

For The Serious User Of Personal Computers

# COMPUTIST

Issue 61

November 1988

USA \$3.75

Canada & Mexico \$7.00

All Others \$8.75

## 🍏 RDEX *Features:*

- **Double DOS**—By Edward Teach
- **Deprotection tips for DOS and ProDOS**—By Gary Verbuch
- **A Search Command for II's**—By Bill Jetzer
- **Zip Chip finally Ships**—By Gary Verbuch & Nemoran Pierre
- **The Product Monitor**—By Jeff Hurlburt
- **Put Where in Europe is Carmen Sandiego on a 3<sup>1/2</sup>" disk**—By Jim S. Hart
- **A self-booting Thexder on a 3<sup>1/2</sup>" disk**—By Jim S. Hart
- **Capture protected RWTS with RWTS Worm**—By Jim S. Hart

## 🍏 RDEX *Softkeys:*

Adventures of Sinbad 🍏 ARCHIVEmath 🍏 Aztec 🍏 Broadsides v2.0 🍏 Calculus Toolkit 🍏 Cannonball Blitz 🍏 Catalyst 3.0  
🍏 Cofa Courseware 🍏 Cross Clues 🍏 Deathsword 🍏 Defender of the Crown 🍏 Dondra 🍏 Gauntlet 🍏 G.I. Joe 🍏 King's Quest  
I 🍏 King's Quest II 🍏 King's Quest III 🍏 Magical Myths 🍏 Math Shop 🍏 Mathblaster 🍏 Mathbusters 🍏 Mavis Beacon Teaches  
Typing 🍏 Megabots 🍏 Micro Addition/Subtraction 🍏 Microzine #25 🍏 Microzine #26 🍏 One-on-One 🍏 PFS Computer Checkup  
🍏 PFS File 🍏 PFS Graph 🍏 PFS Report 🍏 Pirates! 🍏 Pirates! IIgs 🍏 Presenter 🍏 Rescue on Fractalus 🍏 Science Toolkit 🍏 Smart  
Eyes 🍏 Softswitch 🍏 Spiderbot 🍏 Tales from the Arabian Nights 🍏 Thinkware 🍏 Tink's Subtraction Fair 🍏 Tower of Myraglen  
🍏 Ultima V 🍏 Xevious

**IBM RDEX:** *Softkey:* ■ Print Shop ■ Test Drive

COMPUTIST  
PO Box 110846-T  
Tacoma, WA 98411

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

# **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*

## **COMPUTIST**

Editor: Charles R. Haight

Circulation: Karen Fitzpatrick

Advertising, call: (206) 474-5750

Publisher: SoftKey Publishing

Printing: Vakco Graphics & Printing

● Address all advertising inquiries to: COMPUTIST; Advertising Department; PO Box 110816; Tacoma, WA 98411

● Mail all RDEX letters to:  
COMPUTIST  
Apple-RDEX or IBM-RDEX  
PO Box 110846-K  
Tacoma, WA 98411

● COMPUTIST does NOT purchase editorial material. The entire editorial content consists of information submitted to COMPUTIST for publication in the shared interests of all COMPUTISTs.

● Unsolicited material (manuscripts, letters to the editor, softkeys, A.P.T.s, playing tips, questions, etc.) are assumed to be submitted as letters-to-the-RDEX-editor for publication with all and exclusive rights belonging to COMPUTIST.

● Entire contents copyright 1988 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 assume no liability or responsibility for the products advertised in the magazine. Any opinions expressed by the authors are not necessarily those of COMPUTIST magazine, its staff or SoftKey Publishing.

● Apple® is a trademark of Apple Computers. IBM® is the IBM trademark.

**SUBSCRIPTIONS:** Rates (for 12 issues):  
U.S.—\$32 U.S. 1st Class—\$45  
Canada/Mex.—\$45 Other Foreign—\$75

● Send subscription inquiries to: COMPUTIST; Subscription Department; PO Box 110846-T; Tacoma, WA 98411

● For 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.



# Notes 'n things.....

## ■ Send your letters on disk

I've been getting more and more of your letters, on disk and it's great. I love it. A lot less editing and a lot less errors. Of course, if you're one of the people who sent your letter as hardcopy (on paper), you are probably not to thrilled with my response time. (I'm about 2 months behind in my written correspondence.) I'm making a special effort and I'll have most of your letters typed and in the next COMPUTIST. (I hope.) I do want to say again and reemphasize, *if your letter is more than half of a page, please send it on disk.* I'm very good about returning disks. But I'm a rotten typist (die-hard two finger and a thumb type). I wish we had a real typist but they normally don't work for free. In fact, I wish we had a few hundred more subscribers so I could also add another 16 pages for all the new stuff that has been coming in on disk.

## ■ Questions

The following are some questions that got a chuckle out of Karen. She's the person who answers the phone, processes and packs your orders, as well as entering new subscriptions and renewals into the computer. Be nice to her, she works a full day and only gets minimum wage. If she doesn't process your order, we would have too. (You wouldn't want someone who types with two fingers to enter and pack your order, would you?)

From a non-subscriber who ordered a back issue from an ad flyer:

**Q.** Along with my order, do I get the (\$32 value) free subscription?

**A.** Nice try. The subscription is \$32. The FREE software package (\$32 value) comes with it. Gotta buy one to get the other.

From a subscriber who was renewing:

**Q.** Please send a list of currently available products (Disks & Books).

**A.** We send you a new updated list every month. All our products are listed in the back of each issue, however, we have access to a number of wholesalers. If there is something you're looking for, write and ask, we'll tell you the best price we can find.

From a new Foreign subscriber:

**Q.** I would like to ask you, How come the back issues are more expensive then the subscription?

**A.** Because we want you to subscribe... Sorry, just kidding. Regular subscriptions are handled automatically once a month by a mail forwarding service. All of the foreign orders are picked up here and flown to New York where they are sorted and flown to their respective countries. There they are put into the regular mail system. Back issue orders must be handled on an individual basis and sent by regular mail service (1st Class Air). That means they cost more to send and take more time to process.

## ■ Starter Kit Woes!

If your printer just spews paper when you try to print the DOC files, try this tip; when the program asks for the right margin (80), it really wants to know the right indent (normally 0). Tell it zero and things should work fine. Just a syntax problem with the man who wrote the print routine. Mike say's he'll have it fixed in a jiffy.

## ■ Missed Issues?

Issues are mailed at the end of each month and can take up to 4 weeks to get to you. (Depends on where you are and what the US snail is doing about then.) Give the issue enough time to get to you. But, if you feel that too much time has gone by without a COMPUTIST appearing in your mail, write to Karen (or call). Don't wait for 6 months before you say anything. Karen can be very unsympathetic to your tale of woe and will only replace the last issue or two.

# BUGS:

**COMPUTIST #58, page 32.** Computing for 1-3 year olds - In listing 1, there is an extra open parenthesis in line 220. It should read:

220 NN = INT ( RND (1) \* FC + 1)

◊ ◊ ◊ ◊ ◊ ◊ ◊ ◊ ◊ ◊ ◊ ◊ ◊ ◊ ◊

Jim Bancroft	15
Ted E. Bare	24
Captain Dan	36
Dan Halfwit	7
Jim S. Hart	33
MJ Howard	21
Jeff Hurlburt	28
Bill Jetzer	20,25
Blain Johnson	30
Donald Jones	33
Robert Kettle	30
Craig Meekins	23
Jack Nissel	21
Nemoran Pierre	26
Authur Simon	18
Silicon Sorcerer	18
Charles Taylor	19
Edward Teach	12
Dr. Jean-Louis Torre	30
Brian A. Troha	8
Gary Verbuch	22,26
J.L. Walters	23
John E. Wanner	17

IBM IBM

Mike Basford	44
C.J. Blanchard	44
Kevin Dobbins	44
Neal Jhala	44

# COMPUTIST #61

Table Of Contents

November

1988

## 🍏 Softkeys:

<i>Adventures of Sinbad</i> .....	12
<i>ARCHIVEmath</i> .....	35
<i>Aztec</i> .....	23
<i>Broadsides v2.0</i> .....	23
<i>Calculus Toolkit</i> .....	33
<i>Cannonball Blitz</i> .....	21
<i>Catalyst 3.0</i> .....	20
<i>Cofa Courseware</i> .....	19
<i>Cross Clues</i> .....	15
<i>Deathsword</i> .....	19
<i>Defender of the Crown</i> .....	11
<i>Dondra</i> .....	36
<i>Gauntlet</i> .....	19
<i>G.I. Joe</i> .....	21
<i>King's Quest I</i> .....	11
<i>King's Quest II</i> .....	11
<i>King's Quest III</i> .....	36
<i>Magical Myths</i> .....	10
<i>Math Shop</i> .....	35
<i>Mathblaster</i> .....	15
<i>Mathbusters</i> .....	20
<i>Mavis Beacon Teaches Typing</i> .....	9
<i>Megabots</i> .....	16
<i>Micro Addition/Subtraction</i> .....	34
<i>Microzine #25</i> .....	23
<i>Microzine #26</i> .....	24
<i>One-on-One</i> .....	24
<i>PFS Computer Checkup</i> .....	16
<i>PFS File</i> .....	22
<i>PFS Graph</i> .....	22
<i>PFS Report</i> .....	22
<i>Pirates!</i> .....	7
<i>Pirates! IIgs</i> .....	8
<i>Presenter</i> .....	20
<i>Rescue on Fractalus</i> .....	18
<i>Science Toolkit</i> .....	25
<i>Smart Eyes</i> .....	16
<i>Softswitch</i> .....	9
<i>Spiderbot</i> .....	34
<i>Tales from the Arabian Nights</i> .....	11
<i>Thinkware</i> .....	16
<i>Tink's Subtraction Fair</i> .....	21
<i>Tower of Myraglen</i> .....	21
<i>Ultima V</i> .....	37
<i>Xevious</i> .....	7

