors for readers who key in these programs. If you type:

**10HOME:REMCLEAR SCREEN**

The LIST will look like:

10 HOME : REM CLEAR SCREEN

...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 commands. There are two types of spaces: those that have to be keyed and those that don't. Spaces that must be typed appear in COMPUTIST as delta characters (^). All other spaces are there for easier reading. NOTE: If you want your checksums (See ☛...Computing checksums) to match up, you must only type ^ spaces after DATA statements.

## ☛...Keying In Hexdumps

Machine language programs are printed in COMPUTIST as hexdumps, sometimes also as source code. Hexdumps are the shortest and easiest format to type in. You must first enter the monitor:

**CALL -151**

Key in the hexdump exactly as it appears in the magazine, ignoring the four-digit checksum (\$ and four digits) at the end of each line. A beep means you have typed something that the monitor didn't understand and must, therefore, retype that line.

When finished, return to BASIC with:

**3D0G**

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

## ☛...Keying in source code

The source code is printed to help explain a program's operation. To key it in, you will need the S-C Assembler or you will have to translate pieces of the source code into something your assembler will understand. See the table of S-C Assembler directives in COMPUTIST17.

## ☛...Computing checksums

Checksums are 4-digit hexadecimal numbers which tell if you keyed a program exactly as it appears in COMPUTIST. 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). Both appeared in COMPUTIST1 and *The Best of Hardcore Computing*. An update to CHECKSOFT appeared in COMPUTIST18. If the published checksums do not match those created by your computer, then you typed the program incorrectly. The line where the first checksum differs has an error.

## ■..CHECKSOFT instructions:

**LOAD filename**

**BRUN CHECKSOFT**

Get the checksums with: **&** **RETURN** and correct the program line where the checksums differ.

## ☛...CHECKBIN instructions:

**CALL -151**

**BLOAD filename**

Install CHECKBIN at an out of the way place

**BRUN CHECKBIN,AS\$6000**

Get the checksums by typing the starting address, a period and the ending address of the file followed by a **Y** **RETURN**.

**sss.eee** **Y**

Correct the lines at which the checksums differ.

**You have a  
LEGAL RIGHT  
to an  
unlocked  
backup copy  
of your  
commercial  
software**

Our editorial policy is that we do NOT condone software piracy, but we do believe that 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 programs to meet his or her needs.

Furthermore, the copyright laws guarantee your right to such a DEPROTECTED backup copy:

... "It is not an infringement for the owner of a copy of a computer program to make or authorize the making of another copy or adaptation of that computer program provided:

1) that such a new copy or adaptation is created as an essential step in the utilization of the computer program in conjunction with a machine and that it is used in no other manner, or

2) that such new copy or adaptation is for archival purposes only and that all archival copies are destroyed in the event that continued possession of the computer program should cease to be rightful.

Any exact copies prepared in accordance with the provisions of this section may be leased, sold, or otherwise transferred, along with the copy from which such copies were prepared, only as part of the lease, sale, or other transfer of all rights in the program. Adaptations so prepared may be transferred only with the authorization of the copyright owner."

United States Code title 17, §117 (17 USC 117)

# Mag & Disk Combo Subscription

■ You may upgrade your current subscription to a magazine & disk combination by sending \$5.50 (\$6.50 foreign) per remaining issue.

## Time To Renew?

■ Check your mailing label to see if you need to renew your subscription.

Use the order form on this page to renew, and don't forget to send your most current mailing label to help expedite your renewal

## Moving soon?

■ If you're moving, let us know at least 30 days in advance.

■ Issues missed due to non-receipt of Change-of-Address may be acquired at the regular back issue rates.

■ Remember, the Post Office does not forward third class mail unless requested. COMPUTIST is not responsible for replacing issues lost while forwarding order is in effect.

*Things To Do...*



*Subscribe  
to  
Computist*

■ Yes, I want to subscribe to COMPUTIST. Enclosed are funds for an annual (12 issue) subscription.

I am...  A new Subscriber  
 Renewing my current subscription  
 Changing my address (please include last mail label)

■ Subscription type:

- U.S. - \$32
- U.S./Canada/Mexico First Class - \$45
- U.S./Canada/Mexico First Class plus Library Disk - \$100
- All other Foreign - \$75
- All other Foreign plus Library Disk - \$140

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

Address \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_

Country \_\_\_\_\_ Phone \_\_\_\_\_

 \_\_\_\_\_ Exp. \_\_\_\_\_

Signature \_\_\_\_\_ CP49

U.S. Funds drawn on U.S. bank. Allow 4-8 weeks for first issue. Mail to: COMPUTIST; PO Box 110846-T; Tacoma, WA 98411 (206) 474-5750

*Could this be the LAST Diskbusters T-shirt ad to appear in COMPUTIST?*

# COMPUTIST Diskbusters T-shirt

■ previously marketed as:

- CMPC (Computist MultiPurpose Cloth)
- How To Pick Up Girls
- Floppy-Disk coated coat
- We ain't afraid of no disk
- It's summer shirt
- All of the above and more...

Yes, I want \_\_\_\_\_ COMPUTIST Diskbusters T-shirts for \$9.95 each.

SIZES (adult mens): \_\_\_\_\_ Small \_\_\_\_\_ Medium \_\_\_\_\_ Large

..... LIMITED QUANTITIES: Offer good while supply lasts.....

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

Address \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_

Country \_\_\_\_\_ Phone \_\_\_\_\_

 \_\_\_\_\_ Exp. \_\_\_\_\_

Signature \_\_\_\_\_ CP49

Send check/money order to: COMPUTIST; PO Box 110846-T; Tacoma, WA 98411 WA state add 7.8% sales tax.  
Foreign orders add 20% shipping/handling. US funds drawn on US bank.

# Editorial.....

## ● RDEX 49

Here we are in COMPUTIST 49 and the response to our new format is overwhelmingly positive. It's interesting how the magazine changes when you no longer have a paid staff and instead use all volunteer help. It seems that volunteers put more of themselves into what they do. It's not just a job.

We received a few very negative letters on the loss of the hard cover but the overall opinion is positive. Most of you felt that the hard cover was nice but 16 more pages and the new RDEX will make it easy to live with the loss.

## ● Alphabetized and paged

We're working the bugs out of RDEX. In this issue, the softkeys are listed in alphabetical order with page numbers instead of sequential numbers. More of you are writing RDEX and sending material on disk. Remember that sending your letters on disk increases your chances of being published. It also keeps typing errors to a minimum.

## ● The hardware corner

I need to explain something about the hardware corner. The tidbits that we published were just that, tidbits. The full projects were held back until we could find out just how you felt about us using space for construction articles. The letters have been trickling in. The tone is positive but not enough of you seem interested. If I'm wrong then you need to write and let me know now before the whole program is shelved.

## ● Keep right on writing to us

And that brings up something else. All of you subscribers, collectively, control this magazine. By your willingness to sit down and write letters, you decide what COMPUTIST will be covering in the upcoming issues. While we cannot answer all your letters individually, we do read every one and your opinions move this magazine. If your happy, if your annoyed or if you just think something should be changed, write and let us know.

## BUGS

### COMPUTIST 45

- Page 5: *Flight Simulator Troubles*. The response to this letter can be found on page 24, instead of page 28.

### COMPUTIST 46

- Page 20: *Shape Magic* BASIC program listing lines 1100-1140. Replace the up arrow symbol with a percent ( % ) sign to indicate integer variables.
- Page 23: *Amazing Computer Facts*. The start hex location for the Delay Remover routine is 0280, not 1280 as shown.
- Page 26: *Science Toolkit* (column 3). In the code move routine, line 951 and 954 should read:  
951: CE 4A 09  
954: CE 4D 09
- Page 27: *Step by Step*. In step 1 replace C600G with 9600G
- Page 27: *Science Toolkit* controller (column 2) Line 1120, add ( G'' ) to the end of the line.

### COMPUTIST 47

- Page 32: *Alternate Realty*. In step 3, the command should read:  
1900<800.FFFM

### COMPUTIST 48

- Page 37: *Transylvania* controller. The end of line 1025 should read: GOSUB 2000:GOTO 1025

## RDEX contributors:

names	page
Balloonist, La	28
Basford, Mike	14
Bauer, John, III	29
Bilgrien, Steve	34
Brett, B. Dudley	26
Busch, John	33
Cambre, Donnie	24
Cracking Company	25
Crawford, Michael	24
David, Michael	21
Dowst, Henry P.	29
Gagnon, Fay	32
Garland, Steve	34
Garrett, Johnny	24
Grabenstein, Mike	23
Hart, Jim S.	34
Johnson, Tom	30
Jones, Ralph L.	22
Kesslick, Robert M.	22
L., S.	33
Lord, Spectrum	27
M., D.	25
Moravetz, Jack	32
Mueller, James E.	30
Mundel, Ron	34
Onymous, Ann	27
Ownby, Matt	30
Parker, Keith	29
Poulos, Greg	20
Ruth, Clay	31
Troha, Brian A.	15
Vytiaco, Donato 'Mousse'	27

in alphabetical order.....page

Addition Logician.....21  
 Animate.....20  
 Arcade Boot Camp.....25  
 Arctic Fox.....16, 33  
 Bard's Tale II.....29  
 Cat'n Mouse.....32  
 Counting Critters.....21  
 Dam Busters †.....19  
 Destroyer.....18  
 Draw Plus v1.0.....15  
 Dr. Ruth's Comp. Game Of Good Sex.....23  
 Echo 1.0.....19  
 E.D.D. 4.....16  
 Gamemaker.....32  
 Hard Ball.....18  
 Infiltrator.....24  
 List Handler †.....14  
 Locksmith 6.0 Fastcopy †.....27  
 Magic Slate.....26  
 Math Critters.....21  
 Millionaire.....33  
 Mind Mirror.....27  
 One On One.....24  
 Paintworks Plus v1.0.....15  
 Paintworks Plus v1.1.....16  
 PHM Pegasus.....21  
 Portal.....33  
 Quotient Quest.....21  
 Reader Rabbit.....33  
 Saunder's Chemistry CAI.....23  
 Science Toolkit.....26  
 Shanghai.....29  
 Strip Poker †.....23  
 Super Bunny.....18  
 Super Sunday.....25  
 Swordthrust series †.....22  
 Term Paper Writer.....33  
 Thief.....19  
 Top Fuel Eliminator.....33  
 Typing! †.....28  
 Up-n-Down.....20  
 Willy Byte.....31  
 Writer's Choice - Elite v1.0.....15  
 Writing A Character Sketch.....21  
 Writing A Narrative.....21

† Sofikey requires some other issue of COMPUTIST or is incomplete as published in this issue.

# COMPUTIST

Issue 49

November 1987

Publisher-Editor: Charles R. Haight Graphics Editor: Ryuji Circulation: Karen Fitzpatrick  
 Advertising: (206) 474-5750 Publishing: SoftKey Printing: Valco Printing

## Features:

### 6 Eliminate some ProDOS erroneous error messages

by Steve Marvin

This is a ProDOS enhancement feature... or Bug fix... for versions 1.1.1, 1.0.1, 1.0.2, and ProDOS 8.

### 8 Date/time without a clock card

by Steve Marvin

If you have a clock card, don't read this article. If you don't have the card and you want your files 'time stamped', then Steve's PRO.SET.DATE program is just what you need.

### 10 Sector surgery

by Philip Goetz

How to reconstruct the beginning of a sector and recover your lost data and valuable files. This lesson in DOS 3.3 first-aid will help you when your disk dies.

### 12 Generating Applesoft programs 'on-the-fly'

by Philip Goetz

If you want to add lines to a running Applesoft program or if you need a program that can write an entire Applesoft program, this utility is for you.

### 35 Product Monitor

by Jeff Hurlburt

A review of current software packages and books, including: *Wrath Of Denethenor*, *Graphics Scrapbook*, *Realms Of Darkness*, *Saracen*, *Award Maker Plus*, *Paintworks Plus*, *221B Baker St.*, *Logo Data Toolkit*, *Logo Language*, *Poker Night*, *Portal*, *Shanghai*...

## Departments:

4 - Editorial 4 - Bugs 14 - Readers Data EXchange 34 - Most Wanted Softkeys

## Miscellaneous:

How to convert List Handler files into standard text files—14 How to make GRAPHIC.GRABBER v3 run on the IIGs—22 Laser 128 'absolute' RESET—34 *Playing Tips for:* •Bard's Tale II—30 •Conan—30 •Donkey Kong—29 •Hacker I—27 •Hard Hat Mack—33 •Orbitron—33 •Print Shop Companion—33 •Spellbreaker—21 •Spy Hunter—33 •Ultima 4—30,34 *A.P.T.s for:* •Infiltrator—25 •Lode Runner—28 •Montezuma's Revenge—34 •Swordthrust series—22

- Address all advertising inquiries to: COMPUTIST Advertising Department PO Box 110816 Tacoma, WA 98411
- Mail manuscripts or letters to: COMPUTIST PO Box 110846-K Tacoma, WA 98411.
- Unsolicited manuscripts are assumed to be submitted for publication with all and exclusive rights belonging to COMPUTIST. For more information on submitting manuscripts, consult the writer's guide on the inside of the back cover.
- Entire contents copyright 1987 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 volunteer and paid 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 COMPUTIST magazine or SoftKey Publishing.
- Apple usually refers to an Apple II computer and is a trademark of Apple Computers, Inc.
- SUBSCRIPTIONS: Rates (for 12 issues): U.S.—\$32 U.S. 1st Class—\$45 Canada & Mexico—\$45 Other Foreign—\$75
- Direct subscription inquiries to: COMPUTIST Subscription Department PO Box 110846-T Tacoma, WA 98411
- 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.

# proDOS

versions 1.1.1, 1.0.1, 1.0.2, & 8

# Eliminate Some Erroneous Error

by Steve Marvin

## ProDOS Enhancement

( Bug fix )

Have you ever seen these messages:

```
*** UNABLE TO FIND "X.SYSTEM" ***
*** UNABLE TO LOAD "X.SYSTEM" ***
```

...No? Try renaming all the '.System' files on a ProDOS Disk removing the '.System' and then boot the disk. The Basic Interpreter Loader (the BI Loader is a part of the file ProDOS) refuses to recognize that ProDOS has successfully loaded, relocated and has a section of code called the QuitCode to handle loading interpreters. Instead of calling the QuitCode (and allowing you to select a SYS file to use), the BI Loader just locks-up until you reset!

This is one of several cases in Apple Software (not that I'm Anti-Apple, but this situation should not have lived through three revisions of ProDOS), where the left hand refused to see the work already done by the right hand (on the same sheet of paper!).

What should happen is this: The BI Loader should display an error message (to show you what went wrong), then wait a few seconds (to let you read the message), and then exit to the QuitCode (to allow you to select a SYS file from that disk or another). The patch to do this is short and simple really, I had to steal a few bytes from two of the messages, and it has to be changed for each version of ProDOS (but once it's BSAVED you're all set until you upgrade to a later version).

At the end of this article are abbreviated listings of the patch for ProDOS 1.0.1 (Table 3), 1.1.1 (Table 1) and ProDOS8 ver1.3 (Table 2. Yes the same method was used here).

It is best to BLOAD PRODOS, verify the code and enter the patch using the mini-assembler.

The patch also clears up bugs in the BI Loader which were found during dis-assembly.

Incidentally - Let me place a Plug here for *Autoworks* by Alan Bird. I don't use *Appleworks* enough to need the macro's or other add-ons, but *PRODOS8* and *Birds Better Bye* are worth the price. All ProDOS users have experienced the aggravation of trying to remember a Prefix and PathName for an application so that you don't have to Re-Boot.

What an idiot subroutine the Apple QuitCode is. It resides in one bank of the upper 16K of RAM and is accessed by a call to the Machine Language Interface (MLI). Alan Bird re-wrote this code into a menu-like application loader. It locates for you all files of type SYS on each volume, all you do is select the volume/disk with the escape key, and select the file to use from the list on the screen using the up/down arrow keys, then Press **RETURN** to load it. If you have selected a Directory entry (which is shown by a leading '/') the prefix is reset and the SYS files in it are listed. All volumes are available, and all SYS files accessible by this method.

### Brief overview - BI Loader

ProDOS enters the BI Loader at \$800 after the Kernel has been relocated and initialized in high RAM. At this point ProDOS is ready to go, and the Volume Directory (beginning at block 2) has been placed just above the BI Loader for use in locating a '.SYSTEM' file.

The BI Loader first scans the Volume Directory in RAM for the first file of type SYS (\$FF) with a filename ending in '.SYSTEM'. This is the User interface or Interpreter file (Can be BASIC.SYSTEM or WP.SYSTEM or APLWORKS.SYSTEM...). The file is OPEN'ed and the length is checked by the call GET\_FILE\_EOF in the MLI. The file is loaded at a starting address of \$2000, the motherboard ROM is enabled and the loader exits by JMP \$2000.

Errors display a brief message in the center

of the ProDOS logo and the machine locks-up until **RESET** (⏏). The messages are:

```
*** CAN'T FIND A ".SYSTEM" FILE ***
*** UNABLE TO LOAD X.SYSTEM ***
** SYSTEM PROGRAM TOO LARGE **
```

(X is expanded to be the name of the file and these are abbreviated messages).

The only difference in ProDOS8 is that before the first '.SYSTEM' file is found, a file called ATINIT of type \$E2 is found, loaded and called using the same sequence (open, checksize, read) except that a JSR is used vs a JMP to \$2000.

In ProDOS 1.0.1, 1.0.2, 1.1.1 there was a bug in the subroutine to display the 'Program too large' message, (the wrong register is decremented and the message may or may not display).

In ProDOS8 the bug was that one of the message subroutines referenced or JSR'd areas in \$23XX instead of \$8XX, both ATINIT and the 'X.SYSTEM' file use this memory, so there is a good chance it will not do what is intended when JSR'd to.

The patch shown may require changes for your version, I have included three versions and a listing detailing the alterations. In each listing the italicized address is the relocated address in the \$800 page. Enter the patch at the \$2XXX address using the mini-assembler after BLOAD PRODOS, A\$2000, TSYS:

### ProDOS 1.1.1

**BLOAD PRODOS, A\$2000, TSYS**

Enter the patch in Table 1

**BSAVE PRODOS, A\$2000, TSYS**

### ProDOS 1.0.1, 1.0.2

Same code, same relocated address, but different source location:

**BLOAD PRODOS, A\$2000, TSYS**

Enter the patch in Table 3.

**BSAVE PRODOS, A\$2000, TSYS**

# ProDOS

## Messag es

### ProDOS8

This is a little easier to do because there is more room. The bug is in the message routine to display "ATINIT LOAD ERROR". It is JMPed to at \$23XX instead of \$08XX and uses LDX \$233C and LDA \$233C,X. If ATINIT is partially read in, it is possible that this code will not be there. The fix is included here:

**BLOAD PRODOS,AS\$2000,TSYS  
CALL -151**

Enter the patch in Table 2

**Table 3 for ProDOS 1.0.1, 1.0.2**

Existing	Patch
22A6:	BNE 22EE
22AE:	BNE 22EE
22B3:	BNE 2308
22BA:	BCS 2308
22CF:	BEQ 2308
22D1:	BNE 22EE
22D9:	BNE 22EE

.....Patches for subroutines

22E1: LDY #\$1B  
BNE 2303

**22E5<22EE.2307M**

22FD: BMI 230D  
22FF: LDY #\$36  
LDX #\$1B  
LDA 8F4,Y  
STA 7AE,X  
DEY  
DEX

BPL 2319  
230D: STA C082  
LDX #\$20  
LDA #0  
JSR FCA8  
DEX  
BNE 2312  
JSR BF00  
65 ED 08 00  
04 00 00 00 00 00 00

2328: Messages exactly as in Table 1

(NOTE: type the messages exactly as shown. The easiest way is to directly insert the text if you have the enhanced monitor, 'S 'Y 'T like that but any other method is ok, just the length is critical)

**Table 1 for ProDOS 1.1.1**

Existing	Patch	Comment
872 22DE:	BNE 8BA 2326	BNE 231D ;CAN'T_LOAD_SYS_FILE
87A 22E6:	BNE 8BA 2326	BNE 231D
87F 22EB:	BNE 8D4 2340	BNE 2337 ;PROGRAM_TOO_LARGE
886 22F2:	BCS 8D4 2340	BCS 2337
89B 2307:	BEQ 8D4 2340	BEQ 2337
89D 2309:	BNE 8BA 2326	BNE 231D
8A5 2311:	BNE 8BA 2326	BNE 231D
8AD 2319:	LDX #\$27	8AD 2319: LDY #\$1B
231B:	LDA \$8E2,X	BNE 2339
	STA \$7A8,X	8B1 231D: LDY \$94F
	DEX	(Use 231D<2326.233EM to move
	BPL \$231B	a section of code down here)
	BMI 8DF 234B	
		2335: BMI 2345
		8CB 2337: LDY #\$36
		8CD 2339: LDX #\$1B
		LDA \$8F4,Y
		STA \$7AC,X
		DEY
		DEX
		BPL 233B
8BA 2326:	LDY \$94F	8D9 2345: STA \$C082 ;Read from ROM
	LDA #\$27	LDX #\$20
	...	234A: LDA #0
8D0 233C:	BPL 2334	JSR \$FCA8 ;Wait a While
	BMI 8DF 234B	DEX
8D4 2340:	LDY #\$1E	BNE 234A ;And some More
	LDA \$90A,Y	JSR \$BF00 ;Call the MLI
	STA \$7AD,Y	\$65 ;Code for QuitCall
	DEX	;BUG
	BPL 2342	\$ED \$08
		00
8DF 234B:	BMI 8DF 234B	8ED 2359: 04 00 00 00 00 00 00
	BRK	8F4 2360: '* ".SYSTEM" FILE NOT FOUND *'
		237C: ' ".SYSTEM" FILE TOO LARGE *'
8E2 234E:	'** CAN'T FIND SYSTEM **'	
90A	'** PROGRAM TOO LARGE **'	

**Table 2 for ProDOS 8**

22E9<22EA.232FM		;Make one byte of space
22E1:90 06	BCC 80E	;Fix Branch
22FE:D0 2F	BNE 853	;Fix Branch
232E:A2 11	LDX #\$11	
2330:BD 77 08	LDA 877,X	;ATINIT Load Error
2333:9D B2 07	STA 7B2,X	
2336:CA	DEX	
2337:D0 F7	BNE 2330	
2339:8D 82 C0	STA C082	;Read Enable ROMs
233C:A2 20	LDX #\$20	
233E:A9 00	LDA #0	
2340:20 A8 FC	JSR FCA8	;Delay Awhile
2343:CA	DEX	
2344:D0 F8	BNE 233E	
2346:20 00 BF	JSR BF00	;CALL QuitCode
2349:65 71 08		
234C:04 00 00 00 00 00 00		
2353: '*ATINIT LOAD ERR*'		
2430:A2 26	LDX #\$26	
2432:BD 8A 09	LDA 98A,X	
2435:9D A9 07	STA 7A9,X	;Save another Byte
2462:4C 5E 08	JMP 085E	;Delay and Quit



A Utility For...

proDOS

# DATE / TIME Without A Clock Card

by Steve Marvin

**IF YOU HAVE A CLOCK CARD  
DON'T READ THIS ARTICLE!**

For those of us who don't have a Clock Card but still like our files to be date-stamped, some of the applications (such as *Appleworks*) ask for the date each boot. *Appleworks* saves the date on one of the SEG.XX files so it always knows the last time it was used. You can preset the date/time using the STARTUP program from the *ProDOS Users Disk* or Beagle Brothers DATE.SET or a BASIC/binary combo called *ProDater*, but all of these require the presence of the Basic Interpreter (BI) which takes up a lot of disk space, and they all start from scratch (by which I mean, they don't save the date/time anywhere on disk). Most of the newer applications have little disk space to spare. *Word Perfect* has a few blocks, but not enough for the BI and a DATE.SET program.

Alternately, you can usually set the date/time from within the programs (Not on all applications) but it just seems more convenient to me to set it up at boot-time, I particularly like the way *Appleworks* does it, so I made my utility perform a lot like that.

There are three ways that I could see to avoid using BASIC.

1- Add the utility to ProDOS. I used this on ProDOS 1.0 and 1.1.

2- Make it the first ".SYSTEM" file on the disk and somehow search for the second one to exit to. This method ended up taking too much space on disk.

3- Make it the first ".SYSTEM" file and build in the Prefix and Pathname to exit to. This would require installation on each disk separately.

In the process of writing and debugging this utility as a ProDOS add-on, I discovered that

ProDOS8, which I received with *Autoworks* by Alan Bird, and added to my *Word Perfect* and *AppleWorks* disks, calls a file named ATINIT at \$2000 just prior to loading the first ".SYSTEM" file. This must be intended for initialization of the *AppleTalk* Network (thus the name).

Restrictions are: file type must be \$E2, must execute at \$2000, must return with an RTS, should not use RAM below \$2000 (to preserve the BI Loader and the Volume Directory blocks placed there). Other than those, the program can do anything you want it to do. This made it easy for me, and resulted in a 3-block file of type \$E2, and named ATINIT.

## Operation of the utility:

The program uses the last line on the screen, and zero page locations \$70..\$77, these are preserved and restored at the end of the run, no other RAM below \$2000 is used. The data and code take from \$2000 to \$234F and the 1 block (2-sector) data buffer from \$2400.\$25FF.

First the Volume Directory block 2 is read, errors at this point simply exit after restoring the screen. The date in the Global Page at \$BF90.\$BF93 is compared to the volume creation date at Buffer+\$1C. The most recent date/time is placed in the Global Page for use. If the volume date is NOT the most recent, then a "Must.Write" flag is set to ensure update of the disk.

Now the ProDOS-format 4-byte date/time is put into human terms and displayed as MM/DD/YY HH:MM AM (Month,Day,Year Hour,Minute AM/PM). Input is accepted via a subroutine which supports both Left and Right Arrow for motion within the buffer, **RETURN** to accept, **ESC** to restore the original value and a second **ESC** to exit without updating the disk.

When **RETURN** is pressed, the date/time in the buffer is parsed and checked for validity.

Any separator between fields is accepted, two

digits per field are required, 12 or 24 hour format is accepted.

All fields are checked for low/high limits including the correct maximum days in month (February-Leap Year).

If the hour is 12 to 23 the PM indicator is set and the hour is adjusted to show 00-12.

The date/time line is re-written to put back the correct separators and show the adjustment in Hours and AM/PM indicator. The date-time is converted to ProDOS format and stored in the Global Page.

If any change has been made, the "Must.Write" flag is set for update.

If the "Must.Write" flag has been set, the Volume Directory block is re-read (in case you swapped disks during the operation, this prevents trashing the Volume Directory).

The date is moved in to the buffer and re-written to block 2.

Any errors are internally trapped: disk write-protected, door open, bad sector...

Now the screen line and zero page are replaced as they were on entry. The cursor is placed at (1,1) on the screen by TABV in the System Monitor and the program exits as follows:

```
ATINIT:
  RTS      Return to the BI Loader
ProDOS add-on:
  JMP 800  Enter the BI Loader
BRUN Program:
  JSR BEF8 CALL FreeBuffer
  RTS Return
```

I know how much you all like entering hex-dumps, and three sectors is hardly anything at all, so I won't give you the assembler source code

(Actually, I didn't use one; I built this utility from scratch).

I included a brief listing of the Address, and Function of each subroutine and the entry points in the main code so that you can add anything you want to it.



**A) Boot ProDOS**

Enter the PRO.DATE.SET hexdump and check it with CheckBin.

**BSAVE PRO.DATE.SET,AS2000,LS34F**

If You have the *Library Disk*, copy PRO.DATE.SET to your ProDOS disk.

**B) ProDOS 8: CREATE ATINIT,T\$E2**

**BSAVE ATINIT,AS2000,LS34F,T\$E2**

(If you don't like using the name ATINIT, you can rename the file, be sure to change the pathname 'ATINIT' in ProDOS. You can also change the error message \*\*\* UNABLE TO LOAD ATINIT \*\*\* and the file type if you wish \*)

**C) ProDOS 1.0/1 or other version:**

1. Determine the size of ProDOS and the address of the final jump to the BI Loader at \$800. ProDOS 1.0.1 and 1.0.2 end at \$5BFF and the JMP is at \$2127. ProDOS 1.1.1 ends at \$59FF and the JMP is at \$21C0.

You can check out your version this way:

**CALL-151**

**5000:11 N 5001<5000.6FFFF**

**BLOAD PRODOS,AS2000,TSYS**

Locate the first free sector by finding a sector of \$11's. Locate the JMP \$800 by searching for the bytes 4C 00 08.

Change the JMP to JMP RELOC (See below) (If your free sector was \$5A00 then JMP \$5D4B [EOF + \$34B])

BLOAD PRO.DATE.SET,AS5A00 or your first free sector and modify the end of PRO.DATE.SET, add short relocater at \$5D4B

\$5D47: 18 CLC

*Go to BI Loader*

\$5D48: 4C 00 08 DONE: JMP \$0800

*Utility Relocator*

\$5D4B: A0 00 RELOC: LDY #0  
 \$5D4D: B9 00 5A\* MOVE: LDA \$5A00,Y  
 \$5D50: 99 00 20 STORE: STA \$2000,Y  
 \$5D53: C8 INY

**2000<5A00.5DFFM**

\$5D54: D0 F7 BNE MOVE  
 \$5D56: EE 4F 5D\* INC MOVE+2  
 \$5D59: EE 52 5D\* INC STORE+2  
 \$5D5C: AD 52 5D\* LDA STORE+2  
 \$5D5F: C9 24 CMP #\$24  
 \$5D61: D0 E8 BNE RELOC

*ReadEnable ROMS*

\$5D63: 8D 82 C0 STA \$C082

*Go to PRO.DATE.SET*

\$5D66: 4C 00 20 JMP \$2000

**BSAVE PRODOS,AS2000,LS3D69,TSYS**

Replace all \* bytes and Length with the correct values for your version of PRODOS.

**PRO.DATE.SET Abbreviated Listing**

\$2000: JMP ENTRY  
 \$2003: LINEBUFFER  
 \$202B: RWParmList  
 \$2031: DTTABLE M,D,Y,H,m,'A'  
 \$2037: MUST.WRITE

\$2038: DISK.READ/ONLY  
 \$2060: DISK.ERROR  
 \$2088: ZPAGE.BUFFER  
 \$2090: CLEAN.LINE  
 \$20A0: XCHG.MSG.LINE  
 \$20B7: HEXBYTE.TO.ASCII.DEC  
 \$20CB: GET.NEXT.DECIMAL+  
 \$20E9: GET.NEXT.DIGIT  
 \$20F5: CHECK.VALID.DIGIT  
 \$2102: LOCATE.AM.OR.PM  
 \$2116: INIT.DATE.TIME  
 \$214A: PROMPT.INPUT.DATA  
 \$2153: CONVERT.FROM.PRO.DTIME  
 \$2195: SHOW.DATE.TIME  
 \$21CE: GET.AND.CHECK.DATA  
 \$2287: CONVERT.TO.PRO.DATETIME  
 \$22B8: WRITE.VOL.CREATE.DATE  
 \$22FB: ENTRY  
 \$2338: RESTORE.LINE  
 \$2340: ESCAPE.EXIT  
 \$2347: CLC  
 \$2348: RTS

2140: A0 04 B1 70 91 72 88 D0 \$4541  
 2148: F9 60 20 53 21 20 CE 21 \$9CCE  
 2150: B0 F8 60 08 78 A2 C1 AD \$F744  
 2158: 93 BF C9 0C 90 09 A2 D0 \$4DA7  
 2160: C9 0D 90 03 38 E9 0C 8D \$CD83  
 2168: 34 20 AD 92 BF 8D 35 20 \$7FE3  
 2170: AD 90 BF 8D 31 20 29 1F \$C4C4  
 2178: 8D 32 20 AD 91 BF 4A 8D \$6A32  
 2180: 33 20 6E 31 20 AD 31 20 \$1E80  
 2188: 29 F0 4A 4A 4A 4A 8D 31 \$D12E  
 2190: 20 8E 36 20 28 A9 17 20 \$FE9E  
 2198: 5B FB A9 E6 85 74 A9 07 \$713F  
 21A0: 85 75 A0 0F B9 90 20 91 \$5CC0  
 21A8: 74 88 10 F8 AD 33 20 F0 \$7D0A  
 21B0: 1C A0 00 B9 31 20 20 B7 \$39F0  
 21B8: 20 91 74 E6 74 8A 91 74 \$700B  
 21C0: E6 74 C8 C0 05 D0 EC AD \$F62E  
 21C8: 36 20 8D F5 07 60 A2 16 \$ECF9  
 21D0: 86 24 20 0C FD A6 24 C9 \$A6EF  
 21D8: 9B D0 03 E0 17 60 C9 8D \$6C43  
 21E0: F0 19 C9 88 D0 07 E0 16 \$78B2  
 21E8: F0 E6 CA D0 E3 C9 95 F0 \$3D25  
 21F0: 03 9D D0 07 E8 E0 26 D0 \$DDD1  
 21F8: D7 F0 EF A2 16 A0 00 84 \$A23C  
 2200: 77 20 CB 20 B0 67 A4 77 \$F6B0  
 2208: 99 31 20 C8 C0 05 D0 EF \$C160  
 2210: 20 02 21 B0 03 8D 36 20 \$FC4B  
 2218: AC 31 20 F0 50 C0 0D B0 \$1F4C  
 2220: 4C AD 33 20 C0 02 D0 0B \$581F  
 2228: 29 FC CD 33 20 A9 00 69 \$B32E  
 2230: 1D D0 0A A9 AB 18 2A 88 \$756E  
 2238: D0 FC 98 69 1F 85 76 AD \$5831  
 2240: 32 20 F0 29 C5 76 B0 25 \$D36D  
 2248: AD 34 20 C9 0C 90 13 08 \$C1C5  
 2250: A0 D0 8C 36 20 28 F0 0A \$D307  
 2258: C9 18 B0 11 38 E9 0C 8D \$40B6  
 2260: 34 20 AD 35 20 C9 3C B0 \$2CF7  
 2268: 04 A9 8D 18 60 A9 9B 38 \$95D5  
 2270: 60 A2 07 BC 88 20 B5 70 \$CC0C  
 2278: 9D 88 20 98 95 70 CA 10 \$414A  
 2280: F2 60 D3 D4 C5 D6 C5 08 \$CDCC  
 2288: 78 AD 34 20 AE 36 20 E0 \$F9A3  
 2290: D0 D0 03 18 69 0C 8D 93 \$F98F  
 2298: BF AD 35 20 8D 92 BF AD \$DDFA  
 22A0: 31 20 0A 0A 0A 0A 0A 2E \$BCD4  
 22A8: 33 20 6D 32 20 8D 90 BF \$6112  
 22B0: AD 33 20 8D 91 BF 28 60 \$A6E2  
 22B8: AD 37 20 F0 3D 20 00 BF \$F098  
 22C0: 80 2B 20 B0 11 A0 04 B1 \$5AAE  
 22C8: 72 91 70 88 D0 F9 20 00 \$8F2E  
 22D0: BF 81 2B 20 90 24 A2 38 \$5316  
 22D8: C9 2B F0 02 A2 60 86 74 \$B221  
 22E0: A0 27 84 24 20 A2 20 20 \$3DAB  
 22E8: 3A FF 20 0C FD C9 9B F0 \$6D08  
 22F0: 09 C9 8D D0 F2 20 A2 20 \$8464  
 22F8: 30 C3 60 AD 30 BF 8D 2C \$B14B  
 2300: 20 AD 2E 20 8D 89 20 20 \$C4E2  
 2308: 00 BF 80 2B 20 B0 31 20 \$400F  
 2310: 71 22 A9 03 20 A0 20 20 \$D9AF  
 2318: 16 21 20 4A 21 C9 9B F0 \$AFFC  
 2320: 17 20 95 21 20 87 22 A0 \$A15B  
 2328: 04 B1 72 D1 70 F0 03 CE \$FF7F  
 2330: 37 20 88 D0 F4 20 B8 22 \$CC67  
 2338: A9 03 20 A0 20 20 71 22 \$5175  
 2340: A9 00 85 24 20 5B FB 18 \$1FF3  
 2348: 60 00 \$F52D

**PRO.DATE.SET Hexdump**

2000: 4C FB 22 C5 EE F4 E5 F2 \$8699  
 2008: A0 F4 E8 E5 A0 C4 E1 F4 \$7518  
 2010: E5 AF D4 E9 ED E5 BA A0 \$0B13  
 2018: A0 CD CD AF C4 C4 AF D9 \$0F94  
 2020: D9 A0 C8 C8 BA CD CD A0 \$EF20  
 2028: C1 CD A0 03 60 00 24 02 \$A14D  
 2030: 00 00 00 00 00 00 C1 00 \$939C  
 2038: D2 E5 ED EF F6 E5 A0 D7 \$51ED  
 2040: F2 E9 F4 E5 AD D0 F2 EF \$C6B5  
 2048: F4 E5 E3 F4 A0 D4 E1 E2 \$0A62  
 2050: A0 A6 A0 D0 F2 E5 F3 F3 \$8E73  
 2058: A0 D2 E5 F4 F5 F2 EE A0 \$034F  
 2060: C3 E8 E5 E3 EB A0 C4 E9 \$9F0B  
 2068: F3 EB AF C4 F2 E9 F6 E5 \$4C16  
 2070: A0 A6 A0 D0 F2 E5 F3 F3 \$2847  
 2078: A0 D2 E5 F4 F5 F2 EE AF \$4A3C  
 2080: C5 F3 E3 E1 F0 E5 A0 A0 \$32C1  
 2088: 1B 24 8F BF 00 00 00 00 \$9152  
 2090: CD CD AF C4 C4 AF D9 D9 \$A7BA  
 2098: A0 C8 C8 BA CD CD A0 C1 \$0B86  
 20A0: 85 74 A9 20 85 75 A0 27 \$8FB6  
 20A8: B1 74 BE D0 07 99 D0 07 \$C53B  
 20B0: 8A 91 74 88 10 F2 60 A2 \$716D  
 20B8: B0 38 E9 0A 90 03 E8 B0 \$346A  
 20C0: F9 69 0A 09 B0 85 76 8A \$6100  
 20C8: A6 76 60 20 E9 20 B0 14 \$4234  
 20D0: 29 0F A8 E8 20 F5 20 B0 \$FE14  
 20D8: 0D 29 0F 18 C8 88 F0 04 \$ACF2  
 20E0: 69 0A 90 F9 E8 60 98 18 \$BD06  
 20E8: 60 20 F5 20 90 06 E8 E0 \$F150  
 20F0: 26 D0 F6 38 60 BD D0 07 \$3BF5  
 20F8: C9 BA B0 05 C9 B0 90 14 \$66EC  
 2100: 18 60 BD D0 07 29 DF C9 \$0CDD  
 2108: C1 F0 F5 C9 D0 F0 F1 E8 \$73A4  
 2110: E0 26 D0 EE 38 60 A0 01 \$A83C  
 2118: B1 70 D0 03 CE 37 20 B1 \$18ED  
 2120: 72 F0 1D A0 04 B1 72 D1 \$FEEC  
 2128: 70 90 15 D0 05 88 D0 F5 \$C106  
 2130: F0 17 CE 37 20 A0 04 B1 \$4712  
 2138: 72 91 70 88 D0 F9 F0 09 \$C306



*You don't need to call a disk doctor when your disk dies. It's just time for you to do some...*

## A lesson in DOS 3.3 first aid

# SECTOR

## How To Reconstruct And Recover Your

by Philip Goetz

### ■ Requirements:

- nibble editor
- COPYA-type copy program
- sector editor helpful

### Not all I/O errors are total losses

Sometimes you can read a bad sector just by first entering B942:18 from the monitor.

But you won't get off so easily if rewriting one sector erased the beginning of the next.

This is not an uncommon problem, especially if you use the same disk with different drives. **If you initialize with a slow drive and write with a fast drive - watch out.** Fortunately, you can often reconstruct the beginning of a sector and recover your data.

### Disk Format

Since DOS 3.3, data is stored on all Apple II disks in the same way physically. A track can be defined as follows:

**Track:** Sync gap, Sector 0, Sync gap, Sector 1, ..., Sync gap, Sector 15, Sync gap

**Sync gap:** A string of 10-bit bytes, 1111111100, used to bring a drive into sync with the bit stream.

Represented in nibble editors by FFs, sometimes in inverse. See *Beneath Apple DOS*, p. 3-7 through 3-9 for details.

**Sector:** Address field, Sync gap, Data field

**Address field:** D5 AA 96, Volume#, Track#, Sector#, Checksum, DE AA EB

**Data field:** D5 AA AD, 342 bytes data, Checksum, DE AA EB

Apple drives cannot read all values from a disk. They can only read bytes which have the high bit set and no more than one pair of consecutive zero bits (*Beneath Apple DOS*, p. 3-20).

So, data cannot simply be dumped to the disk; it must be encoded before writing and decoded after reading.

Since DOS 3.3 there have been 2 types of encoding used.

### 4-&-4 Encoding

One, called 4-and-4, is used in address fields, where RWTS needs to decode small amounts of data quickly. To encode a byte NN into 4-and-4 bytes XX YY, let XX = binary 1 B7 1 B5 1 B3 1 B1, YY = 1 B6 1 B4 1 B2 1 B0, where NN = B7 B6 B5 B4 B3 B2 B1 B0.

So, for instance, the address field for volume 254 track \$11 sector 0 would be D5 AA 96 FF FE AA BB AA AA FF EF DE AA EB. Don't worry about the checksum; we're going to wipe it out anyway.

### 6-&-2 Encoding

The other type of encoding, called 6-and-2, is used to turn 256 data bytes into 342 disk bytes. Don't worry about it.

### Identifying the Problem

The problem we are dealing with is when the sync gap written after a sector overruns the address field of the next sector.

Since RWTS cannot find the sector without its address field, it reports an I/O ERROR. We must reconstruct the address field and insert it in the sync gap between the two sectors.

To see if you have this problem, you must locate the track with the bad sector. You can

find it by reading sectors with a sector editor such as *Diskedit* (part of your Computist Starter Kit) or *The Inspector* until you get an error (perhaps in conjunction with a disk utility such as *Copy II Plus* which shows where each file is on the disk).

### DOS Tracer

A quicker way is to use my *DOS Tracer* program (*Nibble*, Sept. 86, p. 92-93) to find the bad sector.

**A word of warning:** For some reason *DOS Tracer* may make RWTS less tolerant of error, and so it may crash on an earlier, "good enough" sector. This has happened to me once with a bad file.

Once you know what track the problem is on, read it with a nibble editor (the kind which reads a track of raw, encoded data from the disk).

Most good nibble copiers have nibble editors. Make sure that yours will identify sync bytes and write them back as sync bytes. I used *Locksmith 4.1*.

The nibble editor option in *Locksmith 4.1* will not identify sync bytes, so you must use the disk copy algorithms. Select option 1 (copy disk) from the main menu. Specify the problem track as the range to be copied with manual error retry, and open the drive door of the destination drive so you get a verify error. Choose option 5 from the error menu (nibble-edit current track data) and you will have the fully analyzed track to work with.

### Logical vs Physical Sector Numbers

Don't look for the same sector number as the one you found bad. DOS deals with logical sector numbers, which are different from the physical sector numbers in the address fields. They are translated by an interleave table, which is different for DOS 3.3, ProDOS, Pascal, and CP/M, so forget the logical sector number.

# SURGERY

## The Beginning Of A Sector Lost Data And Valuable Files

Look for an abnormal sector.

**Note:** It is normal to have a lot of sync FFs between sectors 15 and 0.

If you find one with no address field and a few sync bytes in front, you're in luck. (kind of.)

### The Fix

If you have *Locksmith*, you'll have to move the sector before the missing address field up by about 8 bytes, overwriting some sync bytes, to make room for the address field, and you'll have to move the whole thing by hand, one byte at a time. (I thought computers were supposed to do this kind of thing for you.)

### Special *Locksmith* instructions

With *Locksmith*, type:

B

... to move to the beginning of the track and then use the:

J K M keys to move around,

S to make a byte a sync byte,

N to make a sync byte a normal byte.

C

... to enter the change mode.

In the change mode, type the value for a byte, space, the value of the next byte, and so on.

When finished, press:

RETURN

... to leave the change mode.

E

... will take you to the end of the track.

Be careful to do your editing between the beginning and the end of the track. Some nibble editors keep at least two images of the track in memory, but only write back the one between the beginning and end markers.

You should have at least 4 sync bytes in every gap between sectors (5 according to *Beneath*

*Apple DOS*, but they demand you be in sync before the last sync byte. See p. 3-9.)

The sync bytes between the address field and the data field are to give RWTS time to figure out if this is the right sector, so put a few more there if possible.

So, enter the previous sector's address field 4 or more sync bytes after the sector before it.

### You can even eliminate the checksum

If you're pressed for room you can replace that sector's 1-byte checksum and DE AA EB with sync bytes. Leave out the address field's 2-byte checksum and the DE AA EB; we'll tell RWTS not to look for them.

Follow with at least 6 sync bytes, and then the data field.

You will be entering the values that you see further down in memory, moving the whole sector up. It is easiest to move it up by 8 bytes, because then for each byte in the data field you can simply enter the byte immediately below it in the 8-byte wide display.

At the end of the data field, leave out the 1-byte checksum and the DE AA EB. Enter at least 4 sync bytes, duplicate the previous address field, and enter sync bytes up to the data field.

Before you do this you should count the number of bytes in the gap which overwrote the address field to decide how many more bytes you will need, and try to distribute them fittingly among the various sync gaps.

### Replace the blown sector number

Now you need to change the sector number to match the blown sector.

**1** Read the sector number from the previous address field (the 8th and 9th bytes of the address field) and decode it.

Add one to find the number for the new address field, and encode it. If the previous sector were number 15, the new one would be 0. Sector 15 would never overwrite sector 0

on a slow drive, but could on a fast drive.

The following table will help:

Table 1

Two-byte value	Decimal
AA AA	0
AA AB	1
AB AA	2
AB AB	3
AA AE	4
AA AF	5
AB AE	6
AB AF	7
AE AA	8
AE AB	9
AF AA	10
AF AB	11
AE AE	12
AE AF	13
AF AE	14
AF AF	15

**2** Now write the track back to the disk. In *Locksmith*, press ESC and choose option 2 (write track as-is).

**3** Boot a DOS 3.3 disk, type:

CALL-151  
B942:18

This tells RWTS not to look for checksums or DE AA EBs.

**4** Then run a copier such as COPYA which copies unprotected disks (don't use a nibble copier) to make a good copy with checksums and evenly spaced sync gaps.

**Copiers such as Locksmith 5 or 6's fast copy won't work because they will undo our modification.**

**6** Finally, run a sector editor to ensure you have recovered your data.



# Generating Applesoft Programs On—The—Fly

by Philip Goetz

Sometimes you may want to add lines to an Applesoft program while it is running, or you may want your utility to write an entire Applesoft program. You can do this by directing the input vector (\$38-39) to a short machine language program which hands BASIC the correct characters one by one, just as if they had been typed from the keyboard.

As a simple example, the following program writes and executes a canned Applesoft program. It is written for the S-C Macro Assembler.

## Listing 1

```

0300: A9 19 85 38 A9 03 85 39 $0D0A
0308: 20 EA 03 A9 00 8D 16 03 $52FF
0310: 8D 17 03 4C 00 E0 00 00 $FA9E
0318: 00 8E 18 03 AE 16 03 EE $6B67
0320: 16 03 BD 46 03 AE 18 03 $914D
0328: C9 8D F0 01 60 EE 17 03 $374D
0330: AD 17 03 C9 03 F0 03 A9 $BD1C
0338: 8D 60 20 89 FE 20 EA 03 $5742
0340: A9 8D AE 18 03 60 B1 B0 $4A11
0348: BF BA BF A2 D0 D2 C5 D3 $7132

0350: D3 A0 C1 CE D9 A0 CB C5 $D611
0358: D9 A0 C6 CF D2 A0 C1 A0 $ADAC
0360: C3 CF CC CF D2 A0 C4 C5 $12B9
0368: CD CF A2 BB BA C7 C5 D4 $3314
0370: C1 A4 8D B2 B0 C7 D2 BA $E721
0378: C6 CF D2 D8 BD B0 D4 CF $7714
0380: B3 B1 BA C3 CF CC CF D2 $3D7D
0388: BD D8 AF B2 BA D6 CC C9 $52E1
0390: CE B0 AC B3 B9 C1 D4 D8 $6F9F
0398: BA CE C5 D8 D4 8D D2 D5 $6541

03A0: CE 8D $8583
    
```

More complicated is making an Applesoft program add lines to itself. Say you have a curve graphing program which asks the user for an equation such as  $Y = X * \sin(X)$ , and you want to enter the equation as a line number. You can input the equation as a string, add a line number to its left end, pass it to a machine language routine based on the one above which enters the string as a direct Applesoft command, and restart your program on the line after the call. (Note that you can also enter other commands besides adding line numbers this way; but you can do most other things from within the program anyway.)

Unfortunately, adding new lines overwrites

some of the real and integer variables, and will invalidate the pointers to any strings which are stored within the program beyond the point where the new line was added. (The string AS in 110 AS = "HERE I AM" is not duplicated

in memory; instead, the string's pointers point right into the program code.) To avoid problems, Applesoft effectively erases all variables (including DIM'd array space) whenever you enter, change, or delete a line.

## Sourcecode for Listing 1

.AS -/HI THERE/ produces the negative ASCII (high bit set) for string "HI THERE".  
 .BS x tells the assembler to allocate x bytes at that point in the object code.  
 .HS produces a 'hex string', so .HS 8D places a \$8D in the object code.  
 #KEYINT & /KEYINT are the low and high bytes, respectively, of the word KEYINT.

```

* GENERATE & RUN APPLESOFT PROG *
1030 .OR $300
1040 CALLME LDA #KEYINT LOAD NEW INPUT VECTOR
1050 STA $38
1060 LDA /KEYINT
1070 STA $39
1080 JSR $3EA DOS ROUTINE TO SAVE OUR INTERCEPT
* TO BE CALLED AFTER DOS IS THRU *
1100 LDA #0
1110 STA BUFPTR
1120 STA LINCNT
1130 JMP $E000 COLDSTART BASIC
1140 BUFPTR .BS 1
1150 LINCNT .BS 1
1160 TEMP .BS 1 TEMPORARY STORAGE FOR X
1170 KEYINT STX TEMP SAVE X
1180 LDX BUFPTR
1190 INC BUFPTR
1200 LDA PROGRAM,X
1210 LDX TEMP RESTORE X
1220 CMP #$8D RETURN
1230 BEQ .1
1240 RTS
1250 .1 INC LINCNT COUNT LINES SO WE KNOW WHEN WE'RE DONE
1260 LDA LINCNT
1270 CMP #3 2 LINES + RUN
1280 BEQ .2
1290 LDA #$8D
1300 RTS
1310 .2 JSR $FE89 PUT BACK NORMAL VECTOR
1320 JSR $3EA & TELL DOS
1330 LDA #$8D CR
1340 LDX TEMP RESTORE X
1350 RTS
1360 PROGRAM .AS -/10?:?' PRESS ANY KEY FOR A COLOR DEMO";:GETA$/
1370 .HS 8D
1380 .AS -@20GR:FO RX=@T031:COLOR=X/2:VLIN0,39ATX:NEXT@
1390 .HS 8D
1400 .AS -/RUN/
1410 .HS 8D
    
```

..... end of sourcecode .....

Thus, this technique is unlikely to be useful in the middle of a long program.

To add a line number to a program using the routine below, set a string equal to the command you want entered, and then CALL 768,string\$. The routine will return control to the next line in your program, with all variables erased. For instance:

```
150 A$="20 IF AT = 15 THEN 300" :CALL 768,A$
```

...would add line 20 to your program and restart the program at the next line after 150.

This program should not work for Integer BASIC or the original RAM Applesoft for 48K IIs.

### listing 2

```
0300: 20 BE DE 20 7B DD A0 02 $196A
0308: B1 A0 99 06 00 88 10 F8 $A9C1
0310: E6 06 A9 33 85 38 A9 03 $4C6B
0318: 85 39 20 EA 03 A9 00 8D $5028
0320: 2F 03 A5 75 8D 30 03 A5 $039D
0328: 76 8D 31 03 4C 03 E0 00 $8766
0330: 00 00 00 A5 06 F0 17 8C $90E5
0338: 32 03 AC 2F 03 EE 2F 03 $8AD1
0340: B1 07 09 80 AC 32 03 C6 $2271
0348: 06 D0 02 A9 8D 60 20 89 $F286

0350: FE 20 EA 03 AD 30 03 85 $6B2D
0358: 50 AD 31 03 85 51 20 1A $8223
0360: D6 A0 00 B1 9B 85 75 18 $60EF
0368: 69 FF 85 B8 C8 B1 9B 85 $B21D
0370: 76 69 FF 85 B9 4C D2 D7 $FAD0
```

Here is a simple curve-sketching program using this technique:

### Listing 3

```
10 HGR : HCOLOR= 3 : HPL0T 140 , 0 TO 140 , 191 :
HPL0T 0 , 96 TO 279 , 96 : IF PEEK ( 768 ) +
PEEK ( 769 ) < > 222 THEN PRINT CHR$ ( 4 )
" BLOADADDLINE .OBJ
20 HOME : VTAB 21 : INPUT "ENTER*YOUR*FUNCTION*
->*" : F$ : F$ = "40" + F$ : CALL 768 , F$
30 F = 0 : FOR X = - 25 TO 25
50 Y = Y * 2 : X2 = X * 2 : IF Y < - 95.4 OR Y
> 96.4 THEN FOR A = 0 TO 3 : P = PEEK
( 49200 ) : NEXT : NEXT : GOTO 80
60 IF NOT F THEN HPL0T X2 + 140.5 , 96.5 - Y : F
= 1 : NEXT : GOTO 80
70 HPL0T TO X2 + 140.5 , 96.5 - Y : F = 1 : NEXT
80 INPUT "ANOTHER*FUNCTION*?" : F$ : IF F$ =
"N" THEN END
90 INPUT "ERASE*THE*GRAPH*?" : F$ : IF F$ =
"Y" THEN 10
100 GOTO 20
```

### Listing 3 Checksums

10	- \$2CB3	70	- \$0155
20	- \$53EA	80	- \$7250
30	- \$A382	90	- \$37F5
50	- \$6B77	100	- \$3BC0
60	- \$B665		

### Sourcecode for Listing 2

```
*
* ADD A LINE TO AN APPLESOFT PROGRAM
*
1030 .OR $300
1040 CURLIN .EQ $75 AND 76: CURRENT LINE #
1050 DSCRIPT .EQ 6 STRING DESCRIPTOR
1060 LENGTH .EQ 6 OF STRING
1070 ADDR .EQ 7 AND 8: STRING LOCATION
1080 CALLME JSR $DEBE CHECK FOR A COMMA
1090 JSR $DD7B LOCATE THE STRING
1100 LDY #2
1110 .1 LDA ($A0),Y GET STRING DESCRIPTOR (LENGTH & LOCATION)
1120 STA DSCRPT,Y
1130 DEY
1140 BPL .1
1150 INC LENGTH ADD 1 TO SIMPLIFY COUNT LATER
1160 LDA #KEYINT LOAD NEW INPUT VECTOR
1170 STA $38
1180 LDA /KEYINT
1190 STA $39
1200 JSR $3EA DOS ROUTINE TO SAVE OUR INTERCEPT TO
* BE CALLED AFTER DOS IS THRU *
1220 LDA #0
1230 STA CHRPTR START AT BEGINNING OF STRING
1240 LDA CURLIN SAVE CURRENT LINE #
1250 STA LINE
1260 LDA CURLIN+1
1270 STA LINE+1
1280 JMP $E003 WARMSTART BASIC
1290 CHRPTR .BS 1 POINTER INTO STRING TO CURRENT CHARACTER
1300 LINE .BS 2 LINE # ROUTINE IS CALLED FROM
1310 TEMP .BS 1 TEMPORARY STORAGE FOR Y
1320 KEYINT LDA LENGTH
1330 BEQ .2 BRANCH IF ENTIRE LINE HAS BEEN ENTERED
1340 STY TEMP SAVE Y
1350 LDY CHRPTR
1360 INC CHRPTR
1370 LDA (ADDR),Y GET NEXT CHARACTER FROM STRING
1380 ORA #$80 SET HIBIT TO PRETEND IT'S FROM KEYBOARD
1390 LDY TEMP RESTORE Y
1400 DEC LENGTH DONE?
1410 BNE .1
1420 LDA #$80 CR TO ENTER LINE
1430 .1 RTS
1440 .2 JSR $FE89 PUT BACK NORMAL INPUT VECTOR
1450 JSR $3EA & TELL DOS
1460 LDA LINE
1470 STA $50
1480 LDA LINE+1
1490 STA $51
1500 JSR $D61A FIND LINE IN $51,50
* LEAVES $9C,9B POINTING TO THE 1ST BYTE OF THE LINE *
1520 LDY #0
1530 LDA ($9B),Y FIND START OF NEXT LINE
1540 STA CURLIN MAKE NEXT LINE CURRENT LINE
1550 CLC
1560 ADC #$FF POINT TXTPTR TO 1 BYTE BELOW START OF NEXT LINE
1570 STA $B8 $B9, B8 POINTS TO BASIC CODE BEING INTERPRETED
1580 INY
1590 LDA ($9B),Y
1600 STA CURLIN+1
1610 ADC #$FF
1620 STA $B9
1630 JMP $D7D2 RUN REST OF PROGRAM
```

..... end of sourcecode .....





# Readers Data EXchange

for the serious user of Apple computers ( II, II+, //c, //e, IIgs, etc...)



when writing a...

## letter to RDEXed

- Remember that your letters or parts of them may be used in the new Readers Data Exchange even if you don't address it to the RDEX editor. Correspondence that gets published may be edited for clarity, grammar and space requirements.
- Because of the great number of letters we receive and the ephemeral and unpredictable appearance of our all-volunteer staff, any response to your queries will appear only in the RDEX, so it would be more appropriate for you to present technical questions to the readers and ask for their responses which will then be placed in the RDEX section.
- Address your letters to:  
**COMPUTIST**  
**RDEX Editor**  
**PO Box 110846-K**  
**Tacoma, WA 98411**
- Although COMPUTIST can no longer purchase short softkeys and articles, please continue to contribute them but place them in a letter to the editor so that they get published in the RDEX as soon as possible.

-RDEXed

## input

The INPUT column has been made a part of the RDEX (Readers Data Exchange) forum.

Opinions expressed are not necessarily those of COMPUTIST, its volunteer editorial staff or SoftKey Publishing. RDEX is a published forum for all readers where the open exchange of computer information (opinions, softkeys, copy parms and general discoveries) can occur. Such information cannot be verified by the COMPUTIST editorial staff. If errors are detected, please send corrections and updates to.....-RDEXed.

### Mike Basford

#### How to convert List Handler files into standard text files

In COMPUTIST43, James Patton requested information on converting List Handler files to standard text files. I have just completed this conversion to a data base of about 1700 records, and it is fairly straight forward.

I first used the utilities program that comes with List Handler to convert the files to DIF files on a DOS 3.3 disk. This can simply be used by most newer programs as is, but I then converted them to IBM format, using a card from Micro Solutions, called Matchpoint.