## 🍏 A.P.T.s:

<i>Bard's Tale III</i> .....	32,33
<i>Ultima V</i> .....	30,31

## 🍏 Playing Tips:

<i>Ultima V</i> .....	30
<i>Wasteland</i> .....	17

## IBM Softkeys:

<i>Print Shop</i> .....	45
<i>Test Drive</i> .....	44

## IBM Playing Tips:

<i>Leisure Suit Larry in the Land of the lounge Lizards</i> ..	44
--	----

<b>Editorial</b> .....	4
<b>Bugs</b> .....	4
<b>Apple Most Wanted Softkeys</b> .....	37
<b>IBM Most Wanted Softkeys</b> .....	45

## 🍏 Features:

<b>Double DOS</b> .....	12
An easier way to copy disks with multiple address and data mark changes.	
<b>Deprotection tips for DOS and ProDOS</b> .....	22
Important locations and places to poke.	
<b>A Search Command for II's</b> .....	26
Copy the new search command from the enhanced IIe and put it into your older II.	
<b>Zip Chip finally Ships</b> .....	26
A reader review of a long awaited product.	
<b>The Product Monitor</b> .....	28
Jeff Hurlburt takes a look at super-res graphics on the IIgs and Chess on the MAC.	
Put a deprotected...	
<b>Where in Europe is Carmen Sandiego on a 3½" disk</b> ....	33
Get rid of that 5¼" disk and forget about flipping disks.	
<b>A self-booting Thexder</b> .....	34
Put ProDOS and Thexder on a 3½" disk.	
Capture protected RWTS with...	
<b>RWTS Worm</b> .....	35
Here's a neat way to extract the RWTS from a protected disk.	
<b>Some notes on Mindscape</b> .....	16
<b>Some notes on Scholastic</b> .....	16
<b>Some notes on Hartley Software</b> .....	16
<b>Some notes on Prentice Hall</b> .....	16
<b>Some notes on CAI</b> .....	16
<b>Some notes on CTW Series</b> .....	17
<b>Some notes on Garfield</b> .....	17



# Readers Data

For serious users of Apple computers: II, II+, //c, //e, IIGs, Macintosh

New COMPUTIST readers using Apple IIs are advised to read this page carefully to avoid frustration when attempting to follow a softkey or 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 backup copy can usually be copied by the normal copy programs (for example: COPYA, on the DOS 3.3 System Master disk).

## Commands and control keys

Commands which a reader is required to perform are set apart by being in boldface and on a separate line. The **RETURN** key must be pressed at the end of every such command unless otherwise specified. Control characters are specially boxed. An example of both is:

**6** **CP**

Press **6**. Next, place one finger on the **CTRL** key and then press **P**. Don't forget to press **RETURN**.

Other special combination keypresses include **RESET** or **CTRL-RESET**. In the former, press and hold down **CTRL** then press **RESET**. In the latter, press and hold down both **CTRL** and **C** then press **RESET**.

## Software recommendations

The Starter Kit contains most of the programs that you need to "Get started". In addition, we recommend that you acquire the following:

- Applesoft program editor such as "Global Program Line Editor (GPLE)".
- 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 "Essential Data Duplicator (EDD)".
- 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 (in the 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 (COMPUTIST #6 or Book Of Softkeys III) or the "Dual ROM's" article (COMPUTIST #19).

**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 //c warranty.

**Apple //gs:** If you have the 2.x ROM, there is a hidden classic desk accessory (CDA) that allows you to enter the monitor. In order to install the new CDA, you should enter the monitor before running any protected programs (CALL -151) and press "# **RETURN**". This will turn on two hidden CDAs, *Memory Peeker* and *Visit Monitor*. Thereafter press **CTRL-ESC** to go to the Desk Accessories menu. Select "Visit Monitor" and there you are. Use **CTRL-Y** to exit.

## Recommended literature:

- *Apple II Reference Manual*
- *DOS 3.3 manual*
- *Beneath Apple DOS & Beneath Apple ProDOS*, by Don Worth and Pieter Lechner, from Quality Software

## Keying in Applesoft programs:

BASIC programs are printed 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 don't pose a problem except when they are inside of quotes or after a DATA command. 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, only type spaces within quotes or after DATA statements if they are shown as delta (^) characters.

## 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. When finished, return to BASIC with:

**3DOG**

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

The source code is printed to help explain a program's operation. To enter it, you need an "Assembler". Most of the source code is in *S-C Assembler* format. If you use a different assembler, you will have to translate pieces of the source code into something your assembler will understand.

## Computing checksums

Checksums are 4-digit hexadecimal numbers which tell if you typed a program correctly. 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 are on the "Starter Kit".

If your checksums do not match the published checksums then the line where the first checksum differs is incorrect.

**CHECKSOFT** instructions: Install Checksoft (BRUN CHECKSOFT) then LOAD your program. Press **6** to get the checksums. Correct the program line where the checksums differ.

**CHECKBIN** instructions: Enter the monitor (CALL -151), install Checkbin at some out of the way place (BRUN CHECKBIN, A\$6000), and then LOAD your program. Get the checksums by typing the Starting address, a period and the Ending address of the file followed by a **CTRL-Y**.

**SSSS.EEEE CTRL-Y**

Correct the lines at which the checksums differ. .... RDEXed

# EXchange



etc... who want all their software backed up and COPYA-able

when  
writing a  
letter to the...



## RDEX editor

RDEX stands for:

### Reader's Data EXchange

That means that when you send in articles, softkeys, APTs, etc., you are submitting them for FREE publication in this magazine. RDEX does NOT purchase submissions nor do we verify data submitted by readers. We print what you write. If you discover any errors, please let us know.

- Remember that your letters or parts of them may be used in RDEX even if not addressed 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 part-time staff, any response to your queries will appear only in 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 Apple-RDEX.

- Whenever possible, send your articles and letters on disk as standard text files. When we get your letter-article in a text file, it is immediately uploaded into the most current RDEX file. We will return your disks, whenever possible, with the current library disk copied onto it. Conventional letters must be typed in by us... when we have the time.

- When you send your material on disk, try to use a disk formatted for Apple DOS 3.3. It makes things a lot easier for us.

- Address your letters, articles, to:

COMPUTIST  
R D E X Editor  
PO Box 110846-K  
Tacoma, WA 98411

Dan Halfwit

Softkey for...

**Xevious**  
Mindscape

#### ■ Requirements

- COPYA
- Copy II Plus or FID

This was too easy.

**1** Use COPYA and the following pokes to disable the checksum verify:

**POKE 47405,24**  
**POKE 47406,96**  
**POKE 47497,24**  
**POKE 47498,96**  
**RUN COPYA**

**2** Use Copy II Plus to copy a normal DOS over the first 3 tracks of the copy and change the boot file to HELLO.

#### Optional:

**2** Initialize a disk.

**INIT HELLO**  
**DELETE HELLO**

Insert the disk with FID and  
**BRUN FID**

Copy all files from the COPYA copy of XEVIIOUS to the initialized disk using the wildcard (=) character.

Softkey for...

**Pirates!**  
MicroProse

#### ■ Requirements

- Apple IIgs with 2.x ROM upgrade or Apple II w/128K & NMI card
- Two (ProDOS) formatted double-sided disks
- BASIC.SYSTEM from any ProDOS System Master
- Copy II Plus (or a ProDOS file copier)

I didn't make a Cookbook at the end, because you may just as well read the whole article. It will take all of 3 minutes.

I played the game for about 8 hours and nothing funny happened. It would appear that I got all the protection schemes.

Previously, the bulk of MicroProse games were in normal format and the catalog track was moved. Sometimes not even that. You would only get so far, to then find that another protection scheme found you. With Pirates!, you don't even start on square one. Some format that I am not at all familiar with popped up. Fortunately, I found a small hole in it, and that was all I needed.

First, format one of your ProDOS disks with the volume name of /PIRATES.SIDE1. (Don't type the period after the number 1.) Then copy BASIC.SYSTEM to that disk, and rename it to PIRATES.SYSTEM. Starting to get the point? Now, the fun part.

For IIgs users, enter the monitor (CALL-151) and type a # sign at the monitor prompt. This will give you the CDA's (Classic Desktop Accs.). Put Pirates! in drive 1 and type C600G at the prompt to boot the disk. Listen to the drive. After the arm of the disk clicks ONCE, Press **ESC** or the NMI button if you have it.

Select "Visit Monitor" from the Control Panel (IIgs users.) At the monitor prompt, type A0A:EA EA to kill the "ACCESS ERROR #1" message (didn't that feel good, and on an original disk too!) and 9E7:4C 69 FF to replace the command to start ProDOS with a monitor bomb. Now, press **Y** and **RETURN** to return to the control panel. Select QUIT. Pirates! will go a little more and give you a monitor prompt. You now have their funny DOS in memory to do your bidding!

Remove Pirates! and replace it with the ProDOS disk you made earlier with the renamed BASIC.SYSTEM. Type 2000G at the monitor prompt, and in a few seconds, you are in BASIC. You now have a DOS in memory that reads normal and abnormal (Pirates!) DOS. Use the "-" wildcard command to start Copy II Plus (PREFIX /COPYIIPLUS and -UTIL.SYSTEM) or a ProDOS file copier and copy all the files from both sides of Pirates over to a normal ProDOS disk. Make sure that you name the volumes as /PIRATES.SIDE1 and /PIRATES.SIDE2. Now the real fun part.

If you boot up what you now have, the game will go a little past the title page and give you a (believe it or not) "ACCESS ERROR #1". But didn't we just kill that? If you search the disk, you find one occurrence of this in the PIRATES.SYSTEM file, but that isn't it. Search for "1#RORRE SSECCA" (ACCESS ERROR #1 spelled backwards) and lo and behold. This is what we want. We find some most curious looking code at \$E51. Let's take a peek:

```

E51 EOR $B800,X
E54 INX
E55 CPX #$FC
E57 BCC $0E51          (-08)
E59 TAX
E5A BNE $0E61          (+05)
E5C JSR $B863
E5F BEQ $0E90          (+2F)
E61 ... prints ACCESS ERROR #1 and locks up
    
```

At first, the above code didn't make much sense. It looked harmless enough. But if you turn on double hi-res and type E61G, there we see the ACCESS ERROR #1. Here's a little explanation. \$E51-E59 verifies the code at \$B800 to make sure that no one tampered with it. (It checks \$FC bytes.) If it was tampered with, it branches to the bomb. If it was okay, it executes the code at \$B863. It then checks the result, and if it was okay, branched to \$E90. No problem. Just patch up the branches a little and it works. Right?