This card is fantastic for all those people like myself, who have years work of Apple files, and have now moved up to an IBM compatible system. It enables me to read and write to Apple disks, and to copy files to and from the IBM format, and DOS 3.3 or ProDOS. I strongly suggest it to all users of both systems.

#### multi-computer-fluency

Incidentally, this brings up another point. I have a Franklin that is only used for wordprocessing, an Apple II Plus that has been extensively modified for cracking (3 sets of ROM, NMIs, etc.), and a //e with Senior PROM installed. I have EPROM burners, and enough cards for three Apples, but this letter is being written on my IBM AT clone.

My first IBM clone had 640k, 8MHZ clock, 20 meg hard drive, and cost about the same as my //e!!! This one has 44 meg hard drive, 1 meg RAM, 10 MHZ clock, and with colors that would blow your mind. And the price is about the same as a Mac.

The argument about which is best reminds me of the arguing about Chevy or Ford cars. Who cares??? The more we learn about ALL computers, the wiser we will be.

#### IBM RDEX, Amiga RDEX?

Copy protection is copy protection, and when I read about a new way to defeat the curse, I learn more about computers in general. And how many of us would have the machine language knowledge that we have today if it wasn't for mags like this one???

I would like to hear how to defeat ALL copy-protection and there are thousands of IBM type users that want to also.

I suggest a small section for non-Apple cracks, and see how it goes. And please, even if you don't yet own any other type of computer, remember, things change fast in this game - don't get left behind!!!

I agree. Copy-protection is rampant. And RDEX is ready to print any softkeys for any software for any machine. So, if readers submit cracking data for other machines, please specify the system or I will assume its for the Apple.

Many of us are multi-machine fluent and own or operate a variety of computers, both Apple and non-Apple. It's time that we put away our machine biases and realize that there are many non-Apple COMPUTISTS who need softkey info that this magazine can provide.

For openers, look at the top of the page. Note the Apple... Perhaps soon, there'll be another RDEX-type column in COMPUTIST for Amiga, or IBM, or....

COMPUTISTS everywhere should unite against copy-protection, regardless of the machine (s) you personally favor (or disfavor). So let's see more softkeys for locked-up software for any and all systems.....RDEXed.

#### Dots-Perfect

One more product I would like to push (how about other happy users of things letting us all in on it?) and that is Dots-Perfect by Dresselhaus. It is a replacement for the EPROMS in an Epson printer, and it turns it into a near letter-quality printer, and is simple to install.

Hint for a possible Softkey for...

#### List Handler

Silicon Valley Systems

Well, I feel strange not contributing any cracks this time, but as a teaser, the Handler series can all be cracked by copying the disk with COPYA set to ignore errors, and then freeing up tracks \$12-\$22 in the VTOC and copying a copy card version into this area.

It is a little tricky, but anyone with Senior PROM, or the modified F8 ROM from this magazine, should be able to figure it out. It checks for some weird stuff on the early part of the disk, and this is not protected anyway.

Put a fast DOS on and have FUN!!!



## Brian A. Troha

First, In my letter that appeared in COMPUTIST47 page 43, I meant to say that the edit for *Lucifer's Realm* should be used on *Strip Poker*. c.f. means "compare", that is, compare the softkey (or use the softkey) for *Lucifer's Realm*.

My editing error. The softkey update for *Lucifer's Realm* (RDEX 47 page 34) is my erroneous title. Ignore it.

### Breaking Activision's 3 1/2" Disks

#### ■ Requirements:

- Apple Iigs
- ProDOS Block Editor with search ability

#### ■ Tools Used:

- Copy II Plus ProDOS* was used to make a copy of the 3 1/2" disk
- SAND.PRODOS* from Cutting Edge Ent. was used to search the disk and make the block edits. This program is invaluable in the procedure.

## The Protection

Activision keeps coming out with special purpose programs that are 'The Only One of It's Kind' (at least for a few months) for the Apple Iigs. They usually have the same type of protection schemes, that is: The program comes on a 3 1/2" disk and is protected by putting a 'Bad Block' over block 7. The program will check for the error and if not found, it 'knows' it's a copy. But, what if you have a hard disk and would like to upload *Paintworks*, *Draw* and/or *Writer's Choice*?

Well, if you only need a 'Key Disk' that would be fine, but Activision's protection checks the disk loaded from. So here is my quick solution.

## The Solution

You first must make a copy of the 3 1/2" disk with any whole disk copier. Now search the disk for **22 A8 00 E1 08 00**. This is a ProDOS 16 call to 'Volume', that is, use the disk being loaded from. Following right after the ProDOS call is a JSL (jump subroutine long) call to the code that actually checks for the bad block.

Iigs Softkey for...

### **Paintworks Plus v1.0**

Activision

With *Paintworks*, I found the following code on block 291 starting at 1BB (byte BB of buffer 2). Here's a run-down:

```

C2 20      REP #20      LONG ACC
22 A8 00 E1 JSL E100A8  PRODOS 16 "MLI"
08 00      0008      COMMAND #
C6 09 00   0009C6    PARM TABLE LOCAT ION
00         00        END OF MLI PARM TABLE
B0 1A      BCS TRYAGAI N CHECK FOR ERROR
AD CE 09   LDA $09CE
C9 40 06   CMP #0640
D0 12      BNE TRYAGAI N
0B         PHD
A2 DC 09   LDX #09DC
A0 00 00   LDY #0000
22 03 00 00 JSL 000003  CHECK FOR BAD BLOCK
2B         PLD
C9 06 09   CMP $0906
D0 01      BNE TRYAGAI N
60         RTS          IT'S OK START PROGRAM

```

With your block editor load block 291 and change byte 1C9 (byte C9 of buffer 2) from AD to 60 and the program will work fine. If you upload the program to hard disk you need all the files in this directory.

- Paintworks - 16 bit file system
- Painttools - Dir file with support files
- Tool027
- Tool0288
- Tool018
- Helpfile
- Palette
- Fonts - Dir of all fonts used

Iigs Softkey for...

### **Writer's Choice - Elite v1.0**

Activision

I found the code on block 523 starting at byte 51.

```

C2 20      REP #20      LONG ACC
22 A8 00 E1 JSL E100A8  PRODOS 16 "MLI"
08 00      0008      COMMAND #
59 00 00   000059    PARM TABLE LOCAT ION
00         00        END OF MLI PARM TABLE
B0 1A      BCS TRYAGAI N CHECK FOR ERROR
AD 61 00   LDA $0061
C9 40 06   CMP #0640
D0 12      BNE TRYAGAI N
0B         PHD
A2 6F 00   LDX #006F
A0 00 00   LDY #0000
22 4D 0C 00 JSL 000C4D  CHECK FOR BAD BLOCK
2B         PLD
C9 07 00   CMP $0007
D0 01      BNE TRYAGAI N
60         RTS          IT'S OK START PROGRAM

```

With your block editor, load block 523 and change byte 5F from AD to 60 and the program will work just fine. If you upload the program

to a hard disk you need all the files in this directory.

- Writer - 16 bit system file
- Writetools - Dir file with support files

Notice how close the two protection schemes are and how easy (!) it was to defeat, once you know how. With *Paintworks*, I first tried changing the check code to all NOP's (EA's), but the program will start up with the "takes time" symbol (the little watch). It will only work when changing the AD to a 60.

I have been told that a similar trick will work for *Tass Times*, *Hacker II*, and *Shanghi* (GS versions). I don't own any of those programs, so I cannot verify this, but it SHOULD work!

Iigs Softkey for...

### **Draw Plus v1.0**

Activision

For *Draw Plus* the code has been changed a little bit. I found this code on block 516 starting with byte \$20:

```

C2 20      REP #20      LONG ACC
22 A8 00 E1 JSL E100A8  PRODOS 16 "MLI"
08 00      0008      COMMAND #
E0 0C 00   000CE0    PARM TABLE LOCAT ION
00         00        END OF MLI PARM TABLE
B0 2B      BCS TRYAGAI N CHECK FOR ERROR
AD E0 0C   LDA $0CE0
C9 40 06   CMP #0640
D0 23      BNE TRYAGAI N
AD 84 00   LDA $0084
48         PHA
AD 86 00   LDA $0086
48         PHA
0B         PHD
A2 F6 0C   LDX #50CF6
A0 00 00   LDY #50000
22 08 0D 00 JSL 000D08  ACTUAL JSL OF CHECK
2B         PLD
FA         PLX
8E 86 00   STX $0086
FA         PLX
8E 84 00   STX $0084
C9 08 00   CMP #50008
D0 02      BNE TRYAGAI N
AB         PLB
60         RTS          IT'S OK START PROGRAM

```

With your block editor, load block 516 and change, starting with AD E8 0C up to and including D0 02, all to EA's and the program will work just fine. You should see the B0 2B EA ... EA AB 60. That is, you will have to change 39 bytes to EA's. This time, I first tried changing the AD E8 to AB 60 and had some problems (it could have been *Draw* looking for the driver routines and not finding them) but one method should work. If you upload the



program to hard disk you need all the files in this directory:

- Draw.Plus - 16 bit system file
- Drawtools - Dir file with support files

On the *Music Studio*, the ProDOS 16 call is on block 57A, but the protection has been changed to the point where I cannot defeat it (yet). If you can successfully copy *Music Studio*, please let me know how YOU did it.

As for the level of ProDOS you can use, I will say this: for both *Draw* and *Writer's Choice*, you should use level 1.1 or 1.2 of ProDOS 16 because you need the "drivers" for the Imagewriter and the Laserwriter.

*Paintworks v1.0* (the upgrade may be different), on the other hand, has it's own routines and a few modified tools. These tools may cause problems when quitting to another program, as they "cannot find" said tools (font manager). I hope Activision will correct this problem in it's upgrade.

## Second Letter from B. A. Troha

Softkey for...

### Paintworks Plus v1.1

Activision

When *Paintworks Plus* was updated to version 1.1 the protection routine was moved and changed a bit. Version 1.1 fixes some bugs in version 1.0 and also supports "Apple's preferred" paint file format, so you can now use clip art disks for other paint programs.

The steps to follow to make a copy are the same except that I found the protection code in block 48A at byte \$D0. Use your block editor to load block 48A and change byte \$DE from \$AD to \$60. Write the block to disk.

If you upload the program to a harddisk you need:

- Paintworks - 16 bit system file
- Painttools - Dir file with support files
- Paint027
- Paint028
- Paint018
- Helpfile
- Palette
- Screen

/Fonts - Dir of all fonts used

/Drivers - Dir of all drivers used

All the files in both of these directories.

There is also a file called "CLIPBOARD" in the system subdirectory that is needed. As for level of ProDOS, you should use 1.1 or higher of ProDOS 16. However, you should make sure that in the TOOLS subdirectory the file "TOOL19" is not present. Should this file be present the program will respond with "CAN NOT FIND PICTURE MANAGER. 0008" or something like that.

## And Still another letter...

Softkey for...

### Arctic Fox

Electronic Arts

#### Requirements:

- Apple II series with 64K
- A blank disk
- A sector editor
- A disk search utility
- Fast copier (copy II plus or equiv.)

*Arctic Fox* is a modified and re-vamped *Stellar 7* (by the same author Damon style). The method used to protect *Arctic Fox* (and most EA programs) is track imaging, that is, track \$5.5 is a copy of \$5. This is the reason you get a read error on track \$6. You will need a fast copy program that continues on errors. Set your disk search utility for the following four patterns (common to most EA programs and protection schemes)

	will be	or will be
1. 4C 69	4C 69 A0	4C 69 05
2. 20 F8	20 F8 A0	20 F8 05
3. 20 FE	20 FE A0	20 FE 05
4. 20 03	20 02 A0	20 03 05

All occurrences found on track \$01 should be checked out. On my disk there are eight combinations of the above bytes, only six of which must be changed. Each change should be made as follows: 18 60 XX, where XX will set the check sum. The following set of changes should work:

from:	to:
4C 69 A0	18 60 DD
4C 69 05	18 60 42
20 F8 A0	18 60 40
20 F8 05	18 60 A5
20 FE A0	18 60 46
20 FE 05	18 60 AB
20 03 A0	18 60 4B
20 03 05	18 60 B0

There is a second line of protection and that is a memory wipe routine at \$6000. So you should also look for 4C 00 60 and 20 00 60, I found five such patterns. The following is a list of edits I made on a copy:

Track	Sector	Byte(s)	To
\$01	\$0B	\$47	18 60 A5
		\$4C	18 60 B0
\$01	\$0E	\$47	18 60 40
		\$4C	18 60 4B
\$01	\$0F	\$00	18 60 DD
		\$6F	18 60 DD

\$17	\$01	\$BA	EA EA EA
		\$D8	seven EAs
\$18	\$06	\$7C	ten EAs
\$19	\$01	\$80	ten EAs
\$19	\$0B	\$CB	seven EAs
		\$DF	EA EA EA
\$1D	\$09	\$B0	A9 52 85 11 eight EAs

That should do it!

Softkey for...

### EDD 4

Utilico Software

#### Requirements:

- Apple II series
- EDD version 4
- 48K slave disk

*Essential Data Duplicator* has been upgraded to version 4. New features like the hi-res disk scan and examine disk drive are now included. Like the three version before a bitcopy seems hopeless. This is due to the very precise synchronization routines used when the program was recorded. Only the tracks from \$00 to \$0D are used and are written using the 4 by 4 nibble encoding scheme. However there is one thing you should never forget: There are NO unbreakable protection schemes. This is true for at least one reason, the program must somehow load into the machine. So if you can follow the boot of the program you could (eventually) unprotect any program. Being able to "Boot code Trace" a program is a valuable skill. I say skill because, the loading of each program will change and the loader will try every trick in the book to hide what it is doing.

The theory behind boot code tracing relies on two things. First, there is a program in ROM on the disk controller card that will start the loading of all programs. This program will load in sector \$00 of track \$00. If the number located at \$800 is one then control will be passed to \$801. If \$800 has a number larger than one the rom code will load in more sectors. Electronic Arts is famous for loading in five to sixteen sectors to start the boot. Second, if you where to move this program down to RAM you could change it to do anything you would like. The following will give a basic example of this technique.

To begin, move the boot ROM down so we can modify it to jump to the monitor.

8600<C600.C6FFM  
86F8:4C 59 FF

Insert the original *EDD 4*, and start the boot.

8600G

Listing through the \$800 page to the end I looked for jumps to the next stage of the loader.





At \$8AD you will find a JMP \$400, this is the start of what is known as BOOT1 or the first boot stage. This is a jump into the text page, so we will have to move it up to examine it.

To do this three things must be done. One, we must change the code to maintain control and move the text page memory up. Two, you will also have to change the loading page of the zero sector, so the modified boot will not be overwritten. Third and last, you must redirect the boot zero code at \$8600 from the monitor to the modified first stage code.

*Jump to the move routine.*

8AD:4C 00 0F JMP \$0F00

*Load sector zero into \$8000*

8659:80

*Continue the boot*

86F8:4C 01 08 JMP \$0801

The memory move:

*Four pages to move*

F00:A2 04 LDX #\$04

*Start at zero*

F02:A0 00 LDY #\$00

*Load a byte from the text page*

F04:B9 00 04 LDA \$0400,Y

*Store it in a safe place*

F07:99 00 14 STA \$1400,Y

*Increment the pointer*

F0A:C8 INY

*Move a full page*

F0B:D0 F7 BNE \$0F04

*Increment the text page*

F0D:EE 06 0F INC \$0F06

*Increment the store location*

F10:EE 09 0F INC \$0F09

*Out of pages to move*

F13:CA DEX

*No, then move more*

F14:D0 EE BNE \$0F04

*Yes, jump to the monitor*

F16:4C 59 FF JMP \$FF59

Now list through the moved code and you will find a JuMP to \$C00 (at \$1477, which should be \$477). This is the real start of the program, and the point at which it must be stopped.

Well, you have a computer, so let it do the work for you. This means writing a "Tape-worm", or a program to load in each stage, changing the jumps out to maintain control. When the program has loaded in, return control to the user. This is not as hard as it sounds for this boot process.

The ROM loads in code at \$800, the jump to \$801 (the first jump). The new code at \$801 loads more code into the text page then jumps to it from \$8AD (the second jump).

The last piece of code loads in the rest of the program then jumps to the start from \$477 (the third and last jump). There are now three jumps we need to manipulate. To start the worm,

move the ROM boot code to \$8600 and add the code that follows:

8600<C600.C6FFM

*Overwrites the JMP \$0801*

86F8:A9 05

86FA:8D AE 08

86FD:A9 87

*Change the JMP \$0400 to JMP \$8705*

86FF:8D AF 08

*Now jump to the first stage loader*

8702:4C 01 08

8705:A9 59

8707:8D 78 04

870A:A9 FF

*Change the JMP \$0C00 to the monitor*

870C:8D 79 04

*Jump to the second stage loader*

870F:4C 00 04

An 8600G will start the worm, which loads all of EDD 4 into memory and leaves us in the monitor. EDD 4 uses the memory from \$C00 through \$5FFF (this includes the hi-res title page). Part of the memory range from \$B000 through \$BFFF is used by the program for disk access. These parts must be moved down and later replaced. Lastly the whole thing can be saved out of memory. The \$1C00 page was empty so I placed the start up routine there.

## Step by step

**1** Enter the monitor to make the tape-worm.

CALL -151

**2** Move the boot ROM down and enter the code to finish the worm.

8600<C600.C6FFM

86F8:A9 05 8D AE 08 A9 87 8D

8700:AF 08 4C 01 08 A9 59 8D

8708:78 04 A9 FF 8D 79 04 4C

8710:00 04

**3** Insert original and run the tape-worm to load EDD 4.

8600G

**4** Move the used portions of high memory down.

6000<B000.B3FFM

6400<B700.BFFFM

**5** Boot a slave disk with a short hello program.

C600G

**6** Enter the monitor again.

CALL -151

**7** Add the start up code and memory

moves:

1C00:A2 04 A0 00 B9 00 60 99

1C08:00 B0 C8 D0 F7 EE 06 1C

1C10:EE 09 1C CA D0 EE A2 09

1C18:A0 00 B9 00 64 99 00 B7

1C20:C8 D0 F7 EE 1C 1C EE 1F

1C28:1C CA D0 EE A9 00 8D F2

1C30:03 A9 C6 8D F3 03 49 A5

1C38:8D F4 03 AD 57 C0 AD 55

1C40:C0 AD 52 C0 AD 50 C0 2C

1C48:10 C0 AD 00 C0 10 FB 2C

1C50:10 C0 60

**8** Fix the start of the program so it will run:

BFD:20 00 1C

**9** Save the whole program (at last!)

BSAVE ESSENTIAL DATA DUPLICATOR 4,  
ASBFD, L\$60F8

## For versions higher than EDD v4.3

When EDD 4 was upgraded to support the new Apple IIgs (version 4.4 or later) the loader and program was changed. Some of the absolute addresses given have changed and you need to save four more pages of memory.

If you follow along with the article to verify the right locations you can crack any new version that comes out. The changes are as follows:

1. The JMP \$400 is now at \$8AF instead of \$8AD

2. The JMP \$C00 is now at \$47E instead of \$477

3. There is a short routine from \$8A00 - \$8DFF that is needed for the character table used by the hi-res disk scan. It moves itself down to \$800-\$BFF, and is called near the start of the program.

Here are the needed modifications to the step-by-step method:

**2** Move the boot ROM down and enter the code to finish the worm.

8600<C600.C6FFM

86F8:A9 05 8D B0 08 A9 87 8D

8700:B1 08 4C 01 08 A9 59 8D

8708:7F 04 A9 FF 8D 80 04 4C

8710:00 04

**4** Move the used portions high memory down.

6000<B000.B3FFM

6400<B700.BFFFM

6D00<8A00.8DFFM



**7** Add the start up code and memory moves:

```

1C00:A2 04 A0 00 B9 00 60 99
1C08:00 B0 C8 D0 F7 EE 06 1C
1C10:EE 09 1C CA D0 EE A2 09
1C18:A0 00 B9 00 64 99 00 B7
1C20:C8 D0 F7 EE 1C 1C EE 1F
1C28:1C CA D0 EE A2 04 A0 00
1C30:B9 00 6D 99 00 8A C8 D0
1C38:F7 EE 32 1C EE 35 1C CA
1C40:D0 EE A9 00 8D F2 03 A9
1C48:C6 8D F3 03 49 A5 8D F4
1C50:03 AD 57 C0 AD 55 C0 AD
1C58:52 C0 AD 50 C0 2C 10 C0
1C60:AD 00 C0 10 FB 2C 10 C0
1C68:60

```

**9** Save the whole program.

```

BSAVE ESSENTIAL DATA DUPLICATOR 4,
ASBFD, L$64F8

```

Softkey for...

## Destroyer

Epyx

### Requirements:

- 128K Apple II (by the program)
- COPYA
- sector editor

*Destroyer* by Epyx is another battle ship simulation that's in double hi-res, which makes for a great looking game.

The protection is as follows: First the epilogue bytes have been changed from DE AA to FF FF. Second the data check sums have been altered. Third the code at \$B700 is encrypted and will run only after a disk check passes and decodes that memory page. Lastly there is a **bit insertion** check on the disk at \$BB00, when this check passes it decodes \$B700-\$B7FF and jumps to it. Now that you know what's going on, here's how to get things, like they should be, unprotected!

**1** Boot DOS 3.3.

PR# 6

**2** Enter the monitor, make some changes and run COPYA to copy the disk.

```

CALL-151
B942:18
B925:18 60
B988:18 60
RUN COPYA

```

**3** Use your sector editor to make some changes.

Trk	Sctr	Byte(s)	From	To
\$00	\$0A	\$00-\$17	A9 00 A2 F0	AD FF B7 8D
			9A 95 00 E8	09 BB A0 00
			D0 FB A9 0A	A9 00 59 00
			85 FC A6 2E	B7 99 00 B7
			BD 89 C0 BD	88 D0 F5 A6
			8E C0 A9 80	2B 4C 00 B7

This edit gets the value to use in the decoding routine, and decodes the memory page (\$B700-\$B7FF). Then it loads the X-reg. with the "boot slot" times \$10 and continues the boot by jumping to \$B700. This "over-rides" the disk check and acts like the check actually passed. The game will boot and run just fine.

An alternate method would be to load sector \$B of track \$0 into memory. Run a similar type code (to the sector edit) and overwrite sector \$B with the decoded version. Then you will have to load sector \$0, track \$0 and change the 4C 00 BB to 6C FD 08.

Softkey for...

## Hard Ball

Accolade

### Requirements:

- DOS 3.3 & COPYA
- sector editor
- blank disk

To softkey this game is very easy. The protection routine is at \$B260 and is called from sector zero of track zero.

### Step by step

**1** Boot DOS 3.3

**2** Type:

```

CALL-151
B942:18

```

**3** Run COPYA. An error on track \$22 is OK.

RUN COPYA

**4** Change:

Trk	Sctr	Byte(s)	From	To
\$00	\$00	\$4C-\$4F	20 60 B2 F0	EA EA 18 90

The edit skips the JSR, clears the carry bit, and changes the branch equal to branch carry clear. This forces the branch to take place (jumping over the reboot code) and fools the protection routine. Now the game goes on its merry way thinking it's an original.

Softkey for...

## Super Bunny

Datamost

### Requirements:

- Super Bunny
- Super IOB 1.5

*Super Bunny* is a fun little game by Datamost reviewed in COMPUTIST19. There are six levels to master and the ten top scores are saved to disk. As Regonald Rabbit you jump across from one elevator to another, trying to reach the far side while avoiding the other animals. When you have gotten to the other side you eat the magic carrots that transform you into super bunny. Now you must go back and destroy the evil animals by landing on them.

This game is a little older and is very easy to softkey after some hints. The protection (great for 1983) involves synchronization and modified marks. These marks can be found by using a nibble editor, one that shows sync bytes in inverse is preferred. The Address markers are changed to "BE XX E7" for prologue and to "CD D3 XX" for epilogue bytes, where XX changes from sector to sector. The Data markers are changed to "EB AC F6" for prologue and to "EA 9A DC" for epilogue bytes.

This type of protection falls easily and quickly to Super IOB with a swap controller. To get the RWTS image of *Super Bunny* requires a little boot code tracing, or a way to reset into the monitor. The method I will describe can be used by anyone, with or without an old F8 ROM. To begin with, move the boot ROM down to \$8600, type:

```
8600<C600.C6FFM
```

Modify it to stop after loading the RWTS instead of jumping to \$B700 by typing:

```
86F8:A9 59 8D 5B 08 A9 FF 8D 5C 08 4C 01 08
```

Then start everything up with \$8600G. When the computer beeps and control returns, move the RWTS down and boot a slave disk. Install the controller and run Super IOB to complete the softkey.

### Step by step

**1** Boot DOS 3.3 and initialize a slave disk for *Super Bunny*.

```
PR#6
INIT BOOT
```

**2** Make and run a loader to load in the RWTS routine.

```
CALL -151
```



8600<C600.C6FFM  
86F8:A9 59 8D 5B 08 A9 FF 8D  
8700:5C 08 4C 01 08  
8600G

**3** Move the RWTS down to safe memory and boot a slave disk.

1900<B800.BFFFM  
C600G

**4** Save the RWTS to your Super IOB disk.

BSAVE RWTS.BUNNY, A\$1900, L\$800

**5** Install the controller and run Super IOB.

**6** Optional, copy a fast DOS and rename BOOT to HELLO.

## Controller

```
1000 REM SUPER BUNNY
1010 TK = 3 : LT = 34 : ST = 15 : LS = 15 : CD = WR
      : FAST = 1
1020 GOSUB 390 : GOSUB 490 : GOSUB 610
1030 GOSUB 390 : GOSUB 490 : GOSUB 610 : IF
      PEEK (TRK) = LT THEN 1050
1040 TK = PEEK (TRK) : ST = PEEK (SCT) : GOTO
      1020
1050 HOME : PRINT "DONE!" "DOS" "NOT" "COPIED" :
      END
10010 PRINT CHR$ (4) ; "BLOAD" "RWTS.BUNNY,"
      A$1900"
```

## Controller Checksums

1000 - \$356B	1040 - \$1847
1010 - \$0614	1050 - \$29EC
1020 - \$F874	10010 - \$F189
1030 - \$D9CD	

Softkey for...

## Echo 1.0

Street Electronics

### ■ Requirements:

- Echo 1.0
- Old style F8 ROM or another way into monitor

Echo 1.0 was an early and very powerful bit copier that came out in 1982. The program has now evolved into the much talked about Echo Plus copy program. The protection used on Echo 1.0 is also very good for it's time and now. A bit copy seems close to impossible even using the six different bit copiers that are available. Boot code tracing is complicated by

the use of the zero page, stack, and keyboard buffer by the loader. After several attempts always ending in "NICE TRY", I tried the old "Search the entire memory for program code." Using the Know-drive to enter the monitor I figured out what parts of memory are used. The result is a fifty-one sector binary program that can be BRUN any time.

Also included are two patches to the program. The first patch allows you to exit Echo with out powering down. The read key function was intercepted and changed, so now if a **CB** is encountered a cold boot is forced through SFA62 (autostart F8 ROM only). The second patch deals with the way a write-protected target disk are handled. Instead of returning to the menu when a write-protected disk is found, Echo would continue trying to copy the disks. Echo will now do what you would expect: Turn off the disk drive, print "WRITE PROTECTED", ring the bell, and then return to the menu. To convert this program to a normal binary file use the following steps:

**1** Boot Echo 1.0 up to the hi-res title page.

**2** Enter the monitor using your favorite method.

**3** Move volatile memory up and boot a slave disk.

4000<800.9FFM  
C600G

**4** Reenter the monitor and restore memory.

CALL -151  
800<4000.41FFM

**5** Pack high memory into an unused space.

1900<3900.3FFFM

**6** Install the code for the patches and the memory move.

```
C25:4C 15 18
1244:4C 26 18
1815:20 0C FD
1818:C9 82 F0 01 60 20 58 FC
1820:EE F4 03 4C 62 FA BD 88
1828:C0 A0 00 A9 A0 99 51 06
1830:99 51 07 B9 44 18 99 D1
1838:06 C8 C0 12 D0 ED 20 3A
1840:FF 4C F1 0A A0 D7 D2 C9
1848:D4 C5 A0 A0 D0 D2 CF D4
1850:C5 C3 D4 C5 C4 A0 A0 00
1858:B9 00 19 99 00 39 C8 D0
1860:F7 EE 5A 18 EE 5D 18 AE
1868:5D 18 E0 40 D0 EA 4C 58
1870:0A
```

**7** Add a jump to the memory move routine.

7FD:4C 56 18

**8** Save the file to your disk.

BSAVE ECHO 1.0, A\$7FD, L\$3100

Update of the Softkey for...

## Thief

COMPUTIST33, page 24

If you followed the softkey for Thief in COMPUTIST33, you have a fully operational unprotected copy of Thief. However, for simplicity's sake, I went one step further by making a single file out of the game. To make Thief into a single BRUN-able file enter the following:

```
BLOAD ROBOT, A$2000
BLOAD MZ.OBJ0, A$4000
BLOAD TBGEN, A$4C00
BLOAD MSSL.OBJ0, A$5300
BLOAD MZS.OBJ0, A$5C00
BLOAD DROUTS, A$6000
CALL -151
```

```
5B00:2C 50 C0 2C 52 C0 2C 54
5B08:C0 2C 57 C0 2C 10 C0 A2
5B10:FF A9 60 20 A8 FC AD 00
5B18:C0 30 03 CA D0 F3 2C 10
5B20:C0 A2 06 A0 00 B9 00 4C
5B28:99 00 08 C8 D0 F7 EE 27
5B30:5B EE 2A 5B CA D0 EE 20
5B38:00 08 A2 08 A0 00 B9 00
5B40:53 99 00 08 C8 D0 F7 EE
5B48:40 5B EE 43 5B CA F0 EE
5B50:4C 00 40
1FFD:4C 00 5B
```

BSAVE THIEF, A\$1FFD, L\$4DE0

This new file will show the hi-res title page for a couple of seconds (or less if you press a key) then the TBGEN routine is run and MSSL.OBJ0 is moved to it's correct location and the game is run. The fruit of your labors is a single seventy-nine sector BRUN-able version of Thief.