Wrong. Take sector editor out and search for D0 05 20 63 B8 F0 2F or as much as it as you can. Change the first 05 to 03, and change the F0 to 80. You could NOP the JSR \$B863, but it isn't necessary. Boot up the disk and... nothing. It locks up, but why?

Right about now, I was ready to give up, but I remembered the code that verified \$B800. Was there one to verify *this* as well? (Of course there is, but read on.) We trace the file PS3 (the one we just edited) up to it's point of origin and execution, PIRATES.SYSTEM. Here we find some similar code:

```

E9B LDA #00
E9D TAX
E9E EOR $0800,X
EA1 INX
EA2 BNE $0E9E          (-06)
EA4 EOR $0900,X
EA7 INX
EA8 CPX #BE
EAA BNE $0EA4          (-08)
EAC TAX
EAD BEQ $0EB9          (+0A)
EAF ... hang for eternity
    
```

Here we have some similar code that verifies the verify. They're a tidy bunch at MICROPROSE. So all we have to do here is make that BEQ a BRA. Use your sector editor to find D0 F8 AA F0 0A A9 2C 8D 36 23 or as much as you can search for. Change the first F0 to 80. Write it back and you are done!

Lastly, let's take a look at the catalog. Hmmm... We see a bunch of BASIC files. Actually, the bulk of the game that you play is in BASIC. No wonder it's so slow. Somehow, if you could compile it, I think it would screw up the game something severe. And if you are bored, look at the locations in the LIFE file. They are the memory locations that store your stats. And be sure to look in the PICK file.

A parting note. If you don't use their ProDOS file, the game re-boots as soon as it starts PIRATES.SYSTEM. Since this is no real problem, don't worry about it. Perhaps there is a reason other than copy-protection. And for IIGs users with a UNIDrive, the disk access goes MUCH faster.

*\*IIGs Softkey for...*



### ■ Requirements

- Apple IIGs with ROM 01
- Copy II Plus or a ProDOS file copier
- Two Formatted UNIDisks (one with BASIC.SYSTEM on it)

The technique to crack this was very similar to Pirates! IIe/IIc. In fact, with the exception of the graphics and sound, the program code is near identical. And still in BASIC.

First, format 2 disks, one of them should have the volume name of /PIRATES and should have BASIC.SYSTEM on it. Re-name BASIC.SYSTEM to PIRATES.SYSTEM. Enter the monitor with CALL -151. At the monitor (\*) prompt, type # to get the CDA utilities.

Boot PIRATES! and listen to the drive. After 1 or 2 clicks, press **[ESC]** and select "Visit Monitor". Enter "\4C 00 20\<800.1FFFF" to scan for the JMP (4C 00 20) to start ProDOS. I found mine at \$9CC. Replace it with 4C 69 FF to enter the monitor instead of starting. Press **[Y]** and **[RETURN]** to quit and let the program execute further. In a moment, you will see the monitor prompt. Remove PIRATES! and insert the disk with the re-named BASIC.SYSTEM file on it. Type 20000G and you will have their ProDOS and your BASIC. Insert PIRATES! and type CATALOG to verify that this worked. If not, try again.

Now, either start COPY II PLUS with the -UTIL.SYSTEM command or start a ProDOS file copier. Be careful not to boot another ProDOS in the process or what you just did will be destroyed. File copy all the files from the PIRATES! master disk onto your own blank disk. This will take quite a while as there are over 100 files to be copied.

Lastly, whip out ye old block editor and search for 8D 20 B1 A2 00 and replace the 8D 20 B1 with 60 EA EA. The EA's aren't necessary, but I like to be neat. If you don't have a block editor, copy the file PS2 from the backup of PIRATES! you just made, edit it, and copy it back. Now you can enjoy a vastly improved version of PIRATES!

Also, for the novice hacker, here are a few tips that I constantly use. This is the first piece

of IIGs software I ever cracked and they were most helpful:

1. A 140K floppy or an 800K UNIDisk is a lot of info to search through. Try to determine where the protection ISN'T before trying to find where it is. This can be accomplished in many ways:

A. If there are files, determine in what sequence that they are loaded. Look for text within the startup file to see what files it loads. (PS2 was a file loaded from PIRATES.SYSTEM).

B. Plant "bombs" in key locations to see how far the code gets, if it ever gets there at all. Good ones are 4C 00 C6 (re-boot), 4C 69 FF (enter monitor) and 20 3A FF (beep).

2. Things to look for: 20 ED FD (print to the screen), AD 00 C0 (strobe keyboard), BD 8C C0 (disk access), xx 30 C0 (sound), xx 50 C0 (hi-res or double hi-res graphics), and xx 51 C0 (text screen).

I hope that these common things to look for are of some help to those new to hacking.

☺ Does anyone have any hints on how to crack or even copy ALIEN MIND for the IIGs from PBI. Shelling out \$57 for a game makes one a little uneasy thinking of all the nasty things that could happen to it. Kindly add it to the Wanted List. Also, if anyone can copy TOMAHAWK, or better yet, crack it, please let the rest of the uninformed world know of your marvel. Please add it to the Wanted List too.

Brian A Troha

First of all I would like to say to the following people:

**Jack R. Nissel** (COMPUTIST #56, page 20): Realm of Impossibility DOES work on the IIGs, it's EA's protection routine that does not. See COMPUTIST #50, page 20 and also COMPUTIST #49, page 16 for information on how this is done. A salesman friend has asked me to crack several EA programs for him because a customer couldn't get them to run on a IIGs. However, the cracked versions run just fine. Keep in mind: there is a second disk check at \$569 (and \$5F8 or \$5FE) that should be looked into. Some of the EA programs need the JMP \$569 to shut the disk drive off at the end of a disk read, so I edit the calls to \$5F8 (or \$5FE).

**Stephen J. Scalia** (COMPUTIST #57, page 27): It's too bad you feel your subscription wasn't as informative as you were looking for. I started my subscription with COMPUTIST #6, at which time I only knew BASIC programming skills and didn't know anything about cracking software. I still remember the excitement I felt when I cracked my first

program ALL BY MYSELF! It was PACMAN (and DIG DUG at the same time) and I used DEMUFFIN plus. Then I thought what I had done was simply tried one person's steps for another program, with NO REAL effort on my part. Since then, I have learned so many different steps that I can adapt to most protection schemes to overcome them. Now I can write those nifty articles I used to read and think "Boy this guy is really smart, I wish I could do that!"

That was about 4 years ago, it has taken a lot of work to get where I am today. Although I don't think you (Stephen J. Scalia) will ever end up reading this, but I think COMPUTIST will give each reader exactly what they put into it. If you really want to learn about disk protection read ALL the articles making sure you try to understand the longer ones as they have lots of information in them. I have learned so much from reading COMPUTIST (and of course other mags), I am glad I'm able to return the favor and send in articles for other readers to learn from.

Now on to the softkeys:

### Fix for EDD 4

For some odd reason there is a "lone byte" that seems to affect the way EDD 4 runs on a IIGs. Although I have not used EDD 4 in 2 or 3 years (softkeys do a much better job!) I was able to find the bad byte. To fix your copy of EDD 4 try this:

**CALL -151**  
**BLOAD EDD 4** *or what ever you called it*  
**AA60.AA73**

Write down the bytes found at \$AA61 & \$AA60 together and at \$AA73 & \$AA72 together in that order. These are the length and starting address for binary files under DOS 3.3 only. Now type:

**1F4E:EA** *was 02*  
**BSAVE EDD 4, A\$xxxx, L\$yyyy**

Where xxxx is the two byte hex number you found at \$AA73 & \$AA72, and yyyy is the number from \$AA61 & \$AA60.

That should take care of any problems people are having with EDD 4 and my article from COMPUTIST #49.

*\*IIGs Softkey for...*

### Softswitch

Roger Wagner

#### ■ Requirements

- Apple IIGs 512K
- 3½" disk copier
- 3½" disk block editor

The Softswitch (SS) program put out by Roger Wagner Publishing Inc (RWP) could be a very useful program if you worked with many

non-IIGs specific programs. The program allows you capture up to three programs in your IIGs's extra memory. Although I have not used the program that much, I could see where it might come in handy. However I found a few things about the program I didn't like, and one thing that is FLAT OUT WRONG!

First the program is supposed to be a classic desk accessory (CDA) that can be accessed from the control panel. RWP did not follow the correct format for CDAs, a CDA is type \$B8 (not type \$B6) and should be found in the DESK.ACDS subdirectory and not in the TOOL.SETUP subdirectory.

Lastly the program is copy protected, but in a form that must be illegal in some way, if it's not illegal it should be! The program disk has two unformatted tracks, leaving twenty-four blocks unusable, but that's not that bad. The real problem occurs when the disk protection passes! The program proceeds to alter a byte in the BATTERY BACKED UP RAM (BBR). RWP has NO RIGHT to physically alter anything in my computer that would not be changed back when the program is exited. The byte at E1/03BB is changed from a FF to a FE, then the program checks for it like this:

```
A9 FB 00 LDA #00FB $2C0+FB = 3BB
DA PHX
48 PHA
A2 03 0C LDX #0C03 Tool call ±30C, read a byte from
BBR
22 00 00
E1 JSL E10000 Tool locator call
68 PLA
29 03 00 AND #0003
4A LSR
F0 02 BEQ +02 Never taken
90 43 BCC +43 Clear carry = pass
```

Since the Branch on Equal is never taken, I changed it to EA 18 (NOP and CLC) so the next branch is always taken. Let them mess up their own computers if they want, but leave mine alone! When Apple Inc. reserved this memory I hardly think they reserved it for RWP. What if 100 people wrote programs that did this? This type of protection only breeds incompatibility down the line. Well I did find a way around this protection and a way to change your computer back to it's original state. To unprotect the file "TOOL.SETUP.2" make the following block edit (on a copy only):

Block	Byte(s)	From	To
\$111	\$1C7	F0 02	EA 18

Using the book "Inside the Apple IIGs" by Gary Bond I was able to look up all the values in the BBR and check for the altered byte. The book also gives the tool call number for reading and writing to the BBR. There is also a call that sets the BBR checksum if you change one of the bytes directly. So I searched the disk for each of these and found both parts of the

protection, the part that changes the byte and the second part that checks for it. If you want your computer's BBR in it's original state enter the following (the back-slashes are very important):

```
\04 00 00 FF 00 FB 0B 03\U
```

This is a tool call to WriteBParam from the monitor, this call changes the byte back and updates the four byte checksum of the BBR. Now your IIGs should have the BBR in it's original state and SS should still work.