Softkey for...

## The Dam Busters

Accolade

### ■ Requirements:

- fast disk copier or COPYA
- sector editor

The Dam Busters is game in which you pilot



a WWII bomber and try to get behind the German lines to destroy dams.

The protection used is a modified track \$22 and then a check for this track. The disk contains two programs, first a HELLO program that simply runs SPOCKS BRAIN. Inside the binary program you will find the disk check. It looks like this:

```
JSR $2030 - Check track $22
BEQ $202D - OK; continue loading
JMP $C600 - Fail; reboot disk
```

While overwriting the call to the protection I added a DEC \$03F4. So now when you press reset the program will reboot instead of bombing into the monitor.

### Step by step

**1** Run COPYA. Ignore error on track \$22.

**2** Use your sector editor.

Trk	Sctr	Byte(s)	From	To
\$13	\$B	\$29-\$30	20 30 20 F0 03 4C 00 C6	CE F4 03 EA EA EA EA EA

If that doesn't work you should try:

```
BLOAD SPOCKS BRAIN
2025:CE F4 03 EA EA EA EA
BSAVE SPOCKS BRAIN, AS$2000, L$5B1
```

## Greg Poulos

Softkey for...

**Animate**  
Broderbund

Animate from Broderbund, is an excellent window-based animation program similar to Fantavision, but in double hi-res graphics and with more options.

Before doing anything with it, though, I decided to deprotect it, since in the manual, the folks at Broderbund charge you \$15 for a backup, which I don't want to pay if my disk freaks out. Anyway, here's how I did it.

### The Protection

There are three disks to Animate, but only the boot disk is protected. I first tried copying it with Copy II Plus Fast Copy and it copied without any errors. I booted up my copy and sure enough, an 'R' was printed in the upper left hand corner of the screen and the disk

rebooted. I then tried searching for any instructions that branched to the disk booting routine at SC6000. I didn't find any, so I booted the disk again, this time pressing the button on my NMI card to break into the monitor.

Once in the monitor, I started to look through the boot code at \$800, and after using the monitor 'L' command several times, I found the first JMP in the boot code. So I listed the memory where the JMP went to and after fooling around with the code for awhile, I found that there was a routine that printed an 'R' and did something else. This was at \$819E. So I scanned the disk for any JMP's or JSR's to \$819E and found several. After NOPing these out, I booted the disk again. This time it loaded in more tracks and booted further, but once again, it performed another nibble count and rebooted. I went back to searching through memory until I found stuff at \$A42 and \$A89. So, I found a JMP to \$A42 and a JSR to \$A89, which I NOPed out. Upon booting, the program loaded until it prompted for side two to be inserted. So, I put in side two and pressed a key. The bar on the double hi-res screen appeared with all the options, but the disk rebooted again!

I found a routine that did this and put a 60 (RTS) on the first byte of it. Then the program would load in completely, and some options could be used. But some options COULDN'T be used, because before loading in the option, the program would do another nibble count and reboot. I noticed that if [RESET] is pressed, it reboots in the very same manner as when the nibble count is performed. So I searched the disk (side two) for 6C F2 03, an indirect JMP to the reset vector, and found five occurrences. The first two, I just NOPed out, but on the third, I first changed a BEQ instruction to a BNE instruction. Therefore, when the nibble count can't find the proper bytes, it will BNE (Branch if Not Equal) to an RTS. The last two JMP's to \$3F2 I just NOPed out, and Animate worked fine.

### Step by Step

**1** Copy all six sides of Animate with any fast copy program. If you get errors on the boot disk then do the following, otherwise skip to step 2.

```
LOAD COPYA
CALL -151
B925:18 60
B988:18 60
BE48:18
B8FE:00
3D0G
RUN
```

**2** Perform these sector edits to SIDE ONE of the Animate boot disk.

Track	Sector	Byte(s)	From	To
\$00	\$01	\$03	4C 9E 81	EA EA EA
		\$90	4C 9E 81	EA EA EA
\$07	\$00	\$DA	4C 42 0A	EA EA EA
		\$62	20 89 0A	EA EA EA

**3** Perform these sector edits to SIDE TWO of the Animate boot disk.

Track	Sector	Byte(s)	From	To
\$01	\$01	\$D8	A6	60
\$02	\$00	\$97	6C F2 03	EA EA EA
\$02	\$0A	\$47	6C F2 03	EA EA EA
\$03	\$03	\$16	F0	D0
\$08	\$0F	\$37	6C F2 03	EA EA EA
\$0F	\$08	\$43	6C F2 03	EA EA EA

That's it!

Softkey for...

**Up-n-Down**  
Sega

Here's how to crack Up-n-Down from Sega. The protection is simple data mark changes. Once you have it in normal format, the disk gets read errors. To fix this, we must change the old data marks (AA AA) to normal (DE AA).

### Step-by-step

**1** Load COPYA from your System Master and make some changes so it ignores errors.

```
LOAD COPYA
CALL -151
B925:18 60
B988:18 60
BE48:18
B8FE:00
3D0G
RUN
```

**2** Copy your original Up-n-Down.

**3** Use a sector editor and make some changes.

Track	Sector	Byte(s)	From	To
\$03	\$03	\$9B	AA	DE
\$03	\$03	\$E9	AA	DE

That's it! If you want to change the title page, it's on track 1, sector 15 and is read with tracks positive and sectors negative.

Does anyone have any hints for Spellbreaker?



## G. Poulos' second letter

Softkey for...

### PHM Pegasus

Electronic Arts/Lucas Film

Here's a softkey for *PHM Pegasus* from Electronic Arts/Lucas film. The protection is very simple: one JMP to the protection code found at the usual spot on Track \$01, Sector \$0F.

**1** Copy both sides with any copier that will ignore the error on track \$6.

**2** Make one sector edit.

Track	Sector	Byte(s)	From	To
\$01	\$0F	\$00	4C 69 A0	18 60 DD

**3** Perform the above edit to both sides of your disk.  
That's it!

It's not that important, but EA forgot to put the bytes the nibble count searches for (on track \$6) on side 2! If you boot your original from side 2, it performs the protection and then crashes! But if you perform the above sector edit, it'll load in a message instructing you to turn the disk over.

Some Hints ...

### Spellbreaker

Infocom

Here are some hints for *Spellbreaker*, a very difficult "Expert-Level" Infocom game:

You can only GIRGOL once; keep this in mind around rockslides.

That old hermit won't let you take his white cube because it's holding together his imPERFECT home.

Idols yawn when they're tired.

Seems like the Ogre has an allergic reaction to some sort of weed.

Magic Burins sure come in handy to distinguish between white cubes.

## Michael David

In the previous issue (COMPUTIST48 we

printed an incomplete version (my error) of Michael David's RDEX contribution. We now present the complete version in this issue. . . . . RDEXed

## More Softkeys for MECC software

MECC has long been a respected name in educational software. However, MECC appears to have little regard for the teacher's dilemma of not enough money to buy back-up software and the inability to copy MECC software quickly and easily.

Softkeys for...

### Addition Logician

### Writing A Character Sketch

### Writing A Narrative MECC

I was recently given three MECC programs that could not easily be copied. They had 1983-1984 title pages, and appeared worthwhile enough to attempt to crack them. I prepared for a long, arduous cracking session on the following: *Addition Logician*, *Writing a Character Sketch* and *Writing a Narrative*.

This was not to be the case. In fact, the use of a nonmaskable interrupt and RWTS saving was not necessary. Using Super IOB and the lines listed below, the programs were cracked in less than 2 minutes.

## Step-by-step

**1** Load Super IOB v1.5 into memory.

**2** Delete lines 1000 through 9999:

```
DEL 1000,9999
```

**3** Enter the new controller:

## Controller

```

1000 REM MECC CONTROLLER WITHOUT RWTS
1010 TK = 3 : LT = 35 : CD = WR : MB = 151 : ONERR
      GOTO 550
1020 ST = 0 : T1 = TK : GOSUB 490 : RESTORE :
      GOSUB 190 : GOSUB 210 : GOSUB 170
1030 GOSUB 430 : GOSUB 100 : ST = ST + 1 : IF ST
      < 16 THEN 1030
1040 IF BF THEN 1060
1050 ST = 0 : TK = TK + 1 : IF TK < LT THEN 1030
1060 GOSUB 230 : TK = T1 : ST = 0 : GOSUB 490
1070 GOSUB 430 : GOSUB 100 : ST = ST + 1 : IF ST
      < 16 THEN 1070

```

```

1080 ST = 0 : TK = TK + 1 : IF BF = 0 AND TK < LT THEN
      1070
1090 IF TK < LT THEN 1020
1100 HOME : AS = "ALL'DONE" : GOSUB 450 : END
5000 DATA ,170 ,213 ,150 ,213 ,170 ,173 ,222 ,170
      ,222 ,170

```

## Controller Checksums

1000 - \$356B	1060 - \$6AE6
1010 - \$5E3F	1070 - \$22FD
1020 - \$B92C	1080 - \$54D8
1030 - \$E2AA	1090 - \$7FC2
1040 - \$2463	1100 - \$F952
1050 - \$E2BC	5000 - \$ECD8

**4** Copy your favorite DOS onto the disk and boot it.

Softkeys for...

### Quotient Quest

### Counting Critters

### Math Critters MECC

Although this method worked well on the three programs as listed, I was still faced with the task of cracking additional MECC wares with copyright dates varying from 1985-1986: *Quotient Quest*, *Counting Critters* and *Math Critters*.

## Step-by-step

**1** Boot up the original program and wait until the title page stops.

**2** Using a non-maskable interrupt such as Wildcard, stop the program and get into the monitor.

**3** Type:

```
1900<B800.BFFFFM
```

**4** Boot up a disk with basic on it.

```
6(=)P
```

**5** Save the RWTS just captured.

```
BSAVE RWTS.MECC program name, A$1900,
L$800
```

The above steps need to be done for each individual program you wish to crack as each will have a RWTS file of its own.

**6** Run Advanced DeMuffin Plus,



loading the correct RWTS file for each disk to be copied.

**7** Change the default parameters to use a 16-sector disk and copy from Track \$3, Sector \$00 to Track \$22, sector \$0F with an increment of 1. Load the RWTS at \$B8.

**8** Copy the disks and add the DOS of your choice.

**9** Try this crack on other educational wares. I found it worked just as nicely on Nystrom Software's *Super Sleuth*.

## Robert M. Kesslick

Update on Dean's Softkey for...

### Swordthrust series

COMPUTIST42 page 12

I would like to add to James Dean's softkey for the *Swordthrust* series. The softkey works fine. But, by not using the sector modifications in step nine for all games in the series you will lose the ability to save (suspend) and restore play.

So perform step nine on all disks. Although Mr. Dean may have only three games, the series has at least five that I know of.

While snooping around on my newly deprotected disk, I noticed Track \$11 was blank. I did some snooping around and found a standard catalog on Track \$22, so I copied Track \$22 to Track \$11, lo and behold, I could now catalog the disk and load the files.

APT for...

### the Swordthrust series

For those of you who lost or broke the Fabled Sword Excaliber and found you were unable to go back to the "King's Testing Grounds", load in the file "MAIN" and rewrite line 1270, omitting the "GOTO 2000" at the end of the line. Don't forget to write the file back to disk.

Also in the file "NEW CHARACTERS" lines 1010 to 1050 store all of the character's information. By modifying these variables I was able to develop a superman.

Keep the good stuff coming, I look forward to your next issue.

P.S. I just received COMPUTIST45 and I have to complain about the soft cover. The pages of COMPUTIST are very dear to me. And the hard cover is a lot more protection for

them. Have a heart and give us back the hard cover.

**I** The hardcover will return as soon as we can afford it. In the meantime, would you accept 16 more pages on the inside instead of the expensive hard cover?.....RDEXed

## Ralph L. Jones

### How To Make GRAPHIC.GRABBER V3 Run On the IIGS

One of the most liked programs that you have written is the GRAPHIC.GRABBER V3. Since it will not run on my IIGs because of the "Lam" routine in lines 380, 290, and 300, I have searched for a commercial program which will do the same things: two graphic buffers, six sizes of the capturing "cursor", the ability to capture a graphic from the hi-res screen, and the ability to load or save a hi-res screen, or load or save a *Print Shop* graphic to disk. A very useful feature is the ability to capture or load a graphic then expand the "cursor", then load it back on the hi-res screen in a different size "cursor". It works both ways, expand or shrink a graphic.

I finally gave up and decided to try and patch the GRAPHIC.GRABBER to run on the IIGs. My first effort was to write an Applesoft routine to move the memory that is accomplished in the routine at line 380 and lines 290 and 300.

The routine was successful, but slow!! In fact it was so slow that I considered it not a permanent solution. To use the Applesoft routine, just delete lines 290 and 300, and replace line 380 with my routine. You will see why I was dissatisfied with the speed of the Applesoft routine.

```
380 J = 0 : FOR I = 21760 TO 23104 : POKE 20224 +
      J, PEEK (I) : J = J + 1 : NEXT : J = 0 : CALL
      16499 : FOR I = 20224 TO 21560 : POKE
      21760 + J, PEEK (I) : J = J + 1 : NEXT :
      RETURN
```

This is when I started looking for a machine language routine, and I am definitely not a machine language programmer. The routine I am offering is crude and no self respecting programmer would claim it, but it works! I am hoping that someone will come up with a simple memory move routine that will work on the IIGs.

I started out by attaching my routine to the end of the OBJ.GRAPHIC GRABBER but

having a bug in my routine, it would not work. In desperation I moved it to \$300 as a separate program, and then I discovered and removed the "bug". I then decided that since it worked now at \$300, I would not "fix" the program any more by moving it to the end of the OBJ.GRAPHIC GRABBER.

To use my machine language routine, delete lines 290 and 300, and replace lines 100 and 380.

```
100 PRINT CHR$(4) "BLOAD MOVE"
380 CALL 768 : CALL 16499 : CALL 838 : RETURN
```

The machine language program was entered with the IIGs' Mini-assembler. I hope this will not only help those who need this program, but will cause some machine language programmer to supply a simpler routine that can be attached to the end of the OBJ.GRAPHIC GRABBER.

Enter the program by a:

CALL -151

...at the \* prompt type an exclamation mark:  
!

Now you are in the Mini-assembler. Type:

300: LDY #00

For the rest of the lines just type a space then the MNEMONIC a space and the OPERAND as it appears in the listing and, of course, press **RETURN**.

If you make a mistake and want to re-do a line, type the address, a colon, the mnemonic, a space, and the operand. When finished hit **RETURN** a couple of times and you will be back in the Monitor.

Then type **MC RETURN** to get back to Applesoft and save the program.

BSAVE MOVE,AS300,LS8C

## MOVE

```
300: A0 00 LDY #00
302: B9 00 55 LDA 5500,Y
305: 99 00 4F STA 4F00,Y
308: C8 INY
309: D0 F7 BNE 0302
30B: A0 00 LDY #00
30D: B9 00 56 LDA 5600,Y
310: 99 00 50 STA 5000,Y
313: C8 INY
314: D0 F7 BNE 030D
316: A0 00 LDY #00
318: B9 00 57 LDA 5700,Y
31B: 99 00 51 STA 5100,Y
31E: C8 INY
31F: D0 F7 BNE 0318
321: A0 00 LDY #00
323: B9 00 58 LDA 5800,Y
326: 99 00 52 STA 5200,Y
329: C8 INY
32A: D0 F7 BNE 032A
32C: A0 00 LDY #00
```



```

32E: B9 00 59 LDA #5900,Y
331: 99 00 53 STA 5300,Y
334: C8      INY
335: D0 F7   BNE 032E
337: A0 00   LDY #00
339: A2 C0   LDX #C0
33B: B9 00 5A LDA 5A00,Y
33E: 99 00 54 STA 5400,Y
341: C8      INY
342: E8      INX
343: D0 F6   BNE 033B
345: 60      RTS
346: A0 00   LDY #300
348: B9 00 4F LDA 4F00,Y
34B: 99 00 55 STA 5500,Y
34E: C8      INY
34F: D0 F7   BNE 0348
351: A0 00   LDY #00
353: B9 00 50 LDA 5000,Y
356: 99 00 56 STA 5600,Y
359: C8      INY
35A: D0 F7   BNE 0353
35C: A0 00   LDY #00
35E: B9 00 51 LDA 5100,Y
361: 99 00 57 STA 5700,Y
364: C8      INY
365: D0 F7   BNE 035E
367: A0 00   LDY #00
369: B9 00 52 LDA 5200,Y
36C: 99 00 58 STA 5800,Y
36F: C8      INY
370: D0 F7   BNE 0369
372: A0 00   LDY #00
374: B9 00 53 LDA 5300,Y
377: 99 00 59 STA 5900,Y
37A: C8      INY
37B: D0 F7   BNE 0374
37D: A0 00   LDY #00
37F: A2 C0   LDX #C0
381: B9 00 54 LDA 5400,Y
384: 99 00 5A STA 5A00,Y
387: C8      INY
388: E8      INX
389: D0 F6   BNE 0381
38B: 60      RTS

```

## Mike Grabenstein

I have been subscribing to your magazine for about two years. I have a couple of questions for you and/or your readers.

**?** First off, is there any way to make an Apple //c boot from the second drive? I have heard of a hardware way to cause the boot but I can't find someone that knows.

**?** I have also tried to tinker with my //c, but the cover will not unclip; is there some trick to getting the cover off? And which cover does come off, the top half or the bottom half?

Softkey for...

### Dr. Ruth's Computer Game of Good Sex

The Avalon Hill Game Co.

#### Requirements:

- DeMuffin + (COMPUTIST8, page 5)
- DOS 3.3 Initialized Slave Disk
- Blank Disk

I drew a lot on how to break the game from COMPUTIST37, page 9, Avalon Hill Games - thanks John Howard.

**1** INITIALize the blank disk with HELLO as the boot program.

**2** Boot the game disk.

**3** Listen for the drive to read tracks 0 and 1 then press **RESET** after track 1 is read.

**4** Enter the monitor and move the protected RWTS to a safe place.

CALL -151  
8600<B600.BFFFFM

**5** Press **RESET** to re-enter FP Basic.

**6** Boot normal DOS 3.3 and BLOAD DEMUFFIN +.

PR#6

(you may have to PR#6 more than once)

BLOAD DEMUFFIN+

**7** Re-enter the monitor and move the protected RWTS into it's original place.

CALL -151  
B600<8600.BFFFFM

**8** Start DEMUFFIN+ and follow the prompts (Demuffin uses prompting as FID does).

803G

**9** Boot DOS 3.3 slave (turn off the computer and wait 15 seconds before turning back on).

**10** Swap Slave for your duplicate of Dr. Ruth's...

**11**

UNLOCK HELLO

**12**

LOAD HELLO

**13**

Delete line 25 or make it into a REMark.

**14**

You also have to give the CHR\$(4);'BLOAD''s the starting address of the file. Change each of the addresses in the following lines to the correct values shown. Change only the load address and leave the rest of the line the same.

```

30 PRINT CHR$(4);'BLOAD PART 1, A$7028'
35 ..... ,A$0300
40 ..... ,A$9200
112 ..... ,A$0378
130 ..... ,A$2000
145 ..... ,A$2000
312 ..... ,A$0378
327 ..... ,A$2000
345 ..... ,A$2000
512 ..... ,A$0378
520 ..... ,A$300,LS69

```

**15**

SAVE HELLO  
LOCK HELLO

**16**

Copy the back as you did the front, but don't worry about the HELLO program.

Softkey update for...

### Strip Poker

Artworx

The softkey for *Strip Poker* in COMPUTIST42, page 10, did not work on my copy of the game using standard DOS. But the procedure works fine if you copy Pronto-DOS on tracks \$0-2, and you don't have to wait for the girls to undress.

A short Softkey for...

### Saunders Chemistry C.A.I.

Demuffin Plus will deprotect *Saunders Chemistry C.A.I.*

...Except for line 3. Instead of Reseting, press **RESET**. And don't worry about the HELLO program.

## Donnie Cambre & Johnny Garrett

Softkey for...

### One On One

Electronic Arts

*One on One*, a basketball simulation starring Julius Erving, and Larry Bird, is a slightly old (1983) game but it is still a challenge for a lot of people. To copy *One on One* install the following controller into Super IOB and run it. When it is done it will have performed the following sector edits:

Track	Sector	Byte(s)	From	To
\$01	\$0F	\$00-02	4C 69 A0	18 60 DD
		\$6F-71	4C 69 A0	18 60 DD
\$0C	\$04	\$00-02	4C 69 BD	18 60 F9

This controller is very similar to the *Archon* controller. As a matter of fact, just add lines 5010-5120 to the *Archon* controller and you have it. This controller changes the altered data markings from D5 BB CF to a normal format.

### Controller

```

1000 REM ONE ON ONE CONTROLLER
1010 TK = 0 : ST = 0 : LT = 35 : CD = WR
1020 T1 = TK : GOSUB 490 : IF TK > 3 THEN
    RESTORE : GOSUB 210
1030 GOSUB 430 : GOSUB 100 : ST = ST + 1 : IF ST
    < DOS THEN 1030
1035 IF TK = 2 THEN GOSUB 210
1040 IF BF THEN 1060
1050 ST = 0 : TK = TK + 1 + (TK = 4) * 2 : IF TK
    < LT THEN 1030
1060 GOSUB 310 : GOSUB 230 : GOSUB 490 : TK =
    T1 : ST = 0
1070 GOSUB 430 : GOSUB 100 : ST = ST + 1 : IF ST
    < DOS THEN 1070
1080 ST = 0 : TK = TK + 1 + (TK = 4) * 2 : IF BF
    = 0 AND TK < LT THEN 1070
1090 IF TK < LT THEN 1020
1100 HOME : PRINT : PRINT "DONE*WITH*COPY" :
    END
5000 DATA 213 , 187 , 207
5010 DATA 11*CHANGES
5020 DATA 2 , 3 , 71 , 170
5030 DATA 2 , 3 , 81 , 173
5040 DATA 1 , 15 , 0 , 24
5050 DATA 1 , 15 , 1 , 96
5060 DATA 1 , 15 , 2 , 221
5070 DATA 1 , 15 , 111 , 24
5080 DATA 1 , 15 , 112 , 96
    
```

```

5090 DATA 1 , 15 , 113 , 221
5100 DATA 12 , 4 , 0 , 24
5110 DATA 12 , 4 , 1 , 96
5120 DATA 12 , 4 , 2 , 249
    
```

### Controller Checksums

1000	- \$356B	5010	- \$B5A9
1010	- \$3266	5020	- \$49E0
1020	- \$2E67	5030	- \$7037
1030	- \$3866	5040	- \$29FC
1035	- \$7B0B	5050	- \$4444
1040	- \$2B80	5060	- \$118C
1050	- \$7275	5070	- \$8297
1060	- \$0803	5080	- \$20D7
1070	- \$0006	5090	- \$D418
1080	- \$8D0D	5100	- \$BF16
1090	- \$4CD7	5110	- \$28FF
1100	- \$3107	5120	- \$1B21
5000	- \$E127		

Also, in the future, we would appreciate it if the authors of softkeys would show what the FROM bytes were originally because not all of the games have the same protection. This would prevent confusion.

## Michael Crawford

Softkey for...

### Infiltrator

Mindscape

#### ■ Requirements:

- Infiltrator* disk
- Copy II+ or
- Disk Muncher and a sector editor
- A blank disk

I recently purchased *Infiltrator*, one of the newer releases from Mindscape. This game is a direct descendant of *Captain Goodnight*, with a similar humorous style and an arcade adventure setting. You are Johnny McGibbits, one of those people who can do anything and makes everyone else feel like tossing their cookies. Of course it is up to you save the world. Off you go in your WhizBang helicopter, on your way to the headquarters of the evil Overlord. On the way, you must destroy or evade attacking aircraft. (Landing once you reach your destination is a trick in itself). Once on the ground, you must infiltrate the

headquarters, bringing back photographs of important documents and possibly destroying the headquarters. All of this must be completed in a certain amount of time. There are three missions to complete, each progressively harder as your expertise increases. Overall, this is an excellent game, especially for those who really enjoyed *Captain Goodnight* and are looking for a bit more of a challenge.

Upon purchasing the disk, my first thought was 'backup'. Of course, like most games these days, there was copy protection included on the disk at no extra cost. *Disk Muncher*, which is a fast copier and does little error checking, produced an unbootable copy. Now to find out what was up. I went back to the original disk and booted it. After a quick disk access, the program paused for a second and then continued with disk access. The copy crashed after the pause. Obviously the first disk check (or very close to it) was involved in the protection. Now it was time to get out *Copy II Plus* v7.1. This is an extremely flexible program. I began with a Hi-Res Disk Scan on tracks \$00 through \$23. Interestingly enough, there was some strange looking data on track \$23. More than likely, this too was involved in the protection.

Now I moved to the Sector Editor. Track \$00, Sectors \$00-\$09 is the only readable part of the disk in standard DOS 3.3 format. Errors occur on all other tracks. Here is where *Copy II Plus* really comes in handy. The program allows for patches to be made to standard DOS 3.3. Pressing [P] in the Sector Editor takes you to this utility. I selected a custom patch, and using trial and error, found that only the data checksums need to be ignored to read in all tracks and sectors on the disk. Pressing [ESC] two times brought me back to the Sector Editor and it was time to search. Assuming that the protection involved checking the disk, I searched for LDA C08C,X (hex string BD 8C C0). This occurs several times on track \$00, sectors \$00, \$02, \$06, \$07 and \$09. I started with sector \$00 and NOP'ed out several of the occurrences to see if the disk crashed before the copy protection was reached. Upon booting, the disk still made it to the pause and crashed as it had before. Now I moved the opposite end, sector \$09. The hex string BD 8C C0 occurred twice on this sector, both times very close to the beginning of the sector. The second occurrence looked very suspicious upon disassembly.

```

00/0E13: BD 8C C0 LDA $C08C,X
00/0E16: 30 03 BMI 0E1B, (+03)
00/0E18: EE 06 BF INC $BF06
00/0E1B: 18 CLC
00/0E1C: 60 RTS
    
```

Notice that if the comparison done at \$0E13 fails, the only consequence is that the increase by one instruction at \$0E18 is skipped. Asking myself the question 'What if the instruction at





\$0E18 was executed no matter what the disk check found?', I proceeded to remove the comparison by putting in two NOP instructions at \$0E16. After writing this sector back out to disk, I tried booting my copy. Voila! A perfect boot every time.

## COOKBOOK METHOD

**1** Boot *Copy II Plus* and go to the Sector Editor. Enter **P** for the DOS patch utility. Select 'CUSTOM' and set 'CHECK CHECKSUMS' under the heading 'DATA' to 'NO'. Press **ESC** three times to return to the bit copy menu. Now do a manual bit copy from the original of tracks \$00 to \$22. (Or copy the original using a fast copier like *Disk Muncher*).

**2** Return to the Sector Editor. **Make sure you remove the original disk to avoid writing anything to it.** Read in track \$00, sector \$09 from the backup. Make the following changes:

Track	Sector	Byte(s)	From	To
\$00	\$09	\$16	30	EA
\$00	\$09	\$17	03	EA

That's all she wrote !!

A.P.T. for...

### Infiltrator

If you alter any of the bytes on track \$01, sector \$0C you will get a surprise on booting. The game will ask you if you wish to start with mission 1 or mission 2. I have not played the game enough yet to see what effects this might have on the rest of the game. Who knows what other interesting features may lurk on this disk waiting to be unleashed!

## The Cracking Company

Softkey for...

### Super Sunday

Avalon Hill

You wouldn't believe how easy it was to deprotect *Super Sunday* by Avalon Hill. They must have thought a great deal of this game because, unlike the other AH releases, they mildly altered standard DOS 3.3.

Looking through the changes revealed the

write-translate table changed at \$BA29 and the read-translate table changed at \$BA00. Since we're only concerned with getting the disk read properly, \$BA00 (-17750) needs to be changed from \$AA in normal DOS to \$00. The simplest way to accomplish this task during a copy is with our old friend, Super IOB.

Here's how to deprotect *Super Sunday*.

## Step by step

**1** Boot up a standard Apple DOS 3.3 system master and INIT a blank disk with HELLO as the boot program.

```
PR#6
FP
INIT HELLO
```

**2** Load in Super IOB with the standard controller and insert the following lines:

```
1015 POKE -17750,0
1025 GOSUB 170
```

**3** Change the starting track in line 1010 from 0 to 3 (we don't want to wipe out our DOS).

```
1010 TK = 3:ST = 0:LT = 35:CD = WR
```

**4** Now run the Super IOB program and you'll find that you have unlocked *Super Sunday*!

There's just one small problem - no HELLO program. A boot of the disk will stop like any other disk without a HELLO program. So...

**5** Enter BASIC and create a one-line HELLO program.

```
10 PRINT CHR$(4) "BRUN BOOT"
SAVE HELLO
```

The boot file name is BOOT as evidenced by the catalog.

And now the program will boot and run and you can file your original away for safe keeping.

## D. M.

Softkey for...

### Arcade Boot Camp

Penguin Software

Regarding the subscriber's article on *Arcade Boot Camp* softkey, COMPUTIST38 page 4.

The method used depends on having the Senior Prom. Another method that is quick and efficient is as follows.

### Requirements:

- Arcade Boot Camp*
- Locksmith 6.0*
- A Blank disk
- Super IOB v 1.5 and Swap Controller

I will not go into great details of how or why this works. If you have been reading COMPUTIST attentively, you will understand the why and how. If you don't care about how and why, neither do I, so just perform the step-by-step procedure and all will work.

**1** Boot up *Locksmith 6.0* and select the A option (Automatic Boot Trace).

**2** Insert the *Arcade Boot Camp* disk.

**3** Press S to start the boot trace.

The program load is now under control of the *Locksmith* Boot Code Analyzer.

**4** Wait a moment after the disk drive light comes on. Then **RESET**.

You are now in the monitor, with the RWTS from the protected disk sitting at \$B800.

**5** We now move the RWTS down to \$1900.

```
1900<B800.BFFFFM
```

**6** Boot a slave disk with a very short hello program.

I have a disk that I save for this. On this disk, I keep only the Super IOB v1.5 program and the necessary controllers. The hello program just identifies the disk.

**7** BSAVE the RWTS to this disk.

```
BSAVE RWTS.ARCADE BOOT
CAMP,A$1900,LS800
```

**8** Now load in your Super IOB program, use the Swap Controller, and change line 10010 to 'BLOAD RWTS.ARCADE BOOT CAMP'. Run the program, and make your copy.

## Warnings!!!

*Arcade Boot Camp* uses a loader to pick up the program elements. The catalog only shows the Hello program the Loader and Applesoft (track \$11, sector \$C). The program elements are selected by the Loader program, and are zeroed in the VTOC but not claimed by a T/S list. So don't run FIXCAT or any other utility that will correct for Sectors that are not claimed by the T/S lists.



## Softkeying Softdisk Specials

I have used this method on several other Softdisk Specials like *Spy's Demise* and *Caverns of Freitag* with equal success. I do share the reader's opinion that Softdisk is a good buy. While I do enjoy these specials, it annoys me that I can back-up the rest of their disks, but not the side with the commercial programs. The method I used resulted from knowledge I gained in your fine magazine, and a little common sense.

When you copy *Spy's Demise*, heed this word of caution. Track \$12, sector \$6 is used by the program, but is not claimed by it, so be sure that track \$12, sector \$6 is free on the disk you copy the program to, and then zero it out in the VTOC.

*Arcade Boot Camp* takes up most of the disk, but *Spy's Demise* and *Caverns of Freitag* leave plenty of room for other programs on the disk.

## B. Dudley Brett

### Softkey for...

### Magic Slate

Sunburst Education

When I tried Glen Tatum's Softkey (COMPUTIST37, page 7) upon my *Magic Slate* diskette, the copy would not work. After a closer look, I was able to finally produce a COPYA-able version. In the process, I found out a few things which may be of interest to others who may own a *Magic Slate* diskette with some variation of protection.

Glen Tatum described a routine at \$21DB on his diskette which, after the nibble count proved unsuccessful on a copy, would be called. This routine simply prints an error message "unable to load MAGIC SLATE", and then hangs in an endless loop.

He proposed to edit the first three bytes at \$21DB to JMP over the routine to a pointer inserted following the conclusion, to the start of the main program. In my version (1.2), the pointer was not there!

My method is to just remove all references to this jump to the error message routine by finding and then removing all references to these jumps by NOPing them with EAs. The easiest way to find these references is to start by copying both sides of the *Magic Slate* diskette with any sector copy program that will ignore read errors on track \$01, then booting up ProDOS and Bloading the MS file (as Glen Tatum suggests). Get into the monitor and start

to list from \$2000. Find the first JSR to RAM. On the first side (MAGIC SLATE.80) this was JSR \$3845 located at \$203B, and was undeniably the nibble count. Upon the return at \$203E, a branch to \$2043 is made if the nibble count is successful. If not, a JMP \$21DB to the error message is taken at \$2040.

The rest is quite easy! Scan forward and find all other references to \$21DB up to the error message at \$21DB. I found this necessary as there was one branch to the error message very close to it, and a JMP was not used. Now replace the JMP's found with EA's and save the MS file back to the diskette. For the second side (MAGIC SLATE.40/20), JMP \$2230 accessed the error message and was likewise removed with EA's.

## The Softkey

**1** Use *Locksmith 6.0* Fastcopy (or any sector copy program which will ignore track \$01 errors).

**2** Boot ProDos.

**3** Catalog the copied *Magic Slate* diskette:

PREFIX,D1  
CATALOG

**4** Note the length of the MS System file (mine was \$3DFC) and jot it down somewhere.

**5** Load the MS file and enter the monitor.

BLOAD MS, TSYS, AS2000  
CALL -151

**6** Make the changes.

for MAGIC SLATE.80

Block	Byte(s)	From	To
\$2040	4C DB 21	EA EA EA	
\$20C1	4C DB 21	EA EA EA	
	\$21C1	B0 0BA EA	

(was BCC \$21DB)

for MAGIC SLATE.40

Block	Byte(s)	From	To
\$203D	4C 30 22	EA EA EA	
\$205E	4C 30 22	EA EA EA	

**7** Save MS.

UNLOCK MS

for MAGIC SLATE.80  
BSAVE MS, TSYS, AS2000, L\$3DFC

for MAGIC SLATE.40  
BSAVE MS, TSYS, AS2000, L\$3DEC

**8** Positively do **not** RENAME MS to MS.SYSTEM (won't work!).

**9** Relock the file.

## LOCK MS

After softkeying both sides of *Magic Slate* this way, you will now have COPYA-able diskettes!

## New Revised Softkey for...

### Science Toolkit

Broderbund

When I tried the softkey for *Science Toolkit* from Broderbund (COMPUTIST46) it became apparent that I had a different version (slightly!).

Stephan Lau's softkey had to be modified somewhat in order to produce a working copy.

Fortunately, the six sectors on the protected track \$1 were similar and could be captured in an almost identical way. However, I could not just move them to Track \$0, sectors \$4 to \$9 as suggested because my disk, unlike Stephen Lau's version, had data recorded upon sectors \$0-8, not just \$0-3.

Therefore, I decided to simply modify the *Toolkit* controller to write the \$1000-\$15FF code to sectors \$09-\$0E instead.

Here then is the new revised softkey.

**1** Put original in boot drive. Enter the monitor and modify the boot code before we start it.

CALL -151  
9600<C600.C6FFM  
96F8:A9 59 8D 80 08 A9 FF 8D 81 08 4C 01 08  
9600G  
C0E8

**2** Get rid of the nibble count.

1008:EA EA EA

**3** Insert a normal disk without a Hello program and boot it. Save the code.

6[P]  
BSAVE CODE, AS1000, L\$600

**4** Type in the Super IOB controller. Run it, making sure to format the disk with a volume of 1.

**5** Make the following sector edits:

Track	Sector	Byte(s)	From	To
\$00	\$00	\$62-64	FF 0E 0C	0C 0A 08
		\$65-67	0A 08 06	06 04 02
		\$68	04	FF
\$00	\$01	\$46	85	60
\$00	\$06	\$6C	BB	AA
\$08	\$08	\$6C	BB	AA

## Controller

```

1000 REM SCIENCE TOOLKIT CONTROLLER (REV. 1 )
1010 TK = 0 : ST = 0 : CD = WR : RESTORE : GOSUB
      170 : GOSUB 490
1020 GOSUB 430 : GOSUB 100 : ST = ST + 1 : IF ST
      < 15 THEN 1040
1030 GOSUB 230 : GOSUB 490 : ST = 0
1050 TK = 2 : LT = 33 : ST = 15 : LS = 15 : CD = WR
      : FAST = 1
1060 RESTORE : GOSUB 170 : GOSUB 490 : GOSUB
      610
1070 GOSUB 230 : GOSUB 490 : GOSUB 610
1080 IF PEEK (TRK ) = LT THEN 1100
1090 TK = PEEK (TRK ) : ST = PEEK (SCT ) : GOTO
      1060
1100 HOME : PRINT "COPY DONE!" : END
5000 DATA 222 , 187 , 222 , 170
10010 PRINT CHR$ ( 13 ) CHR$ ( 4 ) "BLOAD CODE ,
      "AS3000"
    
```

## Controller Checksums

1000 - \$356B	1070 - \$7340
1010 - \$98C6	1080 - \$2FB0
1020 - \$7711	1090 - \$B34A
1030 - \$F6C9	1100 - \$5DC7
1050 - \$A834	5000 - \$B941
1060 - \$F101	10010 - \$923B

Lam's technique for entering machine code, used in Stephen Lau's original controller, does not seem to work on my Apple (maybe the 65C02 microprocessor won't allow it!). I therefore decided to just use sector edits.

Note that the moving of the \$1000-\$1500 code after booting *Toolkit* is not required in my softkey. Fortunately it is saved starting on sector \$09 rather than \$04, and almost by luck will be read by Boot 0 into pages \$10 to \$15.

## Donato 'MOUSSE' Vytiaco

Softkey for...

### *Mind Mirror*

Electronic Arts

#### ■ Requirements:

- Two blank disks
- A copy program capable of copying specific tracks
- A sector editor

*Mind Mirror* from Electronic Arts belongs

to the category of 'personality' software, alongside *Mind Prober* and *Alter Ego*.

As with all previous EOA releases, the protection scheme used differs only slightly to those discussed in COMPUTIST24.

Like *Seven Cities of Gold* and *Adventure Construction Set*, the disk is in a DOS 3.3 format to begin with. It even has a normal catalog track on track \$11 to load some of its files from!

So the first step would be to copy the program side of *Mind Mirror* onto a blank, formatted disk, ignoring any errors on tracks \$05-\$06. Then copy side A of the Life Simulation Disk onto the back side of your program disk and Side B onto another disk.

Separating Sides A and B of the Life Simulation Disk is highly recommended for those who use two disk drives. You can use the back side of your second disk for saving files.

Only the program side of *Mind Mirror* is protected, and upon consulting my COMPUTIST library for information on EOA disks, I found out that parts of *Archon II*'s softkey were applicable. Changing bytes \$00-\$02 of track \$01, Sector \$0F, from 4C 69 A0 to 18 60 DD would allow your copy to work flawlessly, so it seems, until you await the results of your exercises where the program hangs.

After watching the "marked" disk drive arm of my Disk II move across the disk, I observed that it checked track \$05 two more times. And each time, the drive arm would come from track \$18, coming from track \$11, the catalog track. I then checked the files on the catalog to see which ones have data on track \$18. Examining the three files I found there, "FILEKU" contained familiar DOS checks.

BLOADing FILEKU from DOS at \$3000 and executing it surprisingly starts the disk check procedure. By replacing its first jump (JMP \$306A) instruction with an 18 60 6E, you're done! You can either do this on the program in memory then BSAVE it or on the disk itself. It can be found at track \$18, sector \$08 at bytes \$04-\$06.

## Summary

**1** Copy all three sides, ignoring errors on tracks \$05-\$06 of the program side.

**2** With a sector editor, make the following changes to your copy of *Mind Mirror*'s program disk.

Track	Sector	Byte(s)	From	To
\$01	\$0F	\$00-02	4C 69 A0	18 60 DD
\$18	\$08	\$04-06	4C 6A 30	18 60 6E

Now you have a deprotected version of *Mind Mirror*!

## Spectrum Lord

A note on...

### *Hacker I*

I would like to share a secret with my fellow readers. If you have *Hacker I* by Activision, the password at the start of the game is "AUSTRALIA".

## Ann Onymous

Update for J. Lewis's...

### Capturing Locksmith 6.0 Fastcopy

COMPUTIST43, page 12

My recently purchased *Locksmith 6.0* failed to respond to your Fastcopy extraction. The problem is that it expects some code at \$1D00 before it jumps to \$800.

To make it work, I followed your procedure with the following changes:

**BSAVE FC, AS2002, LS165F**

Alter the text file FC BUILD as follows:

In place of: 2284<50FE.63DDM  
use: **2284<50FE.665FM**

Instead of: 2244:62

use: **2244:E6**

Instead of: 2258:35

use: **2258:37**

In place of:

BSAVE LS 6.0 FASTCOPY, AS2000, LS1563

Use: **BSAVE LS 6.0 FASTCOPY, AS2000, LS17E7**

### Glen Bredon's PROSEL COPY

Good as Fastcopy is, it pales when compared to the COPY program which is part of Glen Bredon's PROSEL package. (Bredon is the author of the *Merlin Assembler*).

Unlike fastcopy, the PROSEL COPY program works on 3.5 inch disks and will copy them in one pass if you have a large enough RAM card. The PROSEL package, which contains many other utilities such as an intelligent block editor, is available directly by mail for \$40 from Glen Bredon, 521 State Road, Princeton, NJ 08540.



## La Balloonist

Maybe you or a reader can help me with my problem. My school just got *Type!* which is a great new educational game from Broderbund. If I don't figure out a softkey for *Type!* soon, my small school will be "playing" *Typing Tutor II* next year.

I think that it is unfair that a school should have to pay for a \$36 program 20 times just to use it on all its computers. A change that I think education publishers should make is to first deprotect their software, but if they refuse to do that, at least they should make the original able to boot on a system but then use a copyable data disk to keep it running, or better yet, have no disk access during the run time.

Anyway, I have taken the first layer of protection off.

Almost a softkey for...

**Typing!**  
Broderbund

**1** I initialized a disk.

### INIT HELLO

**2** I booted up *Type!* and a few seconds after the title screen came up I hit reset a couple times. If the screen fills with inverse @ then try again.

**3** You should get the Monitor prompt. Then, I moved RWTS to a safe place  
1900<B800.BFFFFM

**4** I booted up my IOB disk and saved the RWTS.

BSAVE TYPE! RWTS,A\$1900,L\$800

**5** Then I ran the swap controller with *Type!*'s RWTS.

The disk still won't boot past the title screen, so there must be some kind of check or nibble count somewhere.

Any help would be welcomed.

A.P.T. for...

## Lode Runner

You are a Galactic Commando deep in enemy territory. Power hungry leaders of the repressive Bungeling Empire have stolen a fortune in gold from the people by means of excessive fast food taxes.

Your Task? To infiltrate each of 150

different treasury rooms (and 60 more in *Championship Lode Runner*), evade the deadly Bungeling guards, and recover every chest of Bungeling booty.

### Requirements:

- Lode Runner*
- A blank disk
- Copy II Plus* or another sector editor

A friend and I formed an alliance with the Bungeling Empire and were playing with the *Lode Runner* edit board option and we designed two new boards that I thought were very challenging so we gave our masterpieces to the Bungeling Empire (because they let me copy their plans of their original 200 levels) to finance and build, to keep their treasure safe.

On the first board (level 1), my friend designed most of the puzzles, and I "drew" it up. The second board (level 2), took much less thought and was inspired by one of *Championship Lode Runner*'s boards. On level 1 there is an especially hard puzzle in the lower left of the screen. One that I had trouble figuring out even after my friend told me how it worked.

I will guarantee that both my boards are possible to solve as is. If you have a question on solving either of my boards you can write me at:

La Balloonist  
1404 Greenworth Place  
Santa Barbara, CA 93108

The people there will transfer your mail to my hideout!

### How to enter the levels

**1** Boot up *Lode Runner*, press **[E]** (Edit Menu). If you already have a Data Disk skip to step 2. Otherwise, press **[I]** (Initialize). Read the warning. Place your blank disk in the disk and follow the prompts.

**2** Now boot up your sector editor. Enter the sector editor and type in the hex-dump (Listing 1). Write it to track \$3, sector \$0 (level 1) of your Data Disk.

Type the second hex-dump (Listing 2) and write it to track \$3, sector \$1 (level 2) of your Data Disk.

**3** Boot up *Lode Runner* again. Put your Data Disk in drive one and start playing normally. If you ever pass both the levels and you haven't added a third level (via the Editor) then the program will use level 1 for level 3.

If you have trouble passing the levels (you will) you may want more men or to skip to the next level.

**[P]** adds additional lives per players (**[2]** on //e or newer).

**[N]** advances a level (**[6]** on //e or newer).

If you need more help you can edit my levels or make your own using these steps:

**A** Enter the EDIT mode from demo mode, press:

**[E]**

**B** Place INITed data disk into drive.

**C** Design your own game!

I-J-K-M keys move the cursor

0-9 keys make shapes

**D** Save your game. Press:

**[S]**

**E** Play your game.

**[Q]** to 'Quit' game generator

**[P]** to 'Play'

In edit mode you may use these options:

**E** (for "Edit")  
**P** (for "Play")  
**I** (for "Initialize")  
**C** (for "Clear")  
**M** (for "Move")  
**S** (for "Score")

For a more descriptive and complete explanation refer to the *Lode Runner* documentation.

### Listing 1

```
Track $03 Sector $00...
00 00 00 00 40 44 44 44
00 00 10 70 00 00 40 44
44 44 03 01 00 34 00 00
30 02 11 32 03 00 00 00
00 81 11 17 11 21 31 22
02 30 03 00 00 00 20 11
11 02 11 01 31 00 00 30
03 00 11 21 42 10 71 11
13 71 20 22 22 22 03 00
31 01 70 11 11 11 13 11
21 00 00 11 03 00 30 81
51 31 11 17 13 11 01 00
13 87 03 09 30 11 00 31
00 15 03 17 04 00 13 11
13 11 31 07 17 17 11 11
00 17 14 11 11 11 13 88
31 44 00 00 11 11 15 11
10 11 10 11 13 77 31 44
11 11 01 00 00 00 10 01
10 11 13 01 31 44 11 17
11 11 30 33 13 11 71 01
03 00 31 44 11 17 11 11
33 33 73 03 00 00 03 00
01 44 11 11 07 00 33 33
03 13 11 11 03 00 03 00
00 03 00 00 33 33 33 13
74 77 23 22 12 21 02 22
```



```

22 22 22 22 22 13 30 31
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00

```

## Listing 2

```

Track $03  Sector $01...

37 33 33 33 33 37 33 33
33 37 33 33 33 73 33 73
33 37 33 33 33 38 33 33
33 33 33 33 33 33 33 33
33 73 33 33 37 33 33 33
33 33 33 33 73 33 33 33
33 37 33 33 33 73 33 33
33 73 33 33 33 33 33 33
33 33 33 33 33 33 33 37
33 33 37 73 33 33 73 33
73 33 73 33 33 33 33 33
33 33 33 33 33 33 33 33
33 33 33 33 73 33 33 78
78 73 78 38 33 73 33 33
37 33 33 33 33 11 11 11
11 31 33 33 33 73 33 33
33 33 33 33 33 39 33 37
33 37 33 33 33 37 33 37
33 37 33 33 33 33 33 33
33 33 33 33 73 33 33 33
33 37 33 33 33 73 33 33
33 33 33 33 33 33 33 33
33 33 33 33 33 33 33 33
33 33 37 33 33 33 33 37
33 33 33 37 33 33 33 73
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00

```

## Henry P. Dowst

Softkey for...

**Shanghai**  
Activision

After reading John R. Nicholson's softkey in COMPUTIST46 and Larry Rando's softkey for *Great American Cross-Country Race* in COMPUTIST39, I was able to come up with the following softkey which deprotected my copy of Activision's *Shanghai* game for the Apple II Plus, //e, and //c.

**1** Copy the *Shanghai* disk with *Copy II Plus*, or any other standard disk copier program.

**2** Using your favorite sector editor, look for the string which begins A9 56 85. I found it on Track 5, Sector 5, beginning at byte 58.

**3** Look ahead until you find the string 38 2A 25 FC. I found this string beginning at byte 97.

**4** Starting with the byte with the value A9 located in step 2 above, change all bytes to EA until you have changed the bytes with values 38 and 2A identified in step 3 above.

**5** Change the bytes with values 25 FC to values A9 FF.

Enjoy the feeling of not running a program from an unbacked-up master.

## John Bauer III

Problem with a Softkey for...

**Zorro**  
COMPUTIST44

Help!

I tried to back up *Zorro* (Datasoft, COMPUTIST44) but it didn't work. The copy would boot and the game would run, but if you tried to pick up an item (Key) you could only hold the item for about three seconds. After the three seconds are up, the key was automatically dropped. You would then have to pick it up again. This makes it very difficult to win the game.

If anyone out there can help get rid of this annoying problem, I would be very grateful.

Softkey for...

**Bard's Tale II: Destiny Knight**  
Electronic Arts

I am a super *Bard's Tale* fan and have both the *Bard's Tale* games.

The softkey in COMPUTIST30 didn't seem to work on the new *Bard's Tale II: The Destiny Knight*. So I set about trying to make myself a back-up copy. Here is all I had to do for my version:

**1** Copy the boot side with any copy

program that will ignore errors on tracks 5 and 6.

**2** Get out your handy-dandy sector editor and make this simple change:

Track	Sector	Byte(s)	From	To
\$01	\$06	\$08	?	08

**3** Now copy the other three sides with any copy program.

Playing Tips for...

## Donkey Kong

One last thing, an Playing Tip for *Donkey Kong*. Anytime that you are playing the game, just press  and you will get a free life. You can keep hitting the  key and get even more lives. I hope this works on your version!

## Keith Parker

Here's a neat little program that I recently created. The program flips between the two hires pages continuously until you press a key. You may be saying: "What's so neat about that?". Well, for a few reasons.

1.) The program is relocatable. (Meaning it can be BRUN at any location.)

2.) The delays can easily be changed to suit anyone's needs.

3.) You may not know this, the game *Karateka* by Broderbund uses this procedure to perform the slick animation used in the game. By taking out the keypress you can use this routine as a subroutine for your own games/programs.

## Typing It In

**1** Enter the monitor with:

CALL-151

**2** Type in the following hex dump.

1FD1:	8D 10 C0 8D 50 C0 8D	\$2D10
1FD8:	52 C0 8D 57 C0 8D 54 C0	\$747E
1FE0:	A0 FF A2 FF CA D0 FD 88	\$52D3
1FE8:	D0 F8 8D 55 C0 A0 FF A2	\$5C52
1FF0:	FF CA D0 FD 88 D0 F8 AD	\$67AE
1FF8:	00 C0 10 E1 8D 10 C0 60	\$5C6B

**3** Save the program.

BSAVE FLIP.ROUTINE,A\$1FD1,L\$2F



Here is the disassembly of the program.

```

1FD1- 8D 10 C0 STA $C010 clear the keyboard
1FD4- 8D 50 C0 STA $C050 set graphics mode
1FD7- 8D 52 C0 STA $C052 full page graphics
1FDA- 8D 57 C0 STA $C057 display hi-res
1FDD- 8D 54 C0 STA $C054 display page 1
1FE0- A0 FF LDY #$FF # of loops
1FE2- A2 FF LDX #$FF # to count from
1FE4- CA DEX subtract 1 from X
1FE5- D0 FD BNE $1FE4 if not 0 cont count
1FE7- 88 DEY subtract 1 from Y
1FE8- D0 F8 BNE $1FE2 if not 0 cont loop
1FEA- 8D 55 C0 STA $C055 display page 2
1FED- A0 FF LDY #$FF 2 of times to loop
1FEF- A2 FF LDX #$FF # to count from
1FF1- CA DEX subtract 1 from X
1FF2- D0 FD BNE $1FF1 if not 0 cont count
1FF4- 88 DEY subtract 1 from Y
1FF5- D0 F8 BNE $1FEF if not 0 cont loop
1FF7- AD 00 C0 LDA $C000 get keypress
1FFA- 10 E1 BPL $1FDD if no key, do again
1FFC- 8D 10 C0 STA $C010 store key in buffer
1FFF- 60 RTS return to caller

```

Now whenever you want to use it, simply load in your pictures at \$2000 and \$4000 respectively and:

### BRUN FLIP.ROUTINE

and save the whole thing with:

### BSAVE WHOLE THING,AS1FD1,LS402F

Whenever you want to see your show, simply type:

### BRUN WHOLE THING

Hope this will be helpful to someone.

## James E. Mueller

### Adventure Tips for...

### Bard's Tale II

Dark Domain: You must grow wings in order to save the Princess.

The Tombs: In order to kill the giant, you must be intoxicated.

Fanskar's Castle: The white shall lie, and this knowledge reflects your only answer.

Dargoth's Tower: Water, lie, slave, gold, hate, rooster, large, early, bard, women. Drop all your items or you are lost.

Maze of Dread: The riddler seeks a word, but he'll only understand it backwards. Though seeming of little value, repetitiveness is definitely endurable.

Oscon's Fortress: Pits of Fire, Land of Krill,

Sword of Silence. Dervak is the last destroyer. The word is still - East = Scissor, South = Rock, West = paper.

The Grey Crypt: The Sphinx is very wise. A digital will not work only an analog.

Destiny Stone: The Plan is near. Don't leave the square room on level 2. Zen Master must lead and have the ring. The answer to the snare lies on page 29 of your Bard's Tale II instruction manual.

## Matt Ownby

### Adventure Tips for...

### Ultima IV

I have found more Adventure Tips and APT's for Ultima IV:

### Spells

**Resurrect** - Reagents: Sulpher Ash, Ginseng, Garlic, Spider Silk, Bloodmoss, Mandrake Root

**Gate Travel** - Reagents: Sulpher Ash, Black Pearl, Mandrake Root.

### Tips

**Sextants** - Ask for Item D at any guild shops. (Guild shops are located in Buccaneer's Den and Vesper.)

### Mantras

Virtue	Mantra	Rune	L incity	Shrine L.
Valor	RA	B'O" B'O"		O'F" C'E"
Justice	BEH	A'G" A'N"		A'L" E'J"
Honesty	AHM	A'G" A'I"		E'C" O'J"
Honor	SUMM	B'N" A'C"		M'P" F'B"
Spirituality	OM	A'I" B'B"*		B'D" K'G"*
Humility	LUM	B'N" B'M"***		N'I" O'H"
Sacrifice	CAH	B'O" B'M"		C'N" M'N"
Compassion	MU	A'B" B'J"		F'M" I'A"

\* The rune is located in the castle Britannia in the treasure room. The shrine is in portal #5 when both moons are full.

\*\* The rune is located in the village paws. The shrine is guarded by a lot of deamons. It is not impossible to reach.

**Bells, Candles and Books:** The book of truth is in the Lyceum in a "Jimmy-locked" library. Once you get in, search right below the R in the library.

The candle of lobe is in Cove, a hidden village near the Shrine of Sacrifice. Go across two bridges and if you're lucky there is a pirate's ship. Kill the pirates and sail south until you get to the Cove. Once inside go to the room with the Ankh and "Dispel" the upper-left energy field. Go into the secret room and Search it.

(If you can't get a pirate's ship you can use Ray Darrah's **Ulti-mainland Editor** found in COMPUTIST33 and edit a path leading to it.)

The Bell Of Courage lies in the sea, reachable only by boat. It is in a deep patch of water at the locations of: N' A'', L' A'.

**Britannia Observatory:** When you Search in the Britannia observatory you find a device with the letters A-P. When you type in a letter it shows you a picture of a city, castle or village.

Here is a list of those things:

### Castles:

- A Britannia
- B The Lyceum
- C Empath Abbey
- D Serpant's hold

### Townes:

- E Moonglow
- F Britain
- G Jhelom
- H Yew
- I Minoc
- J Trinsic
- K Skara Brae

### Ruins:

- L Magincia

### Villages:

- M Paws
- N Buccaneer's Den
- O Vesper
- P Cove

I would like to review tips made in COMPUTIST42 by Dr. Destruction.

Nightshade found at J' F" C' O" and also at C' M" M' N" when both moons are dark.

Mandrake Root found at D' G" L' G" when both moons are dark.

## Tom Johnson

### Playing Tips for...

### Conan

Here's another playing tip for Conan that I've known for some time. When you get to the top



of level 3 where you get the jewel, wait for the bird to fly by again. When he comes in range, jump and hit him and it will give you an extra man (and by the way, remember to stay on the bubble as long as you can and watch your points add up!). This has worked on all versions I've played.

If anyone knows how to copy or crack *Greeting Card Maker* by Activision or *Award Maker Plus* by Baudville please write and tell how.

My brother and I like to compile our favorite programs for the fullest extent of our use and for putting them into memory with *Diskquick*. You need deprotected programs for that. Your magazine has been a great help in this.

## Clay Ruth

Softkey for...

**Willy Byte**  
Data Trek

I see that *Willy Byte*, by Data Trek, is still on your Most Wanted List (which, I'm happy to see, just reappeared after a long absence). *Willy Byte* is an excellent animated graphics and sound experience that performs frequent disk access to load various parts of the game. This is just the kind of game for which you most want to keep a backup on hand, since normal play puts a lot of wear and tear on the disk. Unfortunately, *Willy Byte* is very heavily protected. But it need not remain that way! All you need are the following items (in addition to your original *Willy Byte* disk):

### ■ Requirements:

- A blank disk, notched for two-sided use
- A write protect tab
- Your favorite sector copier. One that ignores errors and normalizes the format; e.g., *Locksmith* Fast Disk Backup.
- If your sector copier doesn't ignore errors, you will need a nibble copier (e.g., *Copy II Plus*, *Nibbles Away*, etc).
- Your favorite sector editor (e.g., ZAP from *Bag of Tricks*)

*Willy Byte's* protection was not added as an afterthought. It was written right into the game. And the programmer must have been a disk cracker himself, for he anticipated everything you might do to defeat the protection. Critical

game subroutines are laced with code that verifies that the protection has not been altered, and with other code that actually restores several critical protection opcodes to their intended values! Several patches must be made to circumvent not only the protection but also the verification and restoration routines. Here's a rundown on the process:

**1** Side 2 is the bootable side. It is essentially normal. Copy it with your favorite sector copier. This side **must** be write-protected when you boot it. However, before you affix the write-protect tab, perform the following sector edit:

Track	Sector	Byte(s)	From	To
\$00	\$01	\$45	D8	00
		\$4A	B6	E0

These occur within the sequence (at \$44): A9 D8 8D F2 03 A9 B6 8D F3 03. This change will enable you to **RESET** out of the game instead of having to power down the computer.