This is an INCOMPLETE softkey, the install feature will not work on a copy of the Softswitch disk without the error, if anyone has the fix for that part of SS please let us know about it by writing in. Although I had this incomplete crack about a year ago I didn't want to send it in unless I had the whole thing. Well I have don't have enough time to work with it so I am asking for help from some of the other readers that have this program.

*\*IIGs Softkey for...*

### Mavis Beacon Teaches Typing

Software Toolworks

#### ■ Requirements

- 512K Apple IIGs
- 3½" disk copier
- 3½" disk editor
- 2 blank disks

Mavis Beacon Teaches Typing (MBTT) is a typing tutor with a very neat race car typing game. The program comes on two 3½" disk with the program disk having an unformatted track (track \$4F, side 2) which leaves blocks \$63A through \$63F unusable. Below I will explain the (long) way I tracked everything down.

First I started by making a copy of both disks using Copy II Plus v8. The program disk gave read errors for blocks \$63A through \$63F, while the data disk would copy without any errors. Going to the 3½" bitcopy portion of Copy II Plus I used the sector editor to scan the disk for 22 A8 00 E1 22 which is the code for a ProDOS 16 block read. I found this string once on block \$3E7. Once I located this code I checked it out and found it gets some information, makes the block read, stores a value at \$A15, then returned.

I traced back to the first RTL (6B) before the code and changed the three bytes after that to 68 FA 00; which is PLA, PLX, BRK (this is what I call hitting the BRAKES). I wrote the block back to the disk and booted it, when the program crashes (press **⌘**) to get to the text page/mode) you can see where the program jumped from. The Accumulator will hold the address (less the length of the instruction) and

# Readers Data Exchange

the X-register (10 byte) will hold the memory block.

Using this method it's easy to tell the call was made from \$03/03EE. When checking this new code you will notice the following things: The routine runs from \$03/03A0 thru \$03/04B9. The call made to \$03/D5FB (made once) is to a routine that gets the device number to use in checking for the bad blocks. The calls made to \$03/D5BB (made three times) is to the actual block read routines. After each call is something like (found on block \$37F):

```
LDA 0A15
CMP #0000 No error (either device number or block read)
BEQ CONTINUE
LDA #0000 Load Accum with failed value
BRL FAILED
```

## CONTINUE code

```
LDA #0001 Load Accum with passed value
TAY This is where the 'BRL FAILED' comes to
CLC
ADC #00F7
TCS
TYA
PLD
RTL Finally return to sender
```

## I made the following changes:

Block	Byte(s)	From	To
\$37F	\$2A	22	AF
	\$33	C9 00 00 F0 06	EA EA EA EA EA
		A9 00	A9 01
	\$7B	22	AF
	\$84	C9 00 00 F0 06	EA EA EA EA EA
		A9 00	A9 01
	\$A9	22	AF
	\$B2	C9 00 00 D0 05	EA EA EA EA EA
		A9 00	A9 01
	\$D6	22	AF
	\$DF	C9 00 00 F0 05	EA EA EA EA EA
		A9 00 00 80 03	EA EA EA EA EA

This works fine (producing a cracked version), but I thought there was a better way using less edits. Again applying the BRAKE trick I was able to back-trace the protection one more step and came up with the following code at \$03/6C8E thru \$03/6CC1:

```
6C8E:0B PHD
6C8F:3B TSC
6C90:38 SEC
6C91:E9 FB 00 SBC #00FB
6C94:5B TCS
6C95:69 F6 00 ADC #00F6
6C98:1B TCS
6C99:F4 3A 06 PEA 063A First bad block to look
for
6C9C:F4 01 00 PEA 0001
6C9F:22 A0 03 03 JSL 0303A0 Call to the code
described above
6CA3:7A PLY
6CA4:7A PLY
```

```
6CA5:A8 TAY
6CA6:D0 12 BNE 6CBA (+12) Not zero, then
continue
6CA8:F4 17 00 PEA 0017
6CAB:F4 01 00 PEA 0001
6CAE:F4 C5 64 PEA 64C5
6CB1:22 4C 3A 04 JSL 043A4C Goto insert master
routine
6CB5:7A PLY
6CB6:7A PLY
6CB7:7A PLY
6CB8:80 DF BRA 6C99 (-21) Go back and do
check again
6CBA:7B TDC Must be an original
6CBC:18 CLC
6CBD:69 FB 00 ADC #00FB
6CBF:1B TCS
6CC0:2B PLD
6CC1:6B RTL Return to sender
```

To bypass this step I changed the JSL \$03/03A0 to LDA \$03/03A0 and the BNE \$6CBA to BRA \$6CBA. The edits to the disk are:

Block	Byte(s)	From	To
\$3B3	\$93	22	AF
	\$9A	D0	80

The above edit will produce a completely cracked version of MBTT as did the first edit I showed. The routine at \$03/6C8E seemed very short and didn't return any special values so I tried the BRAKE trick one more time and was rewarded with the following (at \$04/957C):

```
957C:7A PLY
957D:8E 79 09 STX 0979
9580:8D 77 09 STA 0977
9583:22 8E 6C 03 JSL 036C8E Goto the start of the
protection
9587:F4 00 00 PEA 0000
958A:22 F1 6B 03 JSL 036BF1
```

There you have it, three different ways to crack MBTT at three different levels. Once again this is an example that shows there are many ways to do the same thing. To softkey MBTT simply follow these easy steps:

**1** Make copies of both disks (ignore errors on blocks \$63A-\$63F on the program disk)

**2** Make the following change to the copied program disk:

Block	Byte(s)	From	To
\$4B9	\$197	22	AF

NOTE: You could also apply the other edits, but they are not needed.

**3** Write the block back to the copy. Store the originals in a safe place.

Optional: Upload the program to a hard drive.

## \*Iigs Softkey for...

### Magical Myths

Unicorn Inc.

#### Requirements

- 768K Apple Iigs
- 3 1/2" disk copier
- 3 1/2" disk editor
- Two blanks 3 1/2" disks

Magical Myths (MM) is an educational program that teaches Greek mythology along with teaching basic reading and reading comprehension skills. The program makes use of limited speech and good graphics with mouse control. The program comes on two 3 1/2" disks and uses nibble counting (on tracks \$20 and \$21) on the program disk for copy protection. Magical myths is published by the same company that puts out Aesop's Fables and uses almost the exact same protection routine. Like the other Unicorn program if you search for A2 20 A0 01 you would find the copy protection. The code is on block \$1EE and it looks like:

```
14B:E2 30 SEP #30 8 bit Accum
14D:A2 21 LDX #21 Track $21
14F:A0 01 LDY #01 on side one
151:20 3F 6A JSR 6A3F Check for the original
154:80 3B BCS 191 (+3B) On error goto 191
156:E0 B0 CPX #B0
158:98 TYA
159:E9 1D SBC #1D Compare to 1DB0
15B:90 2C BCC 189 (+2C) Too low?, goto 189
15D:E0 78 CPX #78
15F:98 TYA
160:E9 1E SBC #1E Compare to 1E78
162:B0 29 BCS 18D (+29) Too high?, goto 18D
164:A2 20 LDX #20 Track $20
166:A0 01 LDY #01 on side one
168:20 3F 6A JSR 6A3F Check for original
16B:80 24 BCS 191 (+24) On error goto 191
16D:E0 40 CPX #40
16F:98 TYA
170:E9 1F SBC #1F Compare to 1F40
172:90 15 BCC 189 (+15) Too low?, goto 189
174:E0 08 CPX #08
176:98 TYA
177:E9 20 SBC #20 Compare to 2008
179:B0 12 BCS 18D (+12) Too high?, goto 18D
17B:A9 00 LDA #00 Everything passed
17D:48 PHA
17E:22 0A 6B 00 JSL 006B0A Print 'PLEASE WAIT'
182:A9 00 LDA #00
184:EB XBA
185:68 PLA
186:28 PLP
187:AB PLB
188:6B RTL Return to sender
..... "Fail code starts here"
```

# Readers Data Exchange

If you change the LDX #21 (A2 21) at \$14D to BRA \$17B (80 2C) then MM no longer checks for the original and continues to load. The only thing left is to make the change permanent. Try these easy steps:

- 1 Copy the 3 1/2" disk
- 2 Make the following edits to the copy:

Block	Byte(s)	From	To
\$1EE	\$14D	A2 21	80 2C

- 3 Write the block back the copy.

\*Iigs Softkey for...

## Tales from the Arabian Nights

Unicorn Inc.