**2** Side 1 is the playing side. It is normal except for tracks \$5 and \$6, which are formatted identically (both look like Track \$5) and are in perfect synchronism so that the drive will detect Track \$5 whether it is positioned at \$5, \$6, or anywhere in between.

If you use *Locksmith* Fast Disk Backup, it will normalize Track \$6 for you. However, if you use a nibble copier, Track \$6 will still look like Track \$5, preventing COPYA from reading it. You can normalize it by using INIT (from *Bag of Tricks*) to initialize Track \$6 only.

Do not preserve data, as there is no data to preserve. Alternatively, you could start by INITING side 1 with DOS 3.3, then nibble copy only tracks \$00-\$04 and \$07-\$22. Do not write-protect this side, as the program saves high scores and other information on this side.

**3** Use your sector editor to look at Track \$13, Sector \$C. At Byte \$48 of this sector begins the heart of the protection code that checks tracks 5 and 6 for uniformity. Defeating it with an RTS is easy, but every possible place you can do this is double-checked and restored by other routines scattered all over the game. In fact, the first part of this routine itself actually restores original protection code in a couple of other vulnerable places!

After much consternation, I finally decided to put the RTS at Byte \$4A, the third byte of this routine, in place of the opcode (\$8D) of a STA \$B7EC instruction. Look for the sequence A9 04 8D EC B7 AD F4 B7; change the \$8D to \$60.

**4** Still on Track \$13, Sector \$C, look at Byte \$80 for the sequence A9 05 C5 2E F0 03 EE 6E 69 68 A8 C0 0C D0 E1 CE; change the \$E1 (at Byte \$8E) to \$E0. This changes the

offset of a BNE instruction so that this section of code will work properly. This is one of the many self-modified areas of this program that no longer self-modifies properly as a result of other necessary patches described below. You see, *Willy Byte* does not execute properly as written; it requires complete execution of the protection code to make the program itself work properly! Having defeated the protection, we have to modify the program ourselves to make it work!

**5** Now look at Track \$17, Sector \$0. Way down at Byte \$EF is another copy of the protection code we patched on Track \$13. Look here for the same sequence given in Step 3, and make the same patch; this time the affected byte is at \$F1. Various copies of the protection code are called from various parts of the game, so we have to defeat them all.

**6** Move along to Track \$17, Sector \$1. This is a continuation of the code we just patched, and it also suffers from a now-uncorrected offset in a BNE instruction. The sequence here is slightly different from Step 4. Look at Byte \$27 for the sequence A9 05 C5 2E F0 01 00 68 A8 C0 0C D0 E3 CE; change the \$E3 (at Byte \$33) to \$E2.

**7** Now take a look at Track \$1A, Sector \$7. Here is yet another copy of the protection code, starting at Byte \$DE. Look for the same sequence given in Step 3, and make the same patch; this time the affected byte is at \$E0.

**8** Continue to Track \$1A, Sector \$8. Here, again, the code continues with a BNE offset that must be fixed. This time the sequence matches Step 6 and starts at Byte \$16. The patch (change \$E3 to \$E2) is at Byte \$22.

**9** Our next stop is Track \$12, Sector \$C. This code verifies critical bytes of protection code at \$81A2, \$8150, and \$814A. Since our Step 3 modified the byte at \$814A, we must NOP out the branch that follows this particular comparison. Look at Byte \$0D for the sequence 8A CD 4A 81 D0 8E CA BD 00; change both the \$D0 and the \$8E (at Bytes \$11 and \$12) to \$EAs.

**10** Now look at Track \$13, Sector \$3. Bytes \$53 through \$6F are a subroutine devoted solely to the verification of protection code. A pointer on page 0 is aimed at \$8148, the address of the protection routine we modified in Step 3. The accumulator is loaded from a table of expected values, and the Y-register is loaded from a table of corresponding offsets (into the protection routine) at which those values are to be sought. The rare but powerful indirect-indexed compare instruction (opcode \$D1) is used to verify the code. At Byte \$65, all mismatches are vectored to a disastrous BRK

instruction. Look at Byte \$5B for the sequence A2 31 BD 70 F8 BC F9 69 D1 02 D0 05 CA 10 F3 60. Change both the \$D0 and the \$05 (at Bytes \$65 and \$66) to \$EAs.

**11** Finally, look at Track \$22, Sector \$7. Here, in the midst of a critical game subroutine, is a section of code that restores original values to some lookup table bytes used by yet another code verifier.

By changing an ADC \$25 to a LDA  $\pm$ \$60, we make it install RTS instructions instead of the original STAs that we so carefully defeated.

Look at Byte \$8E for the sequence 8D 11 70 8D 0C 70 8D 14 70 65 25 8D 09 70 8D 23 70. Bytes \$97 and \$98 contain \$65 and \$25 respectively; change them to \$A9 and \$60 respectively.

Whew!!!

It was quite an ordeal, but by now your copy of *Willy Byte* should run properly. If it still wants to reboot, even though you found all code exactly as described above and you made all changes as specified, your nibble copy may have been bad. Scan the disk for bad sectors and re-do any defective tracks.

Now that you have a good working copy, here are a couple of tips:

Track \$21, Sector \$F holds the high scores. Use your sector editor to play around with these. The format is Rank (in hexadecimal), Name (22 characters, ASCII capitals, MSB set), Date (in BCD), Level (in hexadecimal).

Track \$22, Sectors \$0 through \$5, holds the various messages that *Willy Byte* tries to move through the computer. The first byte of each of these sectors indicates how many of that sector's messages have been used, so that a different message will come up each time you play. You can adjust these to make your favorite message come through, and you can edit the messages to suit your taste (or lack thereof).

To use *Willy Byte* on the IIGs, use the IIGs Control Panel to set the System Speed to Normal. In the Options section, disable keyboard buffering; this is necessary because the program doesn't properly reset the keyboard strobe after each attempted read.

*Willy Byte* is always fun to play, but to fully appreciate the audiovisual experience of this game, you should play it on a *Mockingboard*-enhanced Apple. The music, voices, and sound effects are superb.

Even the IIGs can use the *Mockingboard* if it is placed in Slot 4 and the Control Panel is set for Slot 4 = Your Card. If you've been playing *Willy Byte* without a *Mockingboard*, find someone who has one and find out what you've been missing. You'll never forget it!

Put your original disk away in a safe place, and have fun with *Willy Byte*!

## Fay Gagnon

Softkey for...

### Cat'n Mouse

#### ■ Requirements:

- Apple II
- Blank Disk
- Super IOB v1.5

*Cat'n Mouse* is an educational software program for ages 5 to 12 (Grades K-6). It combines a maze game with the teaching of word and picture associations. The difficulty of *Cat'n Mouse* can be upgraded as the Player's ability improves. New word/picture sets can also be built to keep up interest. The program is truly "open-ended". It is also copy-protected.

#### Protection

The disk has a normal DOS 3.3 format except for track \$10 (16). Track \$10 is only FF's with a few ID bytes added. The code which searches for these ID bytes is on track 1, sector 1. It is loaded into memory at \$B600.

#### Procedure

A simple return (\$60) placed at the beginning of the checker code will stop the search for ID bytes on track \$10. There is then no reason to have track \$10 with all its oddness. The resulting backup is COPYA-able.

**1** Write-protect the program disk and put the IOB controller into Super IOB 1.5.

**2** Start Super IOB. Let it format the blank disk (vol 254).

Super IOB will copy the program disk, skipping track \$10 and doing the sector edit on track 1, sector 1, byte 0.

#### Controller

```
1000 REM CAT'N MOUSE CONTROLLER
1010 TK = 0 : LT = 16 : ST = 15 : LS = 15 : CD = WR
      : FAST = 1
1020 GOSUB 490 : GOSUB 610
1024 T1 = TK : TK = PEEK (TRK) - 1
1025 RESTORE : GOSUB 310 : TK = T1
1030 GOSUB 490 : GOSUB 610 : IF PEEK (TRK) =
      LT THEN 1045
1040 TK = PEEK (TRK) : ST = PEEK (SCT) : GOTO
      1020
1045 IF LT = 16 THEN TK = 17 : LT = 35 : GOTO
      1020
```

```
1050 HOME : PRINT "COPY'DONE" : END
1100 DATA 1*CHANGES , 1 , 1 , 0 , 96
```

#### Controller Checksums

1000 - \$356B	1030 - \$2A2F
1010 - \$1A22	1040 - \$CACC
1020 - \$6EF2	1045 - \$141C
1024 - \$9166	1050 - \$1CDF
1025 - \$D873	1100 - \$7857

## Jack Moravetz

Thanks to John Nicholson's softkey (COMPUTIST46) for *Labyrinth* and Larry Rando's softkey (COMPUTIST39) for *Great American Cross Country Road Race I* was able to softkey Gary Kitchen's *GameMaker* which was on the Most Wanted List, *Top Fuel Eliminator*, and *Portal*. They all had the same protection only in different locations. *Term Paper Writer* had a little different nibble count routine, but it's been disabled also.

Softkey for...

### Gamemaker

Activision

I first made a copy of the original with *Locksmith 6.0* Fastcopy although any normal fast copier will work. Using John Nicholson's bytes to search for (A9 56 85), I found them on track \$21 sector 05. I also found the other two bytes he mentions in his article (25 FC) on track \$21 sector 05. I tried changing the 64 bytes from the A9 56 85 to the 25 FC to EA. I also changed the 25 FC to A9 FF as in the *Labyrinth* softkey. The disk booted and seemed to function normally.

#### Step by step

**1** Copy the original with any copier.

**2** Using a search utility, scan for the bytes A9 56 85. (I found them on track \$21 sector 05 beginning at byte 54)

**3** Use a sector editor to change the bytes from byte 54, which is currently A9, to byte 93 to EA. (Sixty-four bytes)

**4** Change the 25 FC at bytes 94 and 95 to A9 FF

**5** Write track \$21 sector 05 back to the disk.





Softkey for...

## Top Fuel Eliminator

Activision

- 1 Copy with any copy program
- 2 Using a search utility, scan for the bytes A9 56 85. (*Top Fuel Eliminator* had them on track \$16 sector E)
- 3 Use a sector editor to change the bytes from the A9 56 85 to the 25 FC to all EA.
- 4 Change the 25 FC to A9 FF and write the sector back to disk.
- 5 You may try adding a faster DOS to the backup.

Softkey for...

## Portal

Activision

- 1 Copy the disk with any copy program.
- 2 Using a search utility, scan for the bytes A9 56 85. (*Portal* had them on track \$11 sector 01)
- 3 Use a sector editor to change the bytes from the A9 56 85 to the 25 FC to EA.
- 4 Change the 25 FC to A9 FF and write the sector back to disk.

This seems to be one of Activision's more popular nibble count routines, but thanks to some previous softkeys it's now disabled.

Softkey for...

## Term Paper Writer

Activision

- 1 Copy the disk with any copy program.
- 2 Using a search utility, scan for the bytes 20 51 2D. (I found them on track \$22 sector 09 byte 39)
- 3 Use a sector editor to change the 20 51 2D to EA EA EA.
- 4 Search for the bytes 4C CF 2D 00 (I found them on track \$22 sector 09 byte C1) and change the 00 to 55 (4C CF 2D 55).
- 5 Write the sector back to disk.

Many thanks to Jim S. Hart for his numerous softkeys. He always seems to do the ones I need most.

## S. L.

Softkey for...

## Reader Rabbit

Learning Company

The softkey for *Alice in Wonderland* in COMPUTIST35, page 8 works perfectly for *Reader Rabbit*.

The result is a completely catalog-able copy which I've transferred to 3.5 inch disk operating under Microsparc's *UniDOS 3.3 Plus* which in turn works fine on my IIGs.

A IIGs note on...

## Stickybear Math

*Stickybear Math* will not work on the IIGs. However, *Copy II Plus v7.4* autocopy produces a copy of *Stickybear Math* that works just fine on the IIGs!

## John Busch

Another Softkey for...

## Millionaire

Blue Chip

I've noticed quite a few softkeys for *Millionaire* by Blue Chip. I tried a couple and they didn't work and one was just too long to try.

But I noticed when *Millionaire* was booted it booted normally with a cursor showing at the bottom. So I pressed:

**ⓄC**

and stopped the Applesoft boot program that begins *Millionaire*.

Then I captured its RWTS like I've seen in other softkeys by entering the Monitor and moving it to a safe location:

CALL -151  
1900<B800.BFFM

Then I booted up a disk with no boot program, saved the RWTS code:

BSAVE RWTS.MILLIONAIRE, A\$1900, L\$800

and ran it through Super IOB with the swap controller and that was it!

There is some extra memory on the disk so I put other programs on the deprotected copy.

Softkey for...

## Arctic Fox

Electronic Arts

Use a disk copier that ignores disk read errors (Track 6 doesn't need to be copied.) Then take a sector editor and do the following edits:

Track	Sector	Byte(s)	From	To
\$01	\$0F	\$00	18 60 DD	
\$01	\$0A	\$52	18 60 48	

That's all I did with my version and it works fine.

Here are some playing tips.

Some Playing Tips for...

## Hard Hat Mack

Like on *Cannonball Blitz*, if you hold down the jump key before the second level you can get safely onto the ramp going to the crane. You can start playing on any level by pressing 1-3 and then the key that starts the game.

## Spy Hunter

Press "P" when you start to get more weapons.

## Orbitron

Press **ⓄR** while playing then 1-8 to play on that level.

## Print Shop Companion


Boot the back side of *The Print Shop Companion* and when the drive stops press **ⓄG** and you can play "Driver".




## Steve Bilgrien

As of yet, I am not a subscriber to COMPUTIST, but I have ordered a number of back issues, all of which were superb.


In one of those issues, COMPUTIST36, you published an article and program by Joe Montano entitled "The Bard's Dressing Room". This program was designed to edit characters created for *The Bard's Tale*...

 I would like to know if it will also work for Bard's Tale II.

I do not own *The Bard's Tale*, but recently purchased *Bard's Tale II* and would like to "dress up" my characters. Due to the length of "The Bard's Dressing Room", however, I have yet to type it in and try it out since it may not work. Any information on getting "The Bard's Dressing Room" to work for *Bard's Tale II* would be greatly appreciated.

 What about it, J. Montano?..RDEXed

## Steve Garland

 Do you think anyone could figure out a way that I could use my AMDOS 3.5 inch disks as storage files for *Print Shop Graphics*?

I cannot figure out a way to get *Print Shop* to look to slot 5, drives 1 and 3 which is the way AMDOS makes DOS 3.3 available on the 3.5 inch disk drives.

Any help you can offer would be greatly appreciated.

## Jim S. Hart

A quick A.P.T. for...

### Montezuma's Revenge

I have recently been playing *Montezuma's Revenge* and have enjoyed it. My complaint is that I die too quickly, as the result of clumsy fingers and short attention span.

Hacking around through the code reveals that zero page location \$E0 is where the number of lives are kept.


The instruction to decrement \$E0, in

hexadecimal form, is C6 E0, replace it with A9 01 and write that sector back to disk.

You now can never die in the game.

## Laser 128 'Absolute' RESET

I have a Laser 128 and to change the printer & modem port settings, Laser owners have a built-in control panel, kind of like the Iigs's control panel.

To access the control panel, you must press —  **RESET** (in any order as long as the **RESET** key is the last one released) or hold down the **P** key when you turn the Laser on.

I was curious as to where in the ROM code the control panel was so I could modify it, like using POKES in my hello programs to set the ports instead of having to go to the panel.

I came across a fascinating discovery: **you can go into the monitor whenever you please**, much like having Don Lancaster's "Absolute Reset"!

When going to the control panel, if the **M** key is substituted in place of the **P** key, a jump to the monitor ensues regardless of where the reset vector is pointing to.

The funniest thing is that this is mentioned NOWHERE in the Laser 128 documentation! I'm definitely glad that I found the extra little 'feature' because I can now follow even more COMPUTIST softkeys than before.

**Note:** this jump to the monitor feature is in the 2.9 version of the Laser's ROMs. I do not know if it is present in other versions.

## Ron Mundel

Playing Hint for...

### Ultima IV

I'd like to pass this little tid-bit along. If you use the magic spell Negate, it negates all magic, including your own spells while negate is in effect.

Maybe you already know this. I didn't and I thought I had read everything. But I didn't see that piece of information til now. I was so involved in the fight sequences, that I did not pay attention to anything but actual combat strategy.



# MOST WANTED Softkeys

- ABM Muse
- Accolade Comics Accolade
- Agent U.S.A. Scholastic
- Airheart Broderbund
- Algeblaster Davidson & Associates
- Ballblazer Epyx
- Bandits Sirius Software
- Bank Street Filer Broderbund
- Brain Bank The Observatory
- Create with Garfield DLM
- Cross Clues Science Research
- Crypt of Media Sir Tech
- Earth Orbiting Station (EOS) Electronic Art
- Electric Crayon Polarware
- Fay: The Masked Woman Digitech Software
- Fay's Word Rally Digitech Software
- Fay: Word Hunter Digitech Software
- Fun Bunch Unicorn
- Gemstone Healer SSI
- Goonies Datasoft
- Gutenberg Jr. & Sr. Micromation LTD
- Handicapping System Sports Judge
- J & S Grade Book J & S Software
- Jet V I.O Sublogic
- Jigsaw Microfun
- Kitchen's Game Maker Activision
- Le Francois Par Ordinateur D.C. Heath Co.
- Mathblaster Davidson & Associates
- Microzines (Current)
- Odin Odesta
- Operation Frog Scholastic Software
- Peeping Tom Microlab
- Personal Finance Manager Apple Computer
- Prime Plotter Primesoft Corp.
- Print Master Unision World
- Quiz Castle Digitech Software
- Snoggle Broderbund
- Super Boulder Dash Electronic Arts
- Super Factory Sunburst
- Type Broderbund
- Visiblend Microlab
- Where in USA is Carmen Diego Broderbund
- Work Force II Core Concepts
- The Works First Star Software
- Zorro Datasoft

# PRODUCT MONITOR

## Software Reviewed:

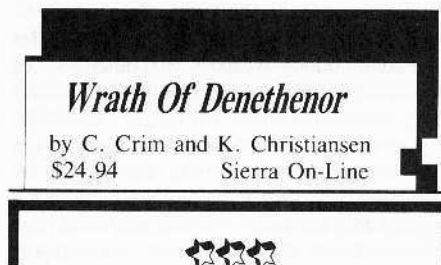
- Wrath Of Denethenor* Sierra On-Line
- Graphics Scrapbook* Epyx
- Realms Of Darkness Strategic Simulations*
- Saracen* DataSoft
- Award Maker Plus* Baudville
- 221B Baker Street* DataSoft
- Terrapin LOGO Language 3.0* terrapin
- Logo Data Toolkit* Terrapin
- PaintWorks Plus* Activision
- Poker Night* Gerhardt Software
- Portal* Activision
- Shanghai* (Hgs) Activision

## Book Reviewed:

- Apple Hgs Toolbox Revealed* Bantam Books

### Ratings

- ★★★★★..... SUPERB
- ★★★★..... EXCELLENT
- ★★★..... VERY GOOD
- ★★..... GOOD
- ★..... FAIR
- ☹..... POOR
- ☹☹..... BAD
- ☹☹☹..... DEFECTIVE



### Requires:

- 64K Apple II series
- one 5 1/4" drive
- second drive and joystick optional

It was an evil day in the four great lands of Deledain when Lord Denethenor turned his considerable talents to pursuit of the dark arts (swarming minions, devastation, etc.). All of which, to a quick-witted rogue like yourself, presents a golden opportunity. Inspired by a vaguely favorable prophecy ("Not armies but a single brave adventurer may defeat the grim lord's design!"), you begin an epic quest for fame and fortune; one bound to incur the awesome *Wrath of Denethenor*.

Once more into the breach?!-- true enough. Still, this particular zap-the-evil-wizard adventure manages to showcase some new ideas plus delivering more castle, town, dungeon, and countryside real estate than *Ultima IV* and *Questron* combined. (Each of the four continents measures about 100x100!)

With no special combat or dungeon displays, all game action including purchases, battles, and exploration, takes place on one of some thirty-plus scrolling maps.

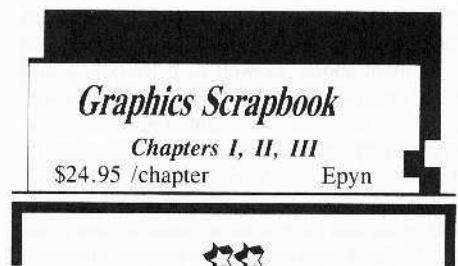
Unlike most adventures, which employ elaborate schemes for advancing a host of attributes, *Denethenor* character development is almost wholly a matter of what you know. Clues and other kinds of information, obtained mainly from individuals you encounter, are essential for planning your explorations, avoiding traps, and acquiring vital magical abilities such as torch-lighting, time-freeze, and invisibility. This applies even to the non-tactical combat, where you must learn which blows (high, middle, or low) are most effective against numerous possible adversaries. Otherwise, your major concerns are obtaining gold, and magical artifacts. Gold buys food which translates into "stamina" and hitpoints; and most spells require using-up a magical item. There are no

great barriers to obtaining a decent weapon; and your few attributes develop adequately without special attention.

You can do a quick 'game save' at almost any time; so death via monster, getting hopelessly lost in a dungeon, etc. are, at worst, very inconvenient-- you simply 'restore' your character. The one real threat to your quest is that-- perhaps upon breaking into a lord's treasure room and being discovered-- you might become involved in a melee and wind up killing a critical information source! (Hint: NEVER save the game after killing a non-monster unless you have heard what he has to say.)

"Denethenor" is possibly the only full-scale map adventure to be entirely player-paced. There are no "experience" points to accumulate or "levels" to achieve before something can happen; nor does the game seem to care how you acquire information. For instance, if a more experienced player simply tells you the name of a spell and how to invoke it, you can! Your character knows what you know and forgets what you forget-- great for 'realism' and 'involvement', and good reason for careful note-taking.

Clever and well-planned, this Sierra On-Line offering is adventuring on a grand scale. The attractive manual is an effective scene-setter, very readable with just enough hard content. Sound is adequate; and, except for the old-style 'multi-crome' text used to show inputs and character status, the display is solid *Ultima III*-class replete with animated monsters and townspeople. If you've been searching for something 'a little different' and alot BIGGER in long-play questing, then, just maybe, it's time you challenged the *Wrath of Denethenor*.



### Requires:

- 64K Apple II series
- Print Shop* or *Print Master*
- one 5 1/4" drive (second drive optional)
- printer

If you count yourself among the throng of regular *Print Shop* or *Print Master* users, then you have just 'struck it rich'! Offering more than a hundred compatible black-and-white graphics per 'chapter', Epyx's *Scrapbook* series represents a printing resources motherload.

"Sports" (Chapter I) spans every popular sport (even chess) with action figures, team symbols, trophies, and assorted paraphernalia. Chapter II, "Off the Wall" is just that, offering a variety of novelty figures. "School", the most recent addition, includes teacher and student figures plus graphics for subject areas, holidays, special events (prom, open house, etc.), and more. Beginning with "School", scrapbooks also supply some extra fonts, borders, and full-panel displays.

For card designs, party decorations, and desktop publishing; whatever you do with *Print Shop* or *Print Master*, now you can do much more with the *Graphics Scrapbook's*.

## Realms Of Darkness

by G. Smith and D. Nghiem  
\$39.95 Strategic Simulations



### Requires:

- 48K Apple II series
- one 5 1/4" drive

Probably, you have wondered what opportunities remain for hard-core questers after the land is saved from bigtime evil enchantress/mad mage types. Well, as things turn out, there are still plenty of upper-grade baddies and deadly missions to insure full employment. In this new SSI fantasy adventure you lead a party of up to eight warriors, sorcerors, clerics, thieves, and others through seven perilous quests into the monster-infested *Realms of Darkness*.

Borrowing freely from the favorites *Shadowkeep* and *Bard's Tale*, *Realms* is a LARGE (three diskette sides), multi-maze adventure. After designing and naming party members you begin in a town which will serve as headquarters for the first four quests. Here you find the requisite inn, tavern, weapons and equipment shops, as well as a friendly guard who functions as a kind of mission dispatcher. In towns, temples, "enchanted gardens", and during travel to dungeons, action is handled via a series of hires frames, picture-text style.

All serious monster bashing, puzzle cracking, and collection of needed artifacts occurs in the ten or so dungeons. These present forward-view, black-and-white line drawings of doors and panels in simulated 3-D. Since graphics execution is virtually instantaneous, movement here is very realistic. Facilitating ease of play, practically all commands (equipping, movement, combat, etc.) are single-key.

However, encountering a significant personage, wall carving, button, etc. produces a message; whereupon you may employ a few simple text commands-- such as "TALK", "EXAMINE", and "PRESS BUTTON". As might be anticipated in a game where text entry is a special purpose add-on, parsing is weak, allowing little deviation from expected inputs. Fortunately, clues abound and required entries are brief.

Monsters appear at random in mixed groups; but only one is actually displayed for the duration of combat. Depending upon abilities, each character may attack, defend, or possibly cast spells. After a few rounds of exchanging blows one side or the other (hopefully the monsters) is exterminated or flees. Victories yield gold, potions, weapons, and other goodies plus level-boosting experience points-- the latter being especially critical for magic users, who acquire new, more powerful spells with each level advance. Most important, winning allows your band of worthies to continue battling the primary foe, the maze itself. Laced with pits, one-way doors, illusionary walls, and teleport traps, several are multi-level; and each is a solid challenge to your maze-mapping savvy.

Supplied with detailed manual, map fragments, and command card, *Realms of Darkness* works best as a two-person enterprise-- one controller and a mapper/navigator. (In fact, flexible party-splitting and switching functions facilitate participation of several players.) Neither artwork nor sound is exceptional; but 'game save' is fast, play is speedy, and the mazes are interesting, challenging, entertaining places to explore. Recovering an enchanted sword, disposing of a cursed globe, polishing-off a rogue robot, and more-- each quest is a fine not-so-mini-adventure good for hours of dungeon delving fun.

## Saracen

by I. Ginsburg  
\$19.95 DataSoft



### Requires:

- 48K Apple II series (128K: boot in 40-column mode)
- one 5 1/4" drive
- joystick optional

For Boulderdashers tired of pushing rocks and anyone else anxious for a new multi-level puzzle arcade, DataSoft has come to the rescue with *Saracen*. According to the boxtop scenario, your goal (as a young Crusader) is to fight, finesse, and otherwise make your way through each "fortress" to locate and zap a Saracen chieftain.

Featuring super-active, scrolling four-screen

displays, smooth animation, and adequate sound, "Saracen" draws upon tried-and-true arcade devices to create an amazing variety of increasingly complex brain benders. In each of one hundred levels you face new arrangements of walls, locked doors, and one-way portals-- along with fatal-to-the-touch bombs, soldiers, and bouncing cannon balls. You use arrows, found scattered about the fortress, for practically everything from shooting bricks out of walls and igniting bombs to zapping soldiers. Keys, almost always well-guarded, must be placed in "holders" to open locked doors; and grenades (activated via arrowshot) are recommended for dispatching chieftains.

The good news is that you can start at any level, there is no time limit, and you get an extra life plus 300 points for potting a chieftain. Bad news? Well, you can carry just one object at a time, an arrow shoots only in the direction it's pointing when picked up, and you start the Crusade with a measly five lives. Also, a High Score is maintained only during a playing session, not saved to disk.

Attractive and very tough, *Saracen* is a bells-and-whistles puzzle feast-- perfect for those times you want a lot of action, but would rather not worry about saving the universe.

## Award Maker Plus

\$39.95 Baudville



### Requires:

- 64K Apple II series
- printer

The first time I saw an *Award Maker Plus* product it was hard to shake off the feeling that maybe I've been wasting my time pursuing a doctoral sheepskin. With this program I could print-up a juicy diploma, slap on a gold seal (20 supplied) and be in business inside of five minutes! (Of course, I would use a sheet of Baudville's fanfold parchment-- a bargain at \$12.95/100.)

Offering four fonts and ten border patterns, this is the package for doing every kind of award, diploma, certificate, license, and title. Among the nearly three hundred available formats you will find numerous school and sports awards, several diploma and certificate styles, novelty awards, and others. Typically, each includes large-format titling, space for name and a few lines of text, and lines for signature(s) and dating. Often, too, there will be a picture, such as the ballet slippers on a dance award. In case you don't see just the right thing (all are shown in the manual), you can select an untitled format. Four styles let you add your own hi-res picture; and *Print Shop* graphics may be lifted to create special borders!

The program supports practically every popular printer and interface, including an option to print borders in color. While you cannot save an award to disk, you can create a "Name List" for producing multiple copies of an award, with a different name on each. Super flexible and easy to use, *Award Maker Plus* makes anyone an expert award giver.



**Requires:**

- 512K Apple IIgs
- RGB monitor
- 3 1/2" drive

*Paintworks Plus* is a super hi-res painting/ drawing/ typing utility that, for starters, lets you prepare full (8.5" x 11") and half sheet documents (320 dots x 200 lines and 320 dots x 396 lines respectively).

Output may be in sixteen levels of gray; but for really spectacular effects an Imagewriter II (or compatible printer) with color ribbon allows printing in up to sixteen colors drawn from a total of 4096.

Text, in six fonts, six styles, and six sizes can be typed-in (and colored!). Plus, you can place any part of your work on a "clipboard" for transfer to materials prepared using Activision's *Writer's Choice Elite*.

Naturally, you can create and save full-screen super hi-res frames for use in programs; and an animation feature permits linking and running sequences at speeds ranging from one to over sixty frames per second.

Wow! Right?! Definitely, and the painting tools are, if anything, even more impressive. A good summary might be: if you can think of something you want to do with, to, or on a graphic, it's probably here.

Accessing the color palette function not only permits adjusting each of the sixteen selected colors for light, dark, and RGB content, but also allows loading-in from/ saving-to a file of 128 sixteen-color palettes.

On-graphic functions include a host of fill, magnification, and editing tools involving use of colors as well as sixteen user-modifiable patterns.

With "cut-and-paste", rectangular patches or figures you "lasso" and lift off the background can be moved, deleted, or copied; and an invisible-grid option aids in precise line and figure placement.

Such features, plus flips, rotation, and even a nifty 1-4 axis "mirror draw" (to help preserve symmetry) all greatly simply otherwise imposing tasks.

Supplied with a handsome, example-filled,

147-page manual and boasting exceptionally smooth, responsive mouse action, this is a package designed for quick, comfortable mastery.

In several enjoyable hours of working with *Paintworks Plus* I've come upon only two identifiable flaws and one "toss-up".

As to the former: the absence of a file "DELETE" command tends to promote "work disk" clutter; and there is a real dearth of valid information regarding picture storage formats. At one point the manual observes that saving the full two-screen canvas (in the 320 x 396 "paint" format) uses much more disk space than the single-screen "screen" format. Actually, paint files are compressed, frequently taking up less than half the 65 blocks (32K bytes) required by screen files. "Paint" is also the only format which saves the current pattern palette.

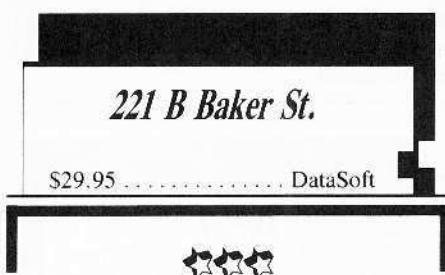
Activision provides no documentation as to how one might lift-out and implement paint decompression routines in user programs. ( Too bad; however, individual program files are open to access and experimentation.) Of course, the real advantage of screen files is that, since they follow the standard IIgs SHIRES arrangement, screen files may easily be employed in a variety of programming applications.

The "toss-up" relates to the fact that tool icons, menu bar, and the two palettes (the "easel frame") occupy about one-third of the screen. Since drawing functions are active only when the frame is in place, you will have to do more moving around of the work area than with some competing products.

On the other hand, all of your tools and palette elements are easily distinguished and, more important, large enough for quick, error-free "clicking".

Perhaps because it is an excellent package, *Paintworks Plus* is bound to stimulate criticism-- mainly along lines of ideas for even bigger, more comprehensive systems.

Make no mistake, this is a graphics powerhouse: Christmas card maker, slideshow generator, leaflet publisher, games graphics medium, artist's canvas, and more. Versatile, yet both easy and fun to work with, *Paintworks Plus* is a IIgs user definite must!



**Requires:**

- 64K Apple II series
- joystick optional

Set in Holmesian London, *221B Baker Street* is a mystery-busting parlor game for one to four players. In each of thirty crime scenarios your

aim is to be first to crack the case by answering the questions presented, along with the scenario introduction, in the "Case Booklet". Usually, this involves naming a murderer, method, and motive; but, reflecting a nice mix of evil doings, you may be asked to prevent a murder, nab a thief, crack a coded message, or even locate a kidnapped songbird!

Moving their tokens around on the scrolling gameboard, players obtain clues at the docks, a tabacconist's shoppe, the theatre, and several other key locations. Rolling an on-screen die determines how many squares you cover on a turn-- up to six, unless you try a "secret tunnel" shortcut.

You obtain just one clue per location. Since these are displayed (for anyone else looking to see), an option allows clues to be hidden via simple alphabet-substitution codes. If selected, each player is assigned his/her own decoder list; the computer then presents your clues in your code.

Unlike earlier mystery favorites, *Murder by the Dozen* and *Felony*, the DataSoft offering emphasizes inter-player competition-- you can actively obstruct the progress of fellow sleuths! Badges, obtained at Scotland Yard, allow locking particularly juicy locations to other players-- requiring keys, obtained from the locksmith, to open. Only one of each may be carried at any time; AND you will need a badge when you move to 221B Baker Street to enter your solution. If you are 100% right, you win; but any error means a race back to Scotland Yard for another badge.

Well thought-out and smooth-running, the game comes with manual and "Case Book", both replete with pithy Holmes quotes. Whether you play a solitaire game or choose to take on adversaries, each *221B Baker St.* scenario is good for about an hour of challenging, puzzle-cracking fun.



**Requires:**

- Terrapin Logo 2.0- 3.0
- 64K Apple II series

Terrapin's *Logo Data Toolkit* is a collection of programs aimed at encouraging the development of interesting, useful applications with a special emphasis on creating and accessing data bases. Included in the package are the diskette and a comprehensive 83-page manual (approx. fifth grade reading level) illustrating numerous uses for each tool. Explicit menus and clear explanations mean that, in no time at all, practically any reader can begin to develop survey questionnaires, do

line, bar, and pie charts, create and edit custom data bases, and experiment with new uses for "Sorting" and "Associations" procedures. Both 64K and 128K-ready versions of all files are supplied-- the main advantage of the latter being less frequent disk access.

Though the book is a tad sterile (charts, etc. but no pictures), the package is a genuine bonanza of nifty project ideas. Entirely accessible with or without teacher guidance, the *Logo Data Toolkit* is a fine introduction to some of the language's more powerful features.



**Requires:**

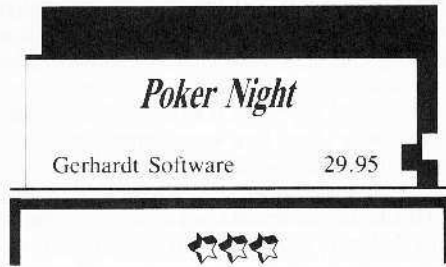
- 64K Apple II series

*Logo version 3.0* is a DOS 3.3 based implementation of Logo which will run on 64K Apples but, on larger machines, makes use of up to 128K of memory. Since the language normally carves out a hefty chunk of the 64K in bank 0, doubling the accessed memory space means owners of newer Apples can write considerably larger, serious application, *Terrapin Logo* programs. Of course, the many utilities and teaching aids developed for earlier Logo's will run under the new version.

In the *Logo Language 3.0* package you will find the "Language" and "Utilities" diskettes, a manual, and a Logo "commands" "Quick Reference" card.

Well-organized and thoroughly indexed, the manual describes each command (i.e. each "primitive" procedure) in detail and includes a large appendix of handy procedure listings, a technical reference, and a built-in tutorial. In fact, starting from scratch, you can begin writing simple Logo programs almost immediately. The "Utilities" diskette is packed with useful procedures, some example programs, an assembler (for writing Logo-callable machine code), and even an "Instant Logo" to get young non-readers started. Evidently, Terrapin protects only its language diskette; so program and utilities diskettes-- the ones most likely to get zapped-- are easily backed up.

Though weak in number-crunching, Logo is a powerful language especially well-adapted to handling lists and general word processing. *Terrapin's Logo 3.0* is a very good implementation; one which, in advanced applications, is hardly the province of school children. As always, a "quick-starter" when it comes to graphics programming, '3.0 retains the responsive, friendly interface which makes this language one of the few it is actually fun to use.



**Requires:**

- 64K Apple II

When I first booted Gerhardt's *Poker Night* it was with a fair degree of scepticism. How could anyone expect to get that special "poker feel" and excitement playing a computer? (Blackjack- OK; after all, the dealers in Vegas even act like computers.)

That *Poker Night* is largely successful is the result of amazing flexibility and variety. Up to seven human players may participate, with any empty chairs open for computer players.

These include 'Ace', 'Dolly', 'Slim', and thirteen others, each with his or her own style and skill level. You decide how many and who will join in. (And, if you're looking for easy pickings, you will definitely want to remember who's who!)

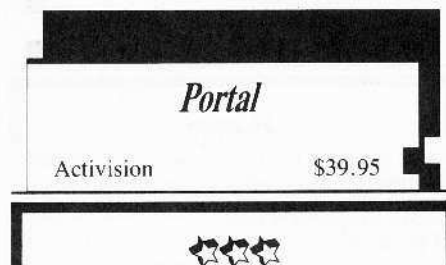
"House Rules" such as maximum bet, ante, discards allowed, etc. can be tailored; so there is no need to adjust to some "foreign" setup.

Altogether, *Poker Night* offers twenty variants: seven draw, ten stud, and three hold 'em games.

Since the 'deal' (i.e. the right to name the game) rotates, if you are playing three or four computer players, you can expect to see everything from "five-card draw high/low" and "High Chicago" to "Texas Hold 'em".

Your Apple takes care of all the dealing, current winnings/losses display for each player, and will even supply a probability analysis in stud games.

Smooth-running, attractive, and fast-executing, the game comes remarkably close to the real thing. I started my first session with the idea of "wrapping it up" after getting a few hundred bucks ahead, and soon found myself fighting to break even! Perfect when your regular group isn't available (or maybe even if it is), *Poker Night* is a fine evening's entertainment.



**Requires:**

- 64K Apple II series
- joystick optional

It is the year 2106 when, emerging from forty years of cryosleep, you find yourself approaching (of all places) Earth. A quick check of the on-board computer (OBC) reveals little more than a decision (from where?) to abort your scheduled 100-year mission of exploration to 61 Cygni; otherwise, OBC memory has been wiped nearly clean.

(How did that happen?) The real shocker, however, is Earth. What good, after all, is a homecoming when *nobody's home*?

*Portal* thrusts you into a 22nd century world of abandoned underground cities and space colonies. Evidently, mankind has simply left the solar system. But Why? How? And where did everyone go?

Fortunately, an extensive communications/databank network called "Worldnet" remains partially operational; and, better still, an advanced machine intelligence named "Homer" seems as desperate as yourself to solve the biggest mystery ever.

Homer, alas, is but one of the net's twelve major data spaces-- hardly the system's master. A 'simple' cataloging/story-telling algorithm on the brink of eternal somnolence, he is heavily dependent upon the information you retrieve for stimulation of long dormant memories.

Thus "Worldnet", gradually awakening in response to renewed usage, becomes your universe.

Here you move among "Homer", "Scitech", "History", and the other dataspaces. Sometimes you find nothing, sometimes a dry recitation of facts, elaborate personal profiles, or even prosaic discourses.

All too soon you are swamped in a tidal wave of obtusely related information.

Mutagens?

The portal?

Migration!...

And why does Homer seem convinced that a boy, Peter Devore, is the key?!

This is not a simple puzzle.

As, perhaps, the first true "electronic novel", *Portal* allows you to control pacing and, to some extent, the order of presentations. The fun, aside from enjoying a good story, is in putting the pieces together and anticipating new developments.

One hint: beware of any tendency towards copious note-taking. I've tried it; and soon had several sheets covered before getting beyond even the first of six diskette sides! Sketchy notes are fine, since (except for Homer's musings) all significant presentations up to the time of your last "data save" are retained. These are titled and available for perusal whenever you wish.

Supplied with just enough documentation to whet your interest and get you started, this is an attractive and-- except for one glitch--smooth-running package.

(The Glitch: When restarting a "saved game" you must select "New" instead of "Continue" after logging on.)

So far I have enjoyed the story. Combining hi-tech medium and futuristic message, *Portal* certainly qualifies as a "classic" entertainment experience.

## Shanghai

Activision

\$44.95



### Requires:

- Apple IIgs
- 3½" drive
- a Mouse

When *Mah Jongg* came to America in the Twenties the game proved so addictive that, supposedly, it was banned in Philadelphia.

Unwilling to let sleeping dragons lay, Activision has reintroduced the 19th century Chinese favorite in the form of *Shanghai*.

As is often true of good games, *Shanghai's* rules are very simple. You begin with an arrangement (called a "dragon") of 144 tiles stacked up to five deep. The object is to remove all the tiles (slay the dragon) by eliminating matched pairs. Only tiles which are not covered and which are unblocked on a horizontal side are available.

With just four of each tile, the trick is to plan your matches so that future pairings are promoted.

The first thing you will notice upon booting *Shanghai* is that, in multi-color super hi-res, the 'flowers', 'seasons', and other elegantly marked tiles produce one of the prettiest displays of any game.

The second is that, once you start playing, it's hard to stop! Besides a super addictive solitaire version, you can participate in timed one-on-one "challenges" and multi-player "tournaments". In each, your score is the number of tiles you remove. Tournament scores (with names) are saved to disk.

Silky-smooth mouse action, realistic "3-D" perspective, quick setup, and immediate move validation make play via computer decidedly preferable to using real tiles.

"Take-back" and "Show All Moves" options even help you refine your technique.

A weakness is that only one tournament at a time is maintained on disk; and the manual is ambivalent in some key areas (such as whether or not a win is possible for every dragon generated).

Rules, however, are thoroughly explained--both in the manual and program "Help" screens.

Programmed via mouthstick, since Mr. Lockard is paralyzed from the neck down, this *Mah Jongg* adaptation is both a software gem and a nice lesson in what chessmasters mean by the "strength" of a "weak" move.

As absorbing as it is subtle, *Shanghai* is great, mind-stretching entertainment for players and kibitzers alike.

## The Apple IIgs Toolbox Revealed

by Danny Goodman  
Bantam Books \$21.95



Inevitably, when someone like Apple introduces a major new, technically complex product, a host of books from various vendors flood the market. (Especially when the someone, like Apple, sits on its collective hands when it comes to delivering useful documentation.) Some of these texts are very good; but too many are not worth the air they displace. Bantam's *The Apple IIgs Toolbox Revealed* belongs solidly in the air ware category.

Beginning with Chapter One the toolbox info-hungry purchaser is greeted with "Under the Hood", a unit nicely characterized by such subtitles as "RAM: the Fuel Tank" and a smiling CPU chip. Here you learn that your IIgs does indeed have RAM and ROM. Moving up through Chapter Three gets you to "Talking to Your IIgs" (not an unlikely result if you spend enough time with this text). Here you discover that today's computers are (really) a lot better than yesterday's computers, that machine code is hard to write, and that interpreters have disadvantages.

Finally, 59 pages into the book, Goodman asks: "What's a Toolbox?" Fair enough. Now, perhaps a reader who knows what's in the IIgs toolbox (and how it works) will write the author. Eliminate the half-way decent chapter on Quickdraw II, and you have a book absolutely free of hard content, let alone anything approaching specifics on the IIgs toolbox. Distinctly reminiscent of Scientific American's famous April 1st issue some years back, *The Apple IIgs Toolbox Revealed* is good Apple group fun-- unless you're the one who bought it.

## UPDATES

Two popular games, Activision's *Hacker II* and Electronic Arts's *Bard's Tale I* have been re-released in IIgs format. *Hacker II's* new graphics and sound effects are, in a word, spectacular. Breaking into a top-secret Russky installation has never been more fun. *Bard's Tale* approaches 'work of art' status with beautiful illustrations, smooth animation, masterful sound, and envelopingly real dungeonscape.

## Vendor Addresses

**Activision**  
P.O. Box 7287  
Mountainview, CA 94039  
(800)227-9759  
in CA:(415)940-6044

**Bantam Books**  
666 Fifth Avenue  
New York, NY 10103  
(212)765-6500

**Baudville**  
1001 Medical Park Dr.  
Grand Rapids, MI 49506  
(616) 957-3036

**Epyx**  
1043 Kiel Ct.  
Sunnyvale, CA 94089

**Gerhardt Software**  
32600 Concord Drive  
Dept. 401  
Madison Heights, MI 48071

**Intellicreations/Datasoft**  
19808 Nordhoff Place  
Chatsworth, CA 91311  
(818) 886-5922

**McGray-Hill**  
2600 Tenth Street  
Berkeley, CA 94710  
(415) 548-2805

**Omnitrend**  
POB 3  
West Simsbury, CT 06092  
(203) 658-6917

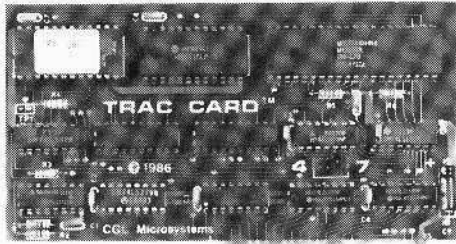
**Sierra On-line**  
P.O. Box 485  
Coarsegold, CA 93614

**Strategic Simulations**  
1046 N. Rengstorff Ave.  
Mountain View, CA 94043

**Terrapin, Inc.**  
222 Third Street  
Cambridge, MA 02142  
(617) 492-8816



## TRAC CARD



### Boot Process Memory Card

- +On-Board Memory Stores Up To 200 Disks Of Accessed Tracks While Power Up
- +All Disks Are Automatically Monitored From The Moment You Power Up. The Tracks Are Divided Into Groups Of "Booted" Disks
- +Save Time When Using Backup Software-The Tracks Accessed May Be Displayed In Numerical Order Or In The Order In Which They Are Read
- +TRAC CARD Gives You Maximum Accuracy For Backing Up Software By Precisely Storing 1/4, 1/2 and 3/4 Tracks, As Well As Full Tracks
- +You May Choose 40 or 80 Column On Monitor Or Dump Data To Printer. Name Each Disk When Printing Track List
- +Choose Either Decimal Or Hexadecimal Readout
- +Use In Any Slot, Including Slot #3 On //e
- +Works With Any Apple Compatible 5 1/4" Drive
- +Works With Apple II, II+ and //e, As Well As Compatibles

Price \$159.95 Plus \$3.00 Shipping & Handling

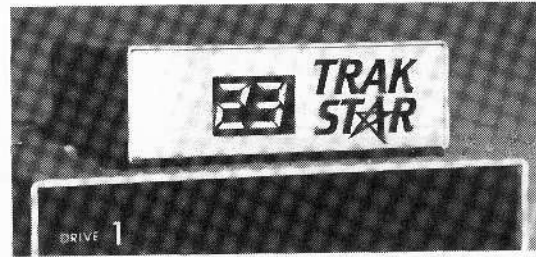
Personal checks, M.O.,  
Visa and Mastercard  
Phone 913 676-7242

Apple is a registered trademark of Apple Computer Inc.

**Midwest**  **Microsystems**

10308 Metcalf, Suite 355  
Overland Park, KS. 66212

## TRAK STAR



### Constant Digital Readout of Disk Drive Head Position

- +Works With Any 5 1/4" Apple Compatible Drive
- +Saves Copying Time With Nibble Programs
- +Copy Only Tracks That Are Displayed
- +If Copied Program Doesn't Run, TRAK STAR Displays Track To Be Recopied
- +Displays Full and Half Tracks
- +Operates With Any Apple Compatible Program, Including Protected Software
- +Displays Up To 99 Tracks and Half Tracks; Compatible With High Density Drives
- +Does Not Use A Slot in the Apple
- +For Apple II, II+ and //e
- +Simple One Minute Installation

Price \$99.95 Plus \$3.00 Shipping & Handling  
Adaptor Cable Required For 2 Drive System \$12.00  
DuoDdisk, 5 1/4" Unidisk and IIc Owners Please Write

### APPLE COMPATIBLES LOWEST PRICES ANYWHERE!

IMEG/80 COL BD w/256K+Software (IIe) .....	\$99
IMEG RAM BD for IIGS w/256K .....	\$89
Above w/512K add \$39 1 MEG add \$99.	
64K/80 COL BD. New Lower Power (IIe) .....	\$35
16K RAM Board (II+) .....	\$35
128K RAM BD. New Lower Power (II+) .....	\$69
80 Column Board. Videx Comp. (II+) .....	\$49
Super Serial Board (II+/e) .....	\$49
Graphic Par BD w/6FT CBL (II+/e/gs) .....	\$45
Z80 CP/M BD Microsoft Comp. (+/e/gs) .....	\$38
Cooling Fan w/surge protect (II+/e) .....	\$29
GS Super Cooling Fan (IIgs) .....	\$25
Numeric Keypad, 16 Keys (IIe) .....	\$35
Joystick (Specify II+/c/e/gs) .....	\$15
Joystick w/Fire on stick (+/c/e/gs) .....	\$25
Mini Vacuum Cleaner w/ Attachments .....	\$10
A/B Switchbox. Parallel or RS-232 .....	\$29
Disk Drive H/H (Specify II+/c/e/gs) .....	\$129
Disk Controller Board (II+/e/gs) .....	\$39
Eprom Programmer (II+/e/gs) .....	\$49

**ONE YEAR WARRANTY ON ALL PRODUCTS  
CALL/WRITE FOR COMPLETE LIST  
ADD \$3 SHIPPING (Per ORDER, not per item)**

**NEXO DISTRIBUTION**  
914 East 8th Street, Suite 109  
National City, CA 92050  
(619) 474-3328 10am-6pm Mon-Fri

**UNIV & SCHOOL P.O.'s WELCOME!  
VISA/MC OKAY—C.O.D. ADD \$2.00**

## A Hackers Challenge

awaits those who enjoy "breaking" protection schemes for fun.

### Knowledgeable with DOS 3.3 and associated utilities?

If so, this game is for you. Includes graphics routines from **The Graphics Magician®** by **Polarware™**.

Inside, protected by layers of graphical puzzles and DOS-tricks *hides a secret*. Being among the 1st ten to discover the *game secret* wins you a computer-game prize! What are these prizes? I'm not saying, yet all the *clues* you need to win along with the *prize-list* are concealed in the game.

*Find the clues by playing the game.*

*Find the clues by using your favorite utilities.*

To answer this challenge send \$10.00 to:

**Mark V. Whitehurst**  
PO Box 485  
Franklin Park, IL 60131

Requires APPLE IIe or IIc.



# Holiday Coupon Specials

\$5

**SAVE**

\$5

## COMPUTIST Back Issues

When you order 5 or MORE, you get \$1 off each back issue. Order 10 and save \$10. Order 15, save \$15...

Include this coupon (or a back-up copy) with an order for 5 or more back issues from the order form on page 46 and you can save \$1 on each issue. Expires January 31, 1988.

\$3

**SAVE**

\$3

## COMPUTIST Library Disks

When you order 2 or MORE, you get \$1.50 off each disk. Order 4 and save \$6. Order 6, save \$9...

Include this coupon (or back-up copy) with an order of 2 or more COMPUTIST library disks from the order form on page 46 and you can save \$1.50 on each library disk. Expires January 31, 1988.

\$5

**SAVE**

\$5

## Book Of Softkeys

When you order any volume of the Book Of Softkeys.

Include this coupon (or backup copy) with an order for any Book Of Softkeys (use order form on page 44) and save \$5 per copy. NOT valid for the Complete Collection. Expires January 31, 1988.

\$7

**SAVE**

\$7

## Diskbusters T-shirt

Super year-end holiday clearance sale! Diskbusters T-shirts for only \$2.99 each. Limited quantities.

These T-shirts are from the hacking hey-days of the old Hardcore Computing era. Include this coupon (or backup copy) with the T-shirt order form on page 3. Expires January 31, 1988.

**FREE**

**5**

**FREE**

## Color Disk Cases

Each bulk-disk order (page 44) is packed in five free color (our choice) disk-cases. Limited quantities.

U.S.A. orders ONLY. Include this coupon (or backup copy) with the order form on page 44 and get 5 free color-coded diskette cases. Clearance sale; we choose the color. Expires January 31, 1988.

**IF**  
You're a  
**COMPUTIST**  
reader

**THEN**  
You can  
really

**SAVE**  
lots of  
hard-earned

**\$\$\$**  
when you  
use these  
**Holiday**  
coupons.

Just include these coupons with your regular orders (see appropriate order form) in this issue and deduct the appropriate savings. Limited quantities of most items. These coupons are good til January 31, 1988 so...

**HURRY**

# COMPUTIST

**48** *Features:* Dungeon Editor & Encounter Editor for Ultima III • APT for Shadowkeep • Softkey for Shadowkeep • Softkey for Apple Business Graphics • *RDEX* *Softkeys:* • 816 Paint GS • Addition Logician • Amnesia • Arctic Fox • Award Maker Plus • Bard's Tale II • Betterworking Word Processor • Beyond Castle Wolfenstein • Black Magic • Bookends Extended • Bop & Wrestle • Chess 7.0 • Chessmaster 2000 • Counting Critters • Deluxe Paint GS • Destroyer • Hacker II • Hacker II GS • Hardball • Infiltrator • Instant Music GS • J-Bird • Mabel's Mansion • Marble Madness • Math Critters • Mean 18 GS Golf • Megabots • Might & Magic • Miner 2049er II • Mouse Word • Music Construction Set GS • Music Studio GS • New Oregon Trail • Paintworks Plus 1.0 GS • Paintworks Plus 1.01 GS • Paul Whitehead Teaches Chess • PHM Pegasus • Poetry Express • Print Shop color version • Quotient Quest • Rambo: First Blood part II • Rocky Horror Show • Sargon III\* • Shanghai GS • Spindizzy • TelePorter • Temple Of Apshai trilogy • Top Draw GS • Transylvania • Ultima I • World's Greatest Baseball Game • Writing A Character Sketch • Writing A Narrative . . . . .

**47** *Features:* Infocom-text Reader Enhancement • Color Ultimapper mod to Ultimapper IV • Towne Mapper utility for Ultima IV • Dungeon Mapper utility for Bard's Tale • Hardware Corner: Interrupting Your Apple • Softkey for Charlie Brown's 1,2,3s • *RDEX* *Softkeys:* • Guitar Wizard • Gemstone Warrior • Notable Phantom • Micro Wine Companion • Stickybear Printer • Note Card Maker • Starcross • Wishbringer • Dinosaur Dig • Dam Busters • Pirate Adventure • Infiltrator • MECC software • Banner Catch • Turtle Tracks • PFS File • Microzine #12, #13, #14 • Marble Madness • Writer Rabbit • Arcticfox • Age Of Adventure • Might And Magic • Space Station • Alternate Reality • Mindshadow • Gemstone Warrior • Strip Poker • Lucifer's Realm • Manuscript Manager • Bank Street Writer III • Kids On Keys • The Missing Ring • Graphic Solution • Empire I, II • Championship Golf . . . . .

**46** *Softkeys* • Advanced Microsystems Technology programs • Word Attack • Star Blazer • Science Toolkit • The Color Enhanced Print Shop • Video Vegas • The Handlers • K.C. Deals On Wheels • Law Of The West • Break The Bank Blackjack • Foundation Course In Spanish • OGRE • Puzzles And Posters • *Features* • The Shift Key/Lower Case Option For II+ • Amazing Computer Facts • Shape Magic utility • *review:* Multiscribe . . . . .

**45** *Softkeys* • Mouse Calc • Sands of Egypt • Number Farm • Agent U.S.A. • Wavy Navy • Kindercomp • Flight Simulator Update • Raid over Moscow • Crime Stopper • Key Perfect 5. • The Final Conflict • Miss Mouse • Snoggle • *Features* • Write Protecting the Microsoft RAM Card • Keys to Success on the Franklin Ace • Modified F8 ROMs on the Apple III • *Core* • Owner's Review of Copy Master II . . . . .

**44** *Softkeys* • Arcade Boot Camp • Goonies • Zorro • Coveted Mirror • Crimson Crown • Compubridge • Fleet System 3 • Microwave • Escape • Catalyst 3.0 • Number Farm • Alphabet Circus • Joe Theisman's Pro Football • Black Cauldron • Intern. Gran Prix • *Features* • Making DOSless Utilities • Pixit Printer Drivers • *Review:* Z-RAM Memory Expansion Board • Reading the Joystick . . . . .

**43** *Softkeys* • Graphics Expander • Information Master • Certificate Maker • Elite • Catalyst 2.0 and 3.0 • Murder On The Mississippi • Temple Of Apshai Trilogy • Troll Associates programs • Spell It • Regatta • Cdex Training programs • Think Fast • *Features* • How to Write-Protect your Slot Zero • Capturing Locksmith 6.0 Fast Copy • Revisiting DOS to ProDOS and Back • *Core* • Computer Eyes / 2: a Review • APTs • Sword of Kadash & Rescue Raiders • Ultimaker IV . . . . .

**42** *Softkeys* • Light Simulator • Beach-Head • Monty Plays Scrabble • Racter • Winnie the Pooh • Infocom Stuff, Kabul Spy, Prisoner II • Wizardry I & 2 • Lucifer's Realm • The PFS Series • Dollars and Sense • Strip Poker • Coveted Mirror • Wizard's Crown • The Swordthrust Series • Axis Assassin • Manuscript Manager • The Crown of Arthain • Address Book • Decimals 3.0 • Dragonfire • *Features* • Auto Duel Editor • Wizard's Crown Editor • Questron Mapper • *Core* • The Games of 1986 in Review • *Adventure Tips* • Ultima IV . . . . .

**41** *Softkeys* • The Periodic Table • Gemstone Warrior • Inferno • Frogger • Story Maker • Adventure Writer • Mummy's Curse • Zaxxon • The Quest • Pitfall II • H.E.R.O. • *Features* • A Two-Drive Patch for Winter Games • Customizing the Speed of a Duodisk • Roll the Presses Part Two: Printshop Printer Drivers • The Games of 1986 . . . . .

**40** *Softkeys* • Adventure Writer • E-Z Learner • Mychess II • Raster Blaster • Cranston Manor • Ghostbusters • Designer's Pencil • The American Challenge • Encyclopedia Britannica Programs • Crime Wave *Features* • Taking the Wiz out of Wizardry • Adding a Printer Card Driver to Newsroom • *Core* Games of 1986 . . . . .

**39** *Softkeys* • MIDI/8+ • Homeword v2.1 • Borrowed Time • Amazon • Speed Reader II • Discovery! • M-ss-ng L-nks series • Donald Ducks's Playground • Mastering the SAT • Copy II Plus 4.4C • Master of the Lamps • One on One • Bridge Baron • A.E. • Great American Cross-Country Road Race • Computer Preparation for the SAT • Castle Wolfenstein • Luscher Profile • Skyfox • Silent Service • Echo Plus • Swashbuckler • Randamn • *Features* • Electronic Disk Drive Swapper • Abusing the Epilogues • Print Shop Companion's Driver Game • *Core* • Keyboard Repair • Fixing the Applesoft Sample Disk . . . . .