### Requirements

- 768K Apple Iigs
- 3 1/2" disk copier
- 3 1/2" disk editor
- Two blank 3 1/2" disks

Tales From the Arabian Nights (TAN) is an educational program that teaches old tales from the middle east along with teaching basic reading and reading comprehension skills. The program makes use of limited speech and nice graphics with mouse control. The program comes on two 3 1/2" disks and uses nibble counting (on tracks \$20 and \$21) on the program disk for copy protection. TAN is published by the same company that puts out Aesop's Fables and Magical Myths and also uses the same protection routine. Like the other two Unicorn programs, if you search for A2 20 A0 01 you would find the copy protection. The code is on block \$553 and it looks like:

```

1B8:E2 30 SEP #30 8 bit Accum
1B0:A2 21 LDX #21 Track $21
1B8:A0 01 LDY #01 on side one
1C1:20 AB 6A JSR 6AAB Check for the original
1C4:80 3B BCS 1FF (+3B) On error goto 1F9
1C6:E0 B0 CPX #B0
1C8:98 TYA
1C9:E9 10 SBC #1D Compare to 1DB0
1CB:90 2C BCC 1F9 (+2C) Too low?, goto 1F9
1CD:E0 78 CPX #78
1CF:98 TYA
1D0:E9 1E SBC #1E Compare to 1E78
1D2:80 29 BCS 1FD (+29) Too high?, goto 1FD
1D4:A2 20 LDX #20 Track $20
1D6:A0 01 LDY #01 on side one
1D8:20 AB 6A JSR 6AAB Check for original
1DB:80 24 BCS 1FF (+24) On error goto 1F9
1DD:E0 40 CPX #40
1DF:98 TYA
1E0:E9 1F SBC #1F Compare to 1F40
1E2:90 15 BCC 1F9 (+15) Too low?, goto 1F9
1E4:E0 08 CPX #08
1E6:98 TYA
    
```

```

1E7:E9 20 SBC #20 Compare to 2008
1E9:80 12 BCS 1FD (+12) Too high?, goto 1FD
1EB:A9 00 LDA #00 Everything passed
1ED:48 PHA
1EE:22 76 6B 00 JSL 006B76 Prim 'PLEASE WAIT'
1F2:A9 00 LDA #00
1F4:EB XBA
1F5:68 PLA
1F6:28 PLP
1F7:AB PLB
1F8:6B RTL Return to sender
..... "Fail code is here"
    
```

If you change the LDX #21 (A2 21) at \$1BD to BRA \$1EB (80 2C) then TAN no longer checks for the original and continues to load. The only thing left is to make the change permanent. Try these easy steps:

- 1 Copy the 3 1/2" disk.
- 2 Edit the copy.

Block	Byte(s)	From	To
\$553	\$1BD	A2 21	80 2C

- 3 Write the block back the copy.

\*Iigs Softkey for...

## King's Quest I & II

Sierra On Line

### Requirements

- 512K Apple Iigs
- 3 1/2" disk copier
- 3 1/2" disk editor with search ability

King's Quest I and II (KQ) have been re-released in a Iigs format, with the same type of cartoon graphics as the 128K Apple II versions. It is sad that Sierra has overlooked the potential of the Iigs, they should have put some effort into the graphics. Well, to continue, the protection is the same as on Space Quest and Leisure Suit Larry. The program comes on two 3 1/2" disk, with both disks having an error over block \$634. After checking the code on both KQ disks, and that of the other two Sierra programs I found a common section of code (along with the edit to crack them). All four disks have this same code (on the disk it looks like):

```

22 00 00 00 JSL 000000
C9 01 00 CMP 0001
F0 04 BEQ (+4)
22 00 00 00 JSL 000000
22 00 00 00 JSL 000000 "22 MUST be chged to AF"
AE 00 00 LDX 0000
AD 00 00 LDA 0000
DA PHX
48 PHA
    
```

These bytes can be found on block \$A0 for KQ1. The patch makes a cracked copy.

However when the patch is used (on block \$2ED) for KQ2, the program would load about 80% then bomb into the monitor. Now you must go back and trace the actual block read code and look for some type of flag. After searching for 34 06 (the number of the bad block) I found it twice on block \$2C4. Checking this code revealed two values (00 02 27 00) are stored in consecutive memory. The value 00 27 is the code returned for a block read error by the protocol converter and ProDOS 16. So searching for C9 27 would reveal the this "check of the disk check" on block \$267. If the 02 27 wasn't found then a JSL was made to some subroutine, changing this to a LDA results in a cracked version of KQ2!

If you use Copy II Plus v8 to search for 00 F0 04 22 00 00 00 22, when the sequence is found, the cursor will be on the byte the needs to be changed. The same exact sequence is on the four (4) forementioned programs and the same change will work for each, however King's Quest II does require an addition patch. When checking a copy of Police Quest (also by Sierra) I could not find the above code or any protection routine for that matter. The softkey for both KQ is as follows:

- 1 Make a copy of all the disks, ignore errors on block \$634.
- 2 Make the following change (on the copy of disk one only):

For King's Quest - Quest for the Crown:

Block	Byte(s)	From	To
\$A0	\$BB	22	AF

For King's Quest II - Romancing the Throne:

Block	Byte(s)	From	To
\$267	\$166	22	AF
\$2ED	\$BF	22	AF

- 3 Write the block back to the copy.

Optional: Upload the whole program to a hard disk. Store the Original in a safe place.

Softkey for...

## Defender of the Crown

Cinemaware Inc.

### Requirements

- 768K Apple Iigs
- 3 1/2" disk copier
- 3 1/2" disk editor

Defender of the Crown (DOTC) is a GREAT new program from Cinemaware. This is the first program in a whole line of "Playable Movies" (Cinemaware even calls the disks "reels") ported from the Amiga and the Mac.

While these programs are memory hungry due to the amount of graphics and sound used, these new programs set a new level of quality and hopefully a new standard for the IIGs. The object of the game in DOTC is to gain control of the whole region by waging war in the eleven hundreds. DOTC does live up to its reputation with great graphics and music, the game also uses the mouse for control which makes for easy playability.

The program comes on two 3 1/2" disks and uses nibble counting on tracks \$20 and \$21 of disk/reel one. So again, we encounter that same "old" commercial disk protection scheme. If you get out your disk searching utility and search for A2 20 A0 01 or A2 21 A0 01 you will find the offending code (on block \$258). The code should look like this (as it is on the disk):

```

03:F4 00 00 PEA 0000
06:F4 00 00 PEA 0000
09:A2 0C 22 LDX #220C
0C:22 00 00 E1 JSL E10000
10:68 PLA
11:9C E1 17 STZ 17E1 Store a pass value in the flag
14:20 E3 17 JSR 17E3 Go to the protection routine
17:AD E1 17 LDA 17E1 Load the flag
1A:F0 03 BEQ 1F (+03) If zero then continue
1C:4C 64 17 JMP 1764 Goto INSERT MASTER msg
1F:22 72 9C 00 JSL 009C72
23:4C 26 18 JMP 1826 Go to the start of the program
26:00 00 0000 This is 17E1 when loaded in
28:08 PHP Start of the protection routine
29:E2 30 SEP #30 8 bit Accumulator
2B:A2 20 LDX #20 Track $20
2D:A0 01 LDY #01
2F:20 FF 17 JSR 17FF Count them up
32:B0 0B BCS 3F (+0B) carry set = failed
34:A2 21 LDX #21 Track $21
36:A0 01 LDY #01
38:20 FF 17 JSR 17FF Count them up
3B:B0 02 BCS 3F (+02) Carry set = failed
3D:28 PLP
3E:60 RTS Return to sender
3F:EE E1 17 INC 17E1 Failed so mess up the flag
42:28 PLP
43:60 RTS And return to sender
44:8E 1E 18 STX 181E Start of 'Count them up'
47:8C 1F 18 STY 181F
4A:5A PHY
4B:DA PHX Push track number
4C:F4 00 00 PEA 0000
4F:F4 21 18 PEA 1821 Push more data.
52:22 8F 9C 00 JSL 009C8F To actual count routine.
56:8D 20 18 STA 1820 Store returned value
59:68 PLA
5A:68 PLA
5B:68 PLA
5C:68 PLA
5D:68 PLA
5E:68 PLA Pull all extra values
5F:AD 20 18 LDA 1820 Get the returned value
62:60 RTS Return to sender
    
```

This one seems real easy to crack, just think about it. First the code stores a zero in the flag and goes to the nibble count, then when it returns checks the flag for a zero. If you change the JSR (20) at 14 to a LDA (AD) the code stores a zero in the flag, loads a value and then checks the flag for a zero! You could also change the BEQ (F0) at 1A to a BRA (80) just for the sake of completeness, although it's NOT needed. If for some reason you cannot find the above code on block \$258 you could search your disk for 9C E1 17 20 E3 17 ADE1 17 and change the 20 to AD. To crack Defender of the Crown follow these easy steps:

- 1 Make copies of both DOTC disks.
- 2 Make the following edits on a copy of disk 1.

Block	Byte(s)	From	To
\$258	\$14	20	AD

- 3 Write the block back to the disk.

Optional: Upload the whole program to a hard disk.

Softkey for...

## Adventures of Sinbad

Unicorn Inc.

### ■ Requirements

- 768K Apple IIGs
- 3 1/2" disk copier
- 3 1/2" disk editor
- Two blanks 3 1/2" disks

Adventures of Sinbad (AOS) is an educational program that uses Sinbad's adventures to teach basic reading and reading comprehension skills. The program comes on two 3 1/2" disks and uses nibble counting (on tracks \$20 and \$21) on the program disk for copy protection. Like the other Unicorn program if you search for A2 21 A0 01 you would find the copy protection. The actual disassembly is so close to other Unicorn products I didn't think it would be worth repeating.

If you change the LDX #21 (A2 21) at 103 to BRA 131 (80 2C) then AOS no longer checks for the original and continues to load. The only thing left is to make the change permanent. Try these easy steps:

- 1 Copy the 3 1/2" disk.
- 2 Make the following edits to the copy:

Block	Byte(s)	From	To
\$47A	\$103	A2 21	80 2C

- 3 Write the block back the copy.

Edward Teach

While I've got your attention, I've got something to say. In issue #56 (page 29) the RDEXed states that there are about 6000 subscribers to COMPUTIST. My question is, What are the rest of you doing? I have been a subscriber since last September. In the past year I have submitted several articles. My first article was one entire page on how to find a jump to \$C600. It wasn't the best work that I have ever done, but at least I contributed.

But what really bothers me, are the people who give 38 softkeys by saying change byte A to B and don't give any indication how they found what to change. If you want to know how to write, look at the articles by Brian Troha, Jim Hart and Charles Taylor. Even on the simple cracks they all still found something "extra" to say about the disk.

I also got a friend to subscribe and he got another friend to subscribe. This is why I can't understand why we have a MOST WANTED LIST. With 6,000 people, out there, why can't we get together and overcome a few protected disks? And boost the number of subscribers in the process. I hope I did not offend the regulars, and to the other 5000+ it won't matter... you never write anyway.

### DOUBLE DOS

I was working on a softkey for a disk that has already appeared in several past issues of COMPUTIST. The other cracks all poked an 18 into \$B942 and copied the disk. A few sector edits later the COPYA disk was finished. On my version there were 11 different address prologues and two different data prologues. Simply poking \$B942 wasn't going to work. My first attempt was to write 11 different controllers and at least get a COPYA version for experimentation. I did manage to make the copy, but a large amount of the code was 'junk'. Apparently the disk was encoded. After some searching I discovered that the read translate table was altered. Copying the copy, using their read table, produced a disk with good code that died horribly during booting. You have probably noticed that there are not 11 controllers at the end of this article. Filling the pages of COMPUTIST with all of these did not seem to be the best way to present my program.

While trying to get around this problem I remembered several other items that I have worked on. (Broderbund and Electronic Arts releases). These disks are all protected in the same way. A table of header bytes is constructed then when a "read" is performed the headers are located in the table and placed in DOS. Using this theory all I had to do (ALL?) was rewrite the DOS read routine, construct header tables and place the header

# Readers Data Exchange

information in the new read code. The premise of the code is to read drive 1 with the protected DOS and drive 2 with DOS 3.3. This is my first attempt at writing assembly code, so if it appears a bit cryptic, bear with me. The heart of the code is DOUBLE.OBJ:

```

8000: LDA #$18      POKE normal
8002: STA $B989     DOS to
8005: STA $B992     ignore
8008: STA $B99C     errors.
800B: LDA #$60
800D: STA $B98A
8010: STA $B993
8013: STA $B99D
8016: LDX #$00     Move $B944
8018: LDA $B944,X to
801B: STA $809A,X $809A
801E: INX
801F: CPX #$5C
8021: BNE $8018
8023: JMP $8055     Jump to next move
8026: STY $9000     Store X,Y,A
8029: STA $9001
802C: STX $9002
802F: LDA $35      Get drive #
8031: AND #$80     turn off bits
8033: BMI $8052    DR=1?/then jump
8035: LDA #$D5
8037: STA $B8E7     store normal
803A: STA $80AB     values in
803D: LDA #$AA     DOS
803F: STA $B8F1
8042: STA $80B5
8045: LDA #$96     * DOS *
8047: STA $80C0     * DIVERTER *
804A: LDA #$AD
804C: STA $B8FC
804F: JMP $808C     normal read
8052: JMP $8065     protected read
8055: LDX #$00
8057: LDA $8026,X Move code from
805A: STA $B944,X $8026
805D: INX
805E: CPX #$30     to
8060: BNE $8057     $B944
8062: NOP
8063: NOP
8064: RTS
8065: LDX $0478     get track number
8068: LDA $80F6,X (D5) table 1
806B: STA $80AB     new read code
806E: LDA $811F,X (AA) table 2
8071: STA $80B5     new read code
8074: LDA $8148,X (96) table 3
8077: STA $80C0     new read code
807A: LDA $8171,X (D5) table 4
807D: STA $B7E7     normal DOS
8080: LDA $819A,X (AA) table 5
8083: STA $B8F1     normal DOS
8086: LDA $81C3,X (AD) table 6
8089: STA $B8FC     normal DOS
808C: LDY $9000     restore Y
808F: LDA $9001     restore A
8092: LDX $9002     restore X

```

```

8095: JMP $809A     jump to new read
8098: CLC
8099: 60

```

This code modifies the normal DOS read routine (\$B944-\$B99F) then relocates it to \$809A. Next the diverter section is moved (from \$8026-\$8052) to replace the \$B944 code. Finally six tables are created (or at least identified) that will hold the header information. Now when DOS tries to read the disk our diverter will determine the drive being accessed and route the code to either the 'normal read' or the 'protected' read. The new \$B944 code saves all the registers, and reads memory location \$35. The current drive number is stored as the high bit of that memory location. If the bit is not set (1) then drive one was the last accessed, if set (0) then drive two was the last disk accessed. This would appear to be backwards. (1 = not set, 0 = set). Anyway, the code either jumps to the read table code at \$8065, or pokes the normal 'D5AA96/D5AAAD' into DOS and then jumps to our relocated read routine.

Just in case I lost the beginners, what all this means is the code will now read whatever headers you tell it to on drive one (D4AA96 for example, or even 9796EB). DOUBLE DOS can handle up to eighty different headers on a 5.25' disk. Also, if you had any trouble writing controllers, now the simplist controller will function without problem. No more trying to write controllers that read one track with one header and the next track with a different header. The tables relate to the six header bytes:

Table #	Byte	Start of Table
1	D5	\$80F6
2	AA	\$811F
3	96	\$8148
4	D5	\$8171
5	AA	\$819A
6	AD	\$81C3

Each table is forty bytes long to accomodate 40 tracks per disk. If the disk has only 35 tracks the program will still function. It just ignores the last few tracks.

The Applesoft part of Double DOS serves as an easy method to input the tables. The majority of the code is for input. The code builds six strings J\$(X) (X=1 to 6), converts the strings to hex bytes and pokes them into memory. I tried to make this as painless as possible. However, writing a full screen editor is less than fun. Other than the four arrow keys for cursor movement the only other keys that you need to know are **[ESC]** to exit the editor, and **[CW]** to write the new tables to memory. After entering all the tables (you can skip the ones you don't need) and pressing **[CW]**, drive one will no longer be able to read DOS 3.3. Any program that you need to load MUST be loaded from drive two. *Do not forget to remove the disk from drive two.* A write protect tab is

also a good idea.

As a final note we need to modify SUPER IOB 1.5 so only 5 tracks are read at one time. Reading more overwrites the code we just entered. Do the following to the SUPER IOB:

```

LOAD SUPER IOB 1.5
10075 IF PEEK(47428) = 140 THEN MB =
      110: GOTO 10085
10080 MB = 151
10085 HOME: DOS = 16: HXS = "00010203040
      506070809101112131415161718191A1B1C
      1D1E1F202122"
SAVE SUPER IOB 1.5

```

This change will peek in memory to see if Double DOS has been installed. If it has then the MB parameter is lowered to copy only five tracks, if not then the normal seven tracks will be copied as usual. This change is completely transparent to you when running the program and once entered will automatically handle my program when ever it is in memory. To find the proper bytes to store in the tables you will need a pencil, paper and a nibble reader. (I use the 'N' function of Locksmith 6.0). Nibble read track \$00 and write down the Address and Data prologue bytes. Then repeat the process for the entire disk. Now, run Double DOS and enter any changes into the correct tables. If, for instance, all the first bytes of the Address are the normal 'D5's then skip table one and go on to table two, if the second bytes are all 'AA's then table three...

I know that this is a lot to key in, however I find it a useful tool in deprotecting programs.

## DOUBLE DOS

```

10 REM SET UP VARIABLES
20 DIM H$(40)
30 F$(1) = "D5" : F$(2) = "AA" : F$(3) = "96"
   : F$(4) = "D5" : F$(5) = "AA" : F$(6) = "AD"
40 B$(1) = "D5^AA^96" : B$(2) = "D5^AA^96" : B$(3)
   = "D5^AA^96" : B$(4) = "D5^AA^AD" : B$(5) =
   "D5^AA^AD" : B$(6) = "D5^AA^AD"
50 X = 1
60 J$(1) = "D5" : J$(3) = "96" : J$(6) = "AD"
70 J$(1) = J$(1) + "D5" : IF LEN (J$(1)) < 80 THEN
   70
80 J$(3) = J$(3) + "96" : IF LEN (J$(3)) < 80 THEN
   80
90 J$(6) = J$(6) + "AD" : IF LEN (J$(6)) < 80 THEN
   90
100 J$(2) = J$(2) + "AA" : IF LEN (J$(2)) < 80 THEN
   100
110 J$(4) = J$(1) : J$(5) = J$(2)
120 POKE 47426,24
130 REM BLOAD DOUBLE.OBJ AND CREATE SIX TABLES
140 PRINT CHR$(4) "BLOAD^DOUBLE.OBJ"
150 CALL 32768
160 FOR X = 33014 TO 33054: POKE X,213: NEXT
170 FOR X = 33055 TO 33095: POKE X,170: NEXT
180 FOR X = 33096 TO 33136: POKE X,150: NEXT
190 FOR X = 33137 TO 33177: POKE X,213: NEXT

```

# Readers Data Exchange

```

200 FOR X = 33178 TO 33218: POKE X,170: NEXT
210 FOR X = 33219 TO 33259: POKE X,173: NEXT
220 X = 1
230 CALL - 936
240 REM SET UP INPUT SCREEN MASK
250 VTAB 1: HTAB 14: INVERSE: PRINT "DOUBLE'DOS"
  : NORMAL
260 VTAB 3: HTAB 09: PRINT "WRITTEN'BY'EDWARD'
  TEACH"
270 VTAB 5: HTAB 6: PRINT "00-*****0A-*****14-^
  *****1E-^"
280 VTAB 6: HTAB 6: PRINT "01-*****0B-*****15-^
  *****1F-^"
290 VTAB 7: HTAB 6: PRINT "02-*****0C-*****16-^
  *****20-^"
300 VTAB 8: HTAB 6: PRINT "03-*****0D-*****17-^
  *****21-^"
310 VTAB 9: HTAB 6: PRINT "04-*****0E-*****18-^
  *****22-^"
320 VTAB 10: HTAB 6: PRINT "05-*****0F-*****19-^
  *****23-^"
330 VTAB 11: HTAB 6: PRINT "06-*****10-*****1A-^
  *****24-^"
340 VTAB 12: HTAB 6: PRINT "07-*****11-*****1B-^
  *****25-^"
350 VTAB 13: HTAB 6: PRINT "08-*****12-*****1C-^
  *****26-^"
360 VTAB 14: HTAB 6: PRINT "09-*****13-*****1D-^
  *****27-^"
370 VTAB 16: HTAB 15: PRINT B$(X)
380 VTAB 18: HTAB 15: PRINT "TABLE->" X
390 U = 5: I = 09
400 R = 1
410 FOR Q = 1 TO 10: VTAB U: HTAB I: PRINT MID$(
  J$(X),R,2):U = U + 1:R = R + 2: NEXT
420 U = 5: I = I + 8: IF I < 35 THEN 410
430 VTAB 20: HTAB 5: PRINT "DO' YOU' WANT' TO'
  CHANGE' THE' " : INVERSE: PRINT F$(X):
  NORMAL
440 PRINT
450 PRINT "*****Y/N^" : GET AS
460 IF AS = "N" THEN X = X + 1: IF X > 6 THEN X = 1
470 IF AS = "Y" THEN 510
480 IF AS = CHR$(23) THEN 780
490 GOTO 370
500 REM SCREEN EDITOR
510 H = 9: V = 5: C = 1: F(X) = 1
520 HTAB H: VTAB V: GET AS
530 IF AS = CHR$(21) AND ((H = 09) OR (H = 17) OR
  (H = 25) OR (H = 33)) THEN H = H + 1: C = C + 1:
  GOTO 670
540 IF AS = CHR$(21) AND ((H = 10) OR (H = 18) OR
  (H = 26) OR (H = 34)) THEN H = H - 1: C = C + 19
550 IF AS = CHR$(21) THEN H = H + 8: IF H > 34 THEN
  H = 9: C = C - 80
560 IF AS = CHR$(21) THEN 670
570 IF AS = CHR$(08) AND ((H = 10) OR (H = 18) OR
  (H = 26) OR (H = 34)) THEN H = H - 1: C = C - 1:
  GOTO 670
580 IF AS = CHR$(08) AND ((H = 09) OR (H = 17) OR
  (H = 25) OR (H = 33)) THEN H = H + 1: C = C - 19
590 IF AS = CHR$(08) THEN H = H - 8: IF H < 09 THEN
  H = 33: C = C + 79
600 IF AS = CHR$(08) THEN 670