**38** *Softkeys* • Cyclod • Alternate Realty • Boulder Dash I & II • Hard Hat Mack (Revisited) • The Other Side • F-15 Strike Eagle • Championship Lode Runner • Gato V 1.3 • I, Damiano • Wilderness • Golf's Best • *Features* • The Enhanced/ Unenhanced II • Looking into Flight Simulator's DOS • *Core* • Appavarex • Installing a RAM disk into DOS 3.3 . . . . .

**37** *Softkeys* • Under Fire • Pegasus II • Take I (revisited) • Flight Simulator II v1.05 (part 2) • Magic Slate • Alter Ego • Rendezvous • Quicken • Story Tree • Assembly Language Tutor • Avalon Hill games • Dark Crystal • *Features* • Playing Karateka on a IIc • Track Finder • Syk to Dif • *Core* • Breaking In: tips for beginners • Copy II Plus 6.0: a review • The DOS Alterer . . . . .

**36** *Softkeys* • Flight Simulator II v 1.05 • AutoDuel • Critical Reading • Troll's Tale • Robot War • General Manager • Plasmania • Telarium Software • Kidwriter v1.0 • Color Me • *Features* • ScreenWriter meets Flashcard • The Bus Monitor • Mousepaint for non-Apples • *Core* • The Bard's Dressing Room • APT • Championship Lode Runner . . . . .

**35** *Softkeys* • Olympic Decathlon • Hi-res Cribbage • Revisiting F-15 Strike Eagle • Masquerade • The Hobbit • Pooyan • The Perfect Score • Alice in Wonderland • The Money Manager • Good Thinking • Rescue Raiders • *Feature:* Putting a New F8 on Your Language Card • *Core* • Exploring ProDOS by installing a CPS Clock Driver . . . . .

# Available Back Issues

**34** *Softkeys* • Crisis Mountain • Terripin Logo • Apple Logo II • Fishies 1.0 • SpellWorks • Gumball • Rescue at Rigel • Crazy Mazey • Conan • Perry Mason: The Case of the Mandarin Murder • Koronis Rift • *Feature* • More ROM Running • *Core* • Infocom Revealed.....

**33** *Softkeys* • Word Juggler • Tink! Tonk! • Sundog v2.0 • G.I. Joe & Lucas Film's Eidolon • Summer Games II • Thief • Instant Pascal • World's Greatest Football Game • Graphic Adventure #1 • Sensible Grammar & Extended Bookends • Chipwits • Hardball • King's Quest II • The World's Greatest Baseball Game • *Feature* • How to be the Sound Master • *Core* • The Mapping of Ultima IV.....

**32** *Softkeys* • Revisiting Music Construction Set • Cubit • Baudville Software • Hartley Software • Bridge • Early Games for Young Children • Tawala's Last Redoubt • Print Shop Companion • Cracking Vol II • Moebius • Mouse Budget, Mouse Word & Mouse Desk • Adventure Construction Set • *Feature* • Using Data Disks With Microzines • *Core* • Super IOB v1.5 a Reprint.....

**31** *Softkeys* • Trivia Fever • The Original Boston Computer Diet • Lifesaver • Synergistic Software • Blazing Paddles • Zardax • Time Zone • Tycoon • Earthly Delights • Jingle Disk • Crystal Caverns • Karate Champ • *Feature* • A Little Help With The Bard's Tale • *Core* • Black Box • Unrestricted Ampersand.....

**30** *Softkeys* • Millionaire • SSI's RDOS • Fantavision • Spy vs. Spy • Dragonworld • King's Quest • Mastering the SAT • Easy as ABC • Space Shuttle • The Factory • Visidex 1.1E • Sherlock Holmes • The Bards Tale • *Feature* • Increasing Your Disk Capacity • *Core* • Ultimaker IV, an Ultima IV Character Editor.....

**29** *Softkeys* • Threshold • Checkers v2.1 • Microtype • Gen. & Organic Chemistry Series • Uptown Trivia • Murder by the Dozen • Windham's Classics • Batter Up • Evelyn Wood's Dynamic Reader • Jenny of the Prairie • Learn About Sounds in Reading • Winter Games • *Feature* • Customizing the Monitor by Adding 65C02 Disassembly • *Core* • The Animator.....

**28** *Softkeys* • Ultima IV • Robot Odyssey • Rendezvous • Word Attack & Classmate • Three from Mindscape • Alphabetic Keyboarding • Hacker • Disk Director • Lode Runner • MIDI/4 • Algebra Series • Time is Money • Pitstop II • Apventure to Atlantis • *Feature* • Capturing the Hidden Archon Editor • *Core* • Fingerprint Plus: A Review • Beneath Beyond Castle Wolfenstein (part 2).....

**27** *Softkeys* • Microzines 1-5 • Microzines 7-9 | Microzines (alternate method) • Phi Beta Filer • Sword of Kadash • Another Miner 2049er • Learning With Fuzzywomp • Bookends • Apple Logo II • Murder on the Zinderneuf • *Features* • Daleks: Exploring Artificial Intelligence • Making 32K or 16K Slave Disks • *Core* • The Games of 1985: part II.....

**26** *Softkeys* • Cannonball Blitz • Instant Recall • Gessler Spanish Software • More Stickybears • Financial Cookbook • Super Zaxxon • Wizardry • Preschool Fun • Holy Grail • Inca • 128K Zaxxon • *Feature* • ProEdit • *Core* • Games of 1985 part I.....

**25** *Softkeys* • DB Master 4.2 • Business Writer • Barron's Computer SAT • Take I • Bank Street Speller • Where In The World Is Carmen Sandiego • Bank Street Writer 128K • Word Challenge • Spy's Demise • Mind Prober • BC's Quest For Tires • Early Games • Homeword Speller • *Feature* • Adding IF THEN ELSE To Applesoft • *Core* • DOS To ProDOS And Back.....

**24** *Softkeys* • Electronic Arts software • Grolier software • Xyphus • F-15 Strike Eagle • Injured Engine • Mr. Robot And His Robot Factory • Applecillin II • Alphabet Zoo • Fathoms 40 • Story Maker • Early Games Matchmaker • Robots Of Dawn • *Feature* • Essential Data Duplicator copy parms • *Core* • DOS-Direct Sector Access.....

**23** *Softkeys* • Choplifter • Mufplot • Flashcalc • Karateka • Newsroom • E-Z Draw • Gato • Dino Eggs • Pinball Construction Set • TAC • The Print Shop: Graphics Library • Death In The Caribbean • *Features* • Using A.R.D. To Softkey Mars Cars • How To Be The Writemaster • *Core* • Wheel Of Money.....

**22** *Softkeys* • Miner 2049er • Lode Runner • A2-PBI Pinball • The Heist • Old Ironsides • Grandma's House • In Search of the Most Amazing Thing • Morloc's Tower • Marauder • Sargon III • *Features* • Customized Drive Speed Control • Super IOB version 1.5 • *Core* • The Macro System.....

**20** *Softkeys* • Sargon III • Wizardry: Proving Grounds of the Mad Overlord and Knight of Diamonds • The Report Card VI.1 • Kidwriter • *Feature* • Apple II Boot ROM Disassembly • *Core* • The Graphic Grabber v3.0 • Copy II+ 5.0: A Review • The Know-Drive: A Hardware Evaluation • An Improved BASIC/Binary Combo.....

**19** *Softkeys* • Rendezvous With Rama • Peachtree's Back To Basics Accounting System • HSD Statistics Series • Arithmetickle • Arithmekicks and Early Games for Children • *Features* • Double Your ROM Space • Towards a Better F8 ROM • The Nibbler: A Utility Program to Examine Raw Nibbles From Disk • *Core* • The Games of 1984: In Review-part II.....

**16** *Softkeys* • Sensible Speller for ProDOS • Sideways • Rescue Raiders • Sheila • Basic Building Blocks • Artsci Programs • Crossfire • *Feature* • Secret Weapon: RAMcard • *Core* • The Controller Writer • A Fix For The Beyond Castle Wolfenstein Softkey • The Lone Catalog Arranger Part I.....

**1** *Softkeys* • Data Reporter • Multiplan • Zork • *Features* • PARMS for Copy II Plus • No More Bugs • APT's for Choplifter & Cannonball Blitz • 'Copycard' Reviews • Replay • Crackshot • Snapshot • Wildcard.....

*Use the handy  
Order Form  
on page 46.*

only **50¢** per disk

When you buy 50 disks  
for only \$25.

- Double-density double-sided 5 1/4" diskettes with:
- Sold ONLY in sets of 50 diskettes
- 100% guaranteed
- Includes:
  - Tyvek sleeves
  - Write-protect tabs
  - reinforced hub rings

Please send me \_\_\_\_\_ sets (50 disks per set) of 5 1/4" DS/DD diskettes. I have enclosed \$25 (shipping/handling included) for each set.

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

Address \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_

Country \_\_\_\_\_ Phone \_\_\_\_\_

  \_\_\_\_\_ Exp. \_\_\_\_\_

Signature \_\_\_\_\_ CP49

Domestic orders, please include street address for UPS delivery.

Washington state orders add 7.8% sales tax.

Offer good while supplies last. Expires January 31, 1987.

Foreign orders: Please inquire for appropriate shipping/handling.

Legends tell of the days when the ancient back issues of Hardcore COMPUTIST were readily available to anyone who wished to purchase them. Those days may be long past, but the treasures with those ancient documents has been diligently transcribed to the pages of a modern reference work:

# The Book(s) of Softkeys

## Volume I

Compiled from issues 1-5

contains softkeys for:

- Akalabeth • Ampermagic • Apple Galaxian
  - Aztec • Bag of Tricks • Budge's Trilogy
  - Buzzard Bait • Cannonball Blitz • Casino
  - Data Reporter • Deadline • Disk Organizer II
  - Egbert II Communic. Disk • Hard Hat Mack
  - Home Accountant • Homeword • Lancaster
  - Magic Window II • Multi-disk Catalog
  - Multiplan • Pest Patrol • Prisoner II
  - Sammy Lightfoot • Screen Writer II
  - Sneakers • Spy's Demise • Starcross
  - Suspended • Ultima II • Visifile
  - Visiplot • Visitrend • Witness • Wizardry
  - Zork I • Zork II • Zork III
- PLUS 'how-to' articles and listings of need-to-have programs used to make unprotected backups.

## Volume II

Compiled from issues 6-10

contains softkeys for:

- Apple Cider Spider • Apple Logo • Artist
  - Arcade Machine • Bank Street Writer
  - Cannonball Blitz • Canyon Climber
  - Caverns of Freitag • Crush, Crumble & Chomp
  - Data Factory V • DB Master • The Dic'tion'ary
  - Essential Data Duplicator I & III • Gold Rush
  - Krell Logo • Legacy of Llylgamyn
  - Mask Of The Sun • Minit Man • Mouskattack
  - Music Construction Set • Oil's Well
  - Pandora's Box • Robotron • Sammy Lightfoot
  - Screenwriter II v2.2 • Sensible Speller 4.4c.4.1c
  - Spy Strikes Back • Time Zone v1.1
  - Visible Computer: 6502 • Visidex • Visiterm
  - Zaxxon • software for: • Hayden • Sierra Online
- PLUS the ultimate cracking program:  
Super IOB 1.5 •••and more!

## Volume III

Compiled from issues 11-15

contains softkeys for:

- Alien Addition • Alien Munchies • Alligator Mix
  - Comp. Prep. SAT • Cut&Paste • Demolition Div.
  - DLM software • EA (Electron. Arts) software
  - Einstein Compiler 5.3 • Escape From Rungistan
  - Financial Cookbook • Flip Out • Hi-res Comp. Golf II
  - Knoware • Laf Pak • Last Gladiator
  - Learning With Leeper • Lion's Share
  - Master Type 1.7 • MatheMagic • Minus Mission
  - Millionaire • Music Construction Set
  - One-on-one • Penguin software • PFS software
  - The Quest • Rocky's Boots • Sabotage
  - Seadragon • Sensible Speller 4 • Snooper Troops II
  - SoftPorn Adventure • Stickybear series • Suicide
  - TellStar • Tic Tac Show • Time Is Money
  - Transylvania • Type Attack • Ultima III Exodus
  - Zoom Graphics • Breaking Locksmith 5.0 Fast Copy
- PLUS feature articles on • C saver  
• The Core Disk Searcher • Modified ROMs.

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

Address \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_

Country \_\_\_\_\_ Phone \_\_\_\_\_

  \_\_\_\_\_ Exp. \_\_\_\_\_

Signature \_\_\_\_\_ CP49

Volume I - \$7.95  
+ \$2 shipping/handling

Volume II - \$12.95  
+ \$2 shipping/handling

Volume III - \$17.95  
+ \$2 shipping/handling

All 3 volumes! - \$30.00  
+ \$2 shipping/handling

Foreign orders (except Canada and Mexico) please add \$5 for shipping and handling. Washington residents add 7.8% sales tax. Most orders are shipped within 5 working days, however, please allow 4-6 weeks delivery. US Funds drawn on US banks only.

Send to: Book Of Softkeys PO Box 110846-T Tacoma, WA 98411 (206) 474-5

# the COMPUTIST shopper

Software Package	II +	II gs	Software Package	II +	II gs
816 Paint	\$45.00	\$45.00	Portal	\$30.00	
Airheart	\$24.00		The Print Shop	\$34.00	\$40.00
Award Maker Plus	\$24.00		Print Shop Companion	\$27.00	
Bank Street Writer 64K	\$46.00		Print Shop Graphics Library:		
Bank Street Speller	\$46.00		Disk One	\$16.50	
Bank Street Mailer 64K	\$46.00		Disk Two	\$16.50	
Bank Street Filer 64K	\$46.00		Disk Three	\$16.50	
Bank Street Writer Plus (128K)	\$46.00		Pro-Byter	\$20.00	
Bank Street Mailer 128K	\$46.00		Racter	\$30.00	
Bank Street Filer 128K	\$46.00		Reader Rabbit	\$27.00	\$34.00
Beagle Graphics	\$35.00		Rescue Raiders	\$23.00	
Beyond Zork	\$27.00		Rocky's Boots	\$34.00	
Black Cauldron	\$27.00		Science Tool Kit: Master Module	\$46.00	
Captain Goodnight	\$24.00		SDI	\$32.00	
Championship Karate	\$14.00		Shanghai	\$24.00	\$30.00
Clip Art Gallery (for Paintworks Plus)	\$20.00		Silent Service	\$23.00	
Copy II Plus	\$23.00		Solo Flight	\$23.00	
Dazzle Draw	\$40.00		Space Quest	\$32.00	\$32.00
Defender of the Crown	\$32.00		Star Trek: The Kobayashi Alternative	\$27.00	
Destroyer	\$25.00	\$27.00	Star Trek II: The Promethean Prophecy	\$27.00	
DiskQuirk	\$18.00		Stickybear Series:		
Draw Plus	\$52.00		ABC's	\$24.00	
F-15 Strike Eagle	\$24.00		Math I	\$24.00	
Fantavision	\$34.00	\$40.00	Math II	\$24.00	
Flight Simulator II	\$36.00		Numbers	\$24.00	
Gamemaker	\$34.00		Reading	\$24.00	
GPLÉ	\$29.50		Shapes	\$24.00	
The Graphics Studio	\$42.00		Typing	\$24.00	
Gunship	\$27.00		Writer	\$24.00	
Hacker II	\$25.00	\$30.00	Sub Battle Simulator	\$25.00	\$27.00
Hardball	\$22.00	\$28.00	Sub Mission	\$27.00	
Hitchhiker's Guide	\$20.00		Summer Games	\$14.00	
Infiltrator II	\$20.00		Super Macroworks	\$30.00	
Jet	\$29.00		Tass Times in Tonetown	\$24.00	\$24.00
Karateka	\$24.00		The Tower of Myraglen	\$32.00	
King's Quest II	\$32.00	\$32.00	Thexder	\$23.00	
Knight of Diamonds	\$23.00		Topdraw	\$66.00	
Labrinth	\$20.00		Type	\$30.00	
Legacy of Lylgamyn	\$26.00		Typing Tutor IV	\$32.00	\$32.00
Loderunner	\$24.00		Video Vegas	\$18.00	
Macroworks	\$20.00		Visualizer	\$53.00	\$59.00
Math Blaster	\$28.00		Where in USA is Carmen San Diego	\$26.50	
Math Rabbit	\$27.00		Where in World is Carmen San Diego	\$30.00	
Mean 18, Ultimate Golf	\$28.00		Winter Games	\$25.00	\$27.00
Merlin ProDOS	\$66.00		Wizards	\$32.00	
Might & Magic	\$32.00		Word Attack	\$28.00	
Millionaire II	\$33.00		Wordperfect w/ Spelling Checker	\$95.00	\$95.00
Mousewrite	\$99.00	\$99.00	World Games	\$25.00	\$27.00
Multiscribe	\$45.00	\$66.00	Writer's Choice Elite	\$60.00	
Music Studio	\$52.00		Writer Rabbit	\$24.00	
Paintworks Plus	\$52.00		Zork Trilogy	\$46.00	

**Do Your  
Christmas  
Shopping  
Now**

**Check our  
LOW  
software  
prices  
for  
Apple II  
and IIGs  
software**

#### How To Order

- **US orders:** Check the appropriate box for your selection. For Apple IIGs software, check the appropriate box in the right-hand column.
- Please add \$3 per order for shipping & handling. Orders over \$200 receive **free** shipping.
- Most orders shipped UPS, so use your **street address**.
- **Washington** state, please add 7.8% sales tax.
- Offer good while supplies last.
- All products are for the Apple II systems unless otherwise specified.

- **Foreign Orders:** Please inquire as to appropriate shipping fees.

Send orders to:

**Softkey Publishing**  
PO Box 110816-T  
Tacoma, WA 98411  
(206) 474-5750

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

Address \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_

Country \_\_\_\_\_ Phone \_\_\_\_\_

  \_\_\_\_\_ Exp. \_\_\_\_\_

Signature \_\_\_\_\_ CP49

- Cost of Software SUBTOTAL \_\_\_\_\_
- Plus SHIPPING \_\_\_\_\_
- Plus (in WA) Sales TAX \_\_\_\_\_

■ TOTAL \_\_\_\_\_

# COMPUTIST BACK ISSUES ORDER FORM

Are you still missing  
a piece of the picture ???!!!

Issue	Mag \$4.75	Disk \$9.95	Both \$12.95
48	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
47	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
46	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
45	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
44	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
43	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
42	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
41	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
40	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
39	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
38	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
37	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
36	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
35	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
34	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
33	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
32	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
31	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
30	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
29	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
28	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
27	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
26	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
25	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
24	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
★ 23	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
22	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
19	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
16	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
★ 7	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Core 2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Core 1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Computing 3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Best of Hardcore Computing	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Core Special Combo \$10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

(All three CORE magazines)

Some disks apply to more than one issue and are shown as taller boxes.  
Special "Both" disk & magazine combination orders apply to one issue  
and its corresponding disk.

★ We have a limited supply of these issues.  
● Back issue is no longer available

## COMPUTIST

back issues and library disks are  
frequently referenced in current issues.

### Back Issues and Library Disk Rates

- US, Canada and Mexico back issue rate - \$4.75 each.
- All other Foreign back issue rate - \$8.75 each.
- US, Canada, Mexico library disk rate - \$9.95 each.
- All other Foreign library disk rate - \$11.94 each.
- "Both" disk and magazine rates for:  
US, Canada & Mexico - \$12.95 each combination.  
All other Foreign - \$18.95 each combination.

### What is a library disk?

A library disk is a diskette that contains programs that would normally have to be entered by the user. Documentation for each library disk can be found in the corresponding issue.

- Library disks are available for all issues of COMPUTIST 1 thru 45. A description of the softkeys and programs covered in each issue is available upon request. Please send your name and address along with a first class postage stamp (US \$ .22).

### Complete Your Collection!

**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 • 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

**Hardcore Computing 3** HyperDOS Creator • Menu Hello • Zephyr Wars • Vector Graphics • Review of Bit Copiers • Boot Code Tracing • Softkey IOB • Interview with 'Mike' Markkula

Please send the back issues indicated:

Please send the following library disks:

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

Address \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_

Country \_\_\_\_\_ Phone \_\_\_\_\_

 \_\_\_\_\_ Exp. \_\_\_\_\_

Signature \_\_\_\_\_ CP49

Send check or money order to:

COMPUTIST  
PO Box 110846-T  
Tacoma, WA 98411.  
(206) 474-5750

Most orders are shipped within 5 working days, however please allow up to 4 weeks delivery for some orders. Most orders shipped UPS, so please use street address. Offer good while supply lasts. In Washington state, add 7.8% sales tax.

US funds drawn on US banks

How to get your letter published in **COMPUTIST's...**

## Readers Data EXchange

### No 'Input' or 'Readers Softkey & Copy Exchange'

The old INPUT and the READERS SOFTKEY AND COPY EXCHANGE columns have now been combined into the new Readers Data Exchange (or RDEX, pronounced AR-DEX).

All softkeys, comments, articles, questions, answers, fixes and bugs and almost any other information that must be printed in a timely fashion will be placed in RDEX.

Although we will print letters and article-softkey submissions that are received as only printed text, we would prefer that they be submitted on disk.

### We no longer purchase articles or softkeys

That's right. The author's 'by-line' is the only payment offered to COMPUTIST writers who get published. Even the staff is composed of unpaid volunteers.

Of course, COMPUTIST reserves the right to edit and publish ANY contribution (whether submitted as a letter-to-the-editor or article-softkey) in the RDEX section.

All published contributions become the property (and are copyrighted as part) of COMPUTIST magazine.

### How to get your name in COMPUTIST RDEX

Whether you are contributing data to RDEX or just asking a question, we urge you to follow these writer's guidelines.

- All contributions and submissions should be submitted in **both** printed format and on disk.
- Your printout must use only one side of the paper.
- Text should be double-spaced using a non-compressed font with both upper and lower case.

- A letter quality mode is preferred, with each page torn at the perforation only. Pages need not be stapled together.
- The first page should contain the following data:

**TITLE OF WORK  
FULL NAME OF AUTHOR  
ADDRESS  
PHONE NUMBER**

- Each page of the manuscript and program listing should include the author's name, the title of the work, and the page number in the upper right hand corner.

### A standard text file on a DOS 3.3 5 1/4" floppy disk

The article and any accompanying program should be submitted as a **standard text file on a DOS 3.3 disk**. Label the disk with the title of the work and the author's full name and address.

**ON DISK, TEXT MUST BE SINGLE-SPACED ONLY.** Please identify your editing program.

We always try to return your disks as soon as possible. Still, to protect ourselves, we must say that we are not responsible for unreturned submissions.

### Here's how best to 'talk' to COMPUTIST readers

**A.** Always assume that your reader is a novice and explain all buzzwords and technical jargon. Pay special attention to grammar and punctuation. We require technical competence and good, readable style.

**B.** A list of hardware and software **Requirements** should be included at the beginning of the manuscript. When published, this list will be offset from the main text.

**C.** Include the manufacturer's or publisher's name when a commercial program or product is mentioned.

**D.** When submitting your own programs, first introduce the purpose of the program and features of special interest. Include background information describing its use.

Tips for advanced uses, program modifications, and utilities can also be included. Avoid long print statements and use TABs instead of spaces.

*Remember:* A beginner should be able to type the program with ease. A program will not be accepted for publication without an accompanying article.

These articles, as well as articles on hardware and DOS modifications **MUST** summarize the action of the main routines and include a fully remarked listing.

**F. GENERAL ARTICLES** may include advanced tips, tutorials, and explorations of a particular aspect of Apple computing.

**G. SOFTKEYS** must contain detailed step-by-step procedures. Number each step.

For each softkey, first introduce the locking technique used and then give precise steps to unlock the copy-protected program.

### Please do NOT send any original commercial disks

We accept articles which explain locking techniques used in several programs published by the same company.

**H. APTs or ADVANCED PLAYING TECHNIQUES** can deal with **alterations** to a program, deleting annoying sounds, acquiring more points in play and avoiding hazards.

Again, provide step-by-step instructions to complete each APT and explain each step's function.

Please mail all letters, short articles and softkeys to the *RDEX Editor*. When accepted for publication, they will be **IMMEDIATELY** edited, typeset and entered into the next RDEX.

Remember that we cannot purchase these contributions.

**RDEXed  
PO Box 110846-T  
Tacoma, WA 98411**

# The Hacker's Ultimate Copy & Deprotection Utility Super IOB Collection !

ALL of our Super IOB controllers (through 1986) in 'ONE' package!

COMPUTIST developed the ultimate copy program to remove copy protection from software:

### The Super IOB program.

Since the introduction of Super IOB, COMPUTIST has used this flexible program to deprotect (or partially deprotect) dozens of commercial programs with far ranging protection schemes.

Super IOB deprotects disks by using a modified RWTS (the subroutine in DOS which is responsible for the reading and writing of disk sectors) for reading from the protected disk and then using a normal RWTS for writing to the deprotected disk.

### This package contains:

▶ **TWO DISKS** (supplied in DOS 3.3). Each disk contains at least 60 Super IOB Controllers including the standard, swap, newspaper and fast controllers. Also included is **version 1.5 of Super IOB**, the Csaver program from COMPUTIST No. 13, and a Menu Hello Program that lists the available controllers and, when you select one, automatically installs it in Super IOB and RUNs the resulting program.\*

▶ A reprint of **Disk Inspection and the Use of Super IOB**, from COMPUTIST No. 17. This article explains how to write your own Super IOB controllers.

▶ **COMPUTIST No. 32**, which contains an extensive article detailing the hows and whys of Super IOB v1.5 and at least 5 articles using the new Super IOB program.

● Several of the controllers deprotect the software completely with no further steps. This means that some programs are only minutes away from deprotection (with virtually no typing).

● The issue of COMPUTIST in which each controller appeared is indicated in case further steps are required to deprotect a particular program.\*\*

### The SUPER IOB Collection

**Volume 1** of the Super IOB collection covers all the controllers from COMPUTIST No. 9 through No. 26. Also included are the newspaper and fast controllers from COMPUTIST No. 32. The following 60 controllers are on volume 1:

Advanced Blackjack, Alphabet Zoo, Arcade Machine, Archon II, Archon, Artsci Software, Bank Street Writer, Barrons SAT, Beyond Castle Wolfenstein, BSW //c Loader, Castle Wolfenstein, Computer Preparation: SAT, Dazzle Draw, DB Master 4 Plus, Death in the Carribean, Dino Eggs, DLM Software, Electronic Arts, F-15 Strike Eagle, Fast Controller, Fathoms 40, Financial Cookbook, Gessler Software, Grandma's House, The Heist, In Search of the Most Amazing Thing, Instant Recall, Kidwriter, Lions Share, Lode Runner, Mastertype, Match Maker, Miner 2049er, Minit Man, Mufplot, Newsroom, Newspaper controller, Penguin Software, Print Shop Graphic Library, Print Shop, Rendezvous with Rama, Rockies' Boots, Sargon III, Sea Dragon, Shiela, Skyfox, Snooper Troops, Standard controller, Stoneware Software, Summer Games, Super Controller, Super Zaxxon, Swap Controller, TAC, Ultima I II, Word Challenge, Xyphus, Zaxxon

**Volume 2** of the Super IOB collection covers all the controllers from COMPUTIST No. 27 through No. 38. The following 65 controllers are on volume 2:

Alice in Wonderland, Alphabetic Keyboarding, Alternate Reality, Autoduel, Checkers, Chipwits, Color Me, Conan.data, Conan.prog, CopyDOS, Crisis Mountain, Disk Director, Dragonworld, Early Games, Easy as ABC, F-15 Strike Eagle, Fantavision, Fast controller, Fishies, Flight Simulator, Halley Project, Hartley Software (a), Hartley Software (b), Jenny of the Prairie, Jingle Disk, Kidwriter, Kracking Vol II, Lode Runner, LOGO II (a), LOGO II (b), Masquerade, Mastering the SAT, Microtype: The Wonderful World of Paws, Microzines 1, Microzines 2-5, Miner 2049er, Mist & View to a Kill, Murder on the Zinderneuf, Music Construction Set, Newspaper controller, Olympic Decathlon, Other Side, Phi Beta Filer, Pitstop II, Print Shop Companion, RDOS, Robot War, Spy vs Spy, Standard controller, Sundog V2, Swap controller, Sword of Kadash, Synergistic Software, Tawala's last Redoubt, Terripin Logo, Threshold, Time is Money, Time Zone, Tink! Tonk!, Troll's Tale, Ultima IV, Wilderness, Word Attack & Classmate, World's Greatest Baseball, World's Greatest Football

**Yes, please send me The Super IOB Collection**

Includes both disks with Super IOB version 1.5, COMPUTIST32, PLUS a reprint of "Disk Inspection and the Use of Super IOB".

- US/Canada/Mexico for \$16.00**  
 **Other Foreign for \$20.00**

Send to: **Super IOB Collection**  
**PO Box 110846-T**  
**Tacoma, WA 98411**  
**(206) 474-5750**

\*Requires at least 64K of memory.

\*\*Although some controllers will completely deprotect the program they were designed for, some will not, and therefore require their corresponding issue of COMPUTIST to complete the deprotection procedure.

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

Address \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_

Country \_\_\_\_\_ Phone \_\_\_\_\_

 \_\_\_\_\_ Exp. \_\_\_\_\_

Signature \_\_\_\_\_ CP49

Most orders are shipped within 5 working days, however, please allow 4 to 6 weeks for delivery. Washington residents, please add 7.8% sales tax.

**US funds drawn on US banks**