```

```

610 IF AS = CHR$(11) AND ((H = 10) OR (H = 18) OR
  (H = 26) OR (H = 34)) THEN H = H - 1: C = C - 1
620 IF AS = CHR$(11) THEN V = V - 1: C = C - 2: IF
  V < 5 THEN V = 14: C = C + 20
630 IF AS = CHR$(11) THEN 670
640 IF AS = CHR$(10) AND ((H = 10) OR (H = 18) OR
  (H = 26) OR (H = 34)) THEN H = H - 1: C = C - 1
650 IF AS = CHR$(10) THEN V = V + 1: C = C + 2: IF
  V > 14 THEN V = 5: C = C - 20
660 IF AS = CHR$(10) THEN 670
670 IF AS = CHR$(08) OR AS = CHR$(21) OR AS = CHR$(
  10) OR AS = CHR$(11) THEN 730
680 IF AS = CHR$(27) THEN 430
690 HTAB H: VTAB V: PRINT AS
700 IF C = 1 THEN J$(X) = AS + RIGHT$(J$(X), LEN
  (J$(X)) - 1): GOTO 720
710 J$(X) = LEFT$(J$(X), (C - 1)) + AS + MID$(
  J$(X), (C + 1))
720 GOSUB 760
730 GOTO 520
740 X = X + 1: IF X > 6 THEN X = 1
750 GOTO 430
760 B = H: N = V
770 H = B: V = N: RETURN
780 X = 1: R = 1: Z = 33014
790 IF NOT F(X) THEN 950
800 FOR P = 1 TO 40
810 H$(P) = MID$(J$(X), R, 2)
820 R = R + 2
830 NEXT
840 REM CONVERT DEC TO HEX
850 R = 1
860 FOR E = 1 TO 40
870 S$ = LEFT$(H$(E), 1): T$ = RIGHT$(H$(E), 1)
880 IF S$ > "A" AND S$ < "F" THEN H = ASC(S$)
  - 55: GOTO 900
890 H = VAL(S$)
900 IF T$ > "A" AND T$ < "F" THEN M = ASC(T$)
  - 55: GOTO 920
910 M = VAL(T$)
920 NU = H * 16 + M
930 POKE Z + (E - 1), NU
940 NEXT
950 Z = Z + 41: R = 1
960 VTAB 23: HTAB 5: PRINT "INSTALLING'TABLE'
  " X
970 X = X + 1: IF X < 7 THEN 790
980 HOME: PRINT "DRIVE' ONE' WILL' NOW' READ'
  PROTECTED'DOS" : PRINT "DRIVE'TWO' WILL'
  NOW'READ'DOS'3.3" : PRINT "RUN'SUPER'IOB'
  1.5'FROM'DRIVE'TWO"
990 END

```

## Checksums

10	- \$BADD	510	- \$BC9A
20	- \$28C7	520	- \$6595
30	- \$3E0A	530	- \$57A2
40	- \$CA9A	540	- \$F603
50	- \$8D2B	550	- \$40FF
60	- \$5371	560	- \$7436
70	- \$BA15	570	- \$A0C7
80	- \$5378	580	- \$07D8
90	- \$AFD2	590	- \$4A93

100	- \$19D0	600	- \$FE1C
110	- \$F2C8	610	- \$50A3
120	- \$8C74	620	- \$711A
130	- \$7EC8	630	- \$409D
140	- \$BBED	640	- \$B3BD
150	- \$AE43	650	- \$2488
160	- \$EFE2	660	- \$F088
170	- \$0548	670	- \$FB1C
180	- \$EA95	680	- \$6660
190	- \$19AB	690	- \$FFC0
200	- \$8D7F	700	- \$F353
210	- \$354B	710	- \$C914
220	- \$188B	720	- \$59E8
230	- \$A730	730	- \$47C2
240	- \$BA1A	740	- \$736E
250	- \$2154	750	- \$BC5F
260	- \$40FF	760	- \$013D
270	- \$15BF	770	- \$9D88
280	- \$A68A	780	- \$E3BD
290	- \$BBA5	790	- \$3A93
300	- \$C672	800	- \$382A
310	- \$D53A	810	- \$6551
320	- \$CB2D	820	- \$46FB
330	- \$D96E	830	- \$5A89
340	- \$EF17	840	- \$E84E
350	- \$FB6F	850	- \$18ED
360	- \$8692	860	- \$96B7
370	- \$8926	870	- \$7EBA
380	- \$507D	880	- \$5FBB
390	- \$918B	890	- \$7D2A
400	- \$BC35	900	- \$3B16
410	- \$AA28	910	- \$F059
420	- \$9E70	920	- \$4176
430	- \$7518	930	- \$836F
440	- \$8861	940	- \$FE1D
450	- \$7007	950	- \$7739
460	- \$90D6	960	- \$6EBF
470	- \$C2DD	970	- \$FCE7
480	- \$18B0	980	- \$631C
490	- \$59FF	990	- \$8B47
500	- \$AB89		

## DOUBLE.OBJ

8000:	A9 18 8D 89 B9 8D 92 B9	\$082F
8008:	8D 9C B9 A9 60 8D 8A B9	\$800B
8010:	8D 93 B9 8D 9D B9 A0 00	\$7B71
8018:	BD 44 B9 9D 9A 80 E8 E0	\$2F71
8020:	5C D0 F5 4C 55 80 8C 00	\$51AF
8028:	90 8D 01 90 8E 02 90 A5	\$4499
8030:	35 29 80 30 1D A9 D5 8D	\$29F6
8038:	E7 B8 8D AB 80 A9 AA 8D	\$0831
8040:	F1 B8 8D B5 80 A9 96 8D	\$1AB9
8048:	C0 80 A9 AD 8D FC B8 4C	\$335F
8050:	8C 80 4C 65 80 A2 00 BD	\$67BA
8058:	26 80 9D 44 B9 E8 E0 30	\$F8A0
8060:	D0 F5 EA EA 60 AE 78 04	\$6E92
8068:	BD F6 80 8D AB 80 BD 1F	\$4B81
8070:	81 8D B5 80 BD 48 81 8D	\$D3A3
8078:	C0 8D 0D 71 81 8D E7 B7	\$CD4F
8080:	BD 9A 81 8D F1 B8 BD C3	\$CFE5
8088:	81 8D FC B8 AC 00 90 AD	\$6306
8090:	01 90 AE 02 90 4C 9A 80	\$F780
8098:	18 60 00 00	\$27A8

## Softkey for...

### Mathblaster

Davidson and Assoc.

This disk has had a few softkeys published over the last months. None of these would work on my disk. My copy had 11 different address prologues and two different data prologues. My first attempt at disk inspection is always to make a COPYA version. Making the copy will be alot easier if you enter these tables into Double DOS:

Table 1

00-D5	0A-DF	14-D5	1E-DF
01-D5	0B-DF	15-D5	1F-DF
02-D7	0C-DD	16-D7	20-F5
03-D7	0D-DD	17-D7	21-F5
04-D5	0E-DF	18-DD	22-F7
05-D5	0F-DF	19-DD	23-D5
06-D7	10-D5	1A-DF	24-D5
07-D7	11-D5	1B-DF	25-D5
08-DD	12-D7	1C-DD	26-D5
09-DD	13-D7	1D-DD	27-D5

Table 3

00-96	0A-9E	14-96	1E-9E
01-97	0B-9F	15-97	1F-9F
02-96	0C-9E	16-96	20-B6
03-97	0D-9F	17-97	21-B7
04-96	0E-9E	18-9E	22-B6
05-97	0F-9F	19-9F	23-96
06-96	10-96	1A-9E	24-96
07-97	11-97	1B-9F	25-96
08-9E	12-96	1C-9E	26-96
09-9F	13-97	1D-9F	27-96

Table 6

00-AD	0A-B5	14-B5	1E-B5
01-AD	0B-B5	15-B5	1F-B5
02-AD	0C-B5	16-B5	20-B5
03-B5	0D-B5	17-B5	21-B5
04-B5	0E-B5	18-B5	22-B5
05-B5	0F-B5	19-B5	23-AD
06-B5	10-B5	1A-B5	24-AD
07-B5	11-B5	1B-B5	25-AD
08-B5	12-B5	1C-B5	26-AD
09-B5	13-B5	1D-B5	27-AD

As an example, these tables tell the read routine to read track \$0F with DFAA9F-D5AAB5, instead of D5AA96-D5AAAD. After the tables have been entered, place Super IOB 1.5 in drive 2 and run it. Then EXEC the simplest controller you have (CON.STANDARD). Be sure to remove the disk from drive 2. Place a formatted blank in drive two, alter the read translate table (CALL-151/ BAD5:03) and copy away. You could also try cataloging both drives. The Mathblaster will now catalog in drive 1 as will the DOS 3.3 in drive 2. The copy does not need their DOS. But for the novices it is important that you know what a protected DOS looks like.

From here on I am relying on the softkey from Fred Long (COMPUTIST #53).

Davidson and Assoc. checks their disks with the use of "PEEK(40324)". Where this shows up in the code we must change the code. Reboot and load a normal DOS, then:

```

RENAME H[02]ELLO, HELLO
LOAD HELLO
POKE104,32
POKE33,33
LIST400
    
```

Change the = to <> and the <> to =.

```

POKE104,8
SAVE HELLO
LOAD M[02]ATH BLASTER
LIST 360
    
```

Change the = to <>.

```

LIST400
    
```

Change the = to <> and the <> to =

```

SAVE M[02]ATH BLASTER
    
```

The disk is now COPYable. Enjoy

### Notes:

Whoever placed ALGEBLASTER and WORD ATTACK on the MOST WANTED LIST, try DOUBLE DOS to softkey it. Has anyone else noticed that the list is getting long? I don't understand why some of the items are on the list. Certificate library volume one for instance. The softkey in COMPUTIST #43 for CERTIFICATE MAKER worked for me. The library disks can be copied with COPYA with a few changes.

```

RUN COPYA
70
CALL-151
B942:18
3A1:18
RUN
    
```

The drive will grind over track 21, but this is alright. If the program disk is deprotected then the library disks will also work. Next, my Wheel of Fortune was NOT protected. I don't think that Sharedata uses any protection on their disks. The MECC Puzzles and Posters can be copied by capturing their RWTS and using the SWAP controller.

Jim Bancroft

## Softkey for...

### Cross Clues

SRA

Well I finally got into Cross Clues by SRA. You just have to keep on peeling off the layers of protection. First the disk format is in DOS 3.2 with epilog modification. Second all file names have 8 control-H's at their ends making

them impossible to type at the keyboard. Third the graphics controller file is encoded on the disk and counts on SRA's DOS to do part of the decoding. Fourth the usual control-D DOS commands are replaced by really strange control-D control-various one letter entry commands. In combination with the control characters appended to the file names, this makes really odd disk access routines in the BASIC programs.

This job would not have been done without the Senior Prom, the Zap editor from Bag of Tricks and the disk utilities of Copy II Plus. With all the protection uncovered however, you can now survive with just the Copy II Plus utilities if you don't mind a lot of extra effort.

**1** Capture the RWTS and use it to copy the disk to a normally initialized DOS 3.3 or fast DOS disk. The startup program should be called "A". I used the Senior Prom and COPY B which is on its utility disk for this job. It takes approximately 2 minutes to get your normalized copy.

**2** Use Copy II Plus to "RENAME" the "FILES" on the disk. Remove the control characters from each name. This will give you files A, B, C, and D in BASIC; file G, which is the picture of the Sherlock Holmes hat; file F which is the ASCII data file; and file E, which looks like garbage but is the encoded graphics controller file that I'm certain has stopped deprotection of this program up until now.

**3** Capture file E in decoded form. Obviously the file must be decoded if it is operating throughout the program. Since Copy II Plus just gave us the address (16384) and length (2559) of file E on our normalized disk, we simply have to stop program execution get normal DOS booted and BSAVE it.

Again the Senior Prom or the Fingerprint + printer interface makes this a cinch; but given the file's location in memory, its quite likely that a **GO RESET** to normal DOS should allow a clean save. I did not try that however. Why take a chance on losing code when you have two of the best memory capture devices ever made for the Apple.

**4** Normalize the DOS commands and the names of files which are called by the programs. Check the BASIC lines which do this job. They show how putting in protection must drive programmers up a wall. The simple job of opening, positioning, reading and closing a text file becomes a one of kind code set, useless on any other disk. While you're altering the DOS calls, eliminate the decoding of file E after its BLOAD in programs B, C, and D.

Their command set:

```

CTRL-D;CTRL-Q = RUN (used on B, C & D)
CTRL-D;CTRL-B = BLOAD (used on E & G)
CTRL-D;CTRL-O = OPEN (used on F)
CTRL-D;CTRL-P = POSITION (used on F)
    
```

CTRL-D; CTRL-R = READ (used on F)  
CTRL-D; CTRL-C = CLOSE (used on F)

You will notice that before any DOS operation, there is a for-next loop which adds CHR\$(8) to the name of the file to call. That is how they add the control-H's to the file name. You will of course eliminate that loop since you have normalized the names. It would be impossible to save such names to disk from the keyboard.

In program A: make line 16 REM and normalize the command in line 20

In Program B: make line 10 PRINT CHR\$(4)"BLOAD E". Do not call the decode routine. You will have an already decoded E file on your disk. Eliminate the file name loop and normalize the DOS commands in lines 108 and 294.

In Program C: make line 3 PRINT CHR\$(4)"BLOAD E":GOTO 10. In line 1540 and 11400 normalize the file names and the DOS commands. In the routine running from 13028 to 13040, REM line 13028 to kill the name change loop and alter the DOS commands in lines 13030 and 13040.

In Program D: make line 3 PRINT CHR\$(4)"BLOAD E":GOTO 10 and change the DOS command in line 32020  
A note regarding the decoding of file E:

File E is BLOADED by file B and also by file C and file D. In all three programs immediately after the BLOAD of E there is a call to a location inside E which does the decoding. The answer is simple take out the call. To make life a little more interesting most calls within the BASIC programs are done as interdependent variables (P1 = P2 + 5 etc.) After extensive code following within E, I found that only the call immediately after the BLOAD will execute a decode. There is no alternate entry methods used. It was unnecessary to place any RTS within file E to disable the decode routine.

Softkey for...

## Thinkware

SRA

Search the disk for BD 8C C0 10 FB C9 xx 00 01 (where xx is a strange byte like E7) and replace the 01 with 1D. What you just did was bypass the check for funny sync bytes and branch to the check for the normal data header. This set contains many nice Science Simulations such as The Village Fish Farmer and The Town Engineer Planner.

## Some notes on Mindscape

Mindscape uses several different protections. Be sure to check COMPUTIST #28 page 22 for some of them but here are a few more.

On their disks using FORTH as the operating system, check track \$00, sector \$03. It contains a jump to reboot make any BNE a D0 00 before the reboot (4C 00 C6) in this sector. Now read track \$00, sector \$05 and what you find is the code that Spinnaker and Designware use all the time on their track \$00, sector \$3. Referring to COMPUTIST #22 page 4 "deprotecting Storymaker" we find that changing byte \$71 from 02 to 00 eliminates the protection.

On their disks using Dinky DOS changing the 4C 00 C6 on track \$00, sector \$0B to 4C 06 17 (which appeared a few bytes previous) took care of that protection.

In their LEARNING WELL series modifying DOS 3.3 to ignore the first byte of the data header (B8E6:29 00) allowed COPYA to handle this set. Then replace their DOS with a normal or fast DOS and your all set.

Their Pascal is simple. Use the Zap editor of Bag of Tricks to take block \$00 from a normal Pascal disk and put it onto a copy of their program made with COPYA. This same method works for Scholastic's new ProDOS protection if you take block \$00 from a normal ProDOS disk

## Some notes on Scholastic

Speaking of Scholastic they have released a version of Newsroom with protection slightly different from Springboard's original. On track \$00, sector \$03, change byte \$42 from 38 to 18. On Track \$02, sector \$0F their are two JSRs followed by BCS (20 24 1F BCS xx or 20 27 1F BCS xx). They check the ID of the disk. Change them to EAEA EAEA. Elsewhere, on the disk, you will find a disk check (BD 8C C0) which looks for some funny bytes like E9 and F7. make the first four BNE's of the sector D0 00 and you're set. This routine was on the disks I checked at track \$12, sector \$03 and again at track \$17, sector \$0E, so you should check the entire disk.

## Some notes on Hartley software

Hartley normally copies by turning off error checking (B942:18) and using COPYA. However if an IO error comes up after the DOS prompt appears, check your Hello program for a call to DOS (up in the 40000's), this checks to see if track \$23 contains certain code. Remove the call and you should be set. The Voyage of the Mimi uses this same protection.

Softkey for...

## Megabots

Spinnaker (UXB)

Spinnaker (under the company name UXB) has a little adventure game called Megabots. Dropping the CALL 4001 in the Hello program eliminates the protection check. Notice

that 4001 is actually part of the Hello in memory so you should not drop the line since that would relocate or lose the attached binary code at the end of the Hello. Change the CALL4001 to REM4001 and you will not create that problem.

Softkey for...

## Smart Eyes

Addison Wesley

Addison Wesley has Smart Eyes which contains disk check code (B6 BD 8C C0) in a file called SMART.OBJ on track \$21, sector \$03. You must load the accumulator and set the carry flag before returning so change the code to 9D 88 C0 38 60 and you are set.

## Some notes on Prentice Hall

Prentice Hall has a series of normal format software that gives nasty messages when you copy it and run it. A disk map shows track \$05 sector \$03 in use but no file using the sector. Byte \$52 of the sector has disk access (BD 8C C0) code. At byte \$61 there is a 4C 66 4B, at byte \$64 there is a A0 00 DD 88 C0 60, at byte \$25 there is a BEQ (F0 xx) which lands at byte \$64. Decreasing the value of xx by 3 will land you at byte \$61 and you will have passed the disk check successfully.

## Some notes on CAI

CAI (Computer Academic Ideas) has a variety of game show style quiz programs. You will find the usual BD 8C C0 followed by many C9 E7's in their protection. Finally there will be a BEQ (F0 xx) to some location. If the code at that location is 18 90 3E change it to EA EA EA. If it isn't, study the code in front of your location. If it is a JMP then change the code at your location to the same JMP. Things should then work fine.

Softkey for...

## PFS Computer Checkup

XPS Diagnostics is now PFS Computer Checkup but the protection remains the same. Copy with Locksmith fast back-up ignoring errors on track \$03 and also on tracks \$1D-22. They are nibble count and unused tracks. Look for 20 47 17 on the copy change it to A0 29 A9 0C 60. This loads Y with 29 and A with 0C and then returns to the protection check call without checking. These values are then stored at \$3A and \$3B to be used later by code at \$1742. The Senior Prom told me this and it didn't lie. The copy boots much faster without the extensive and precise nibble count.

## Some notes on CTW Series

In COMPUTIST #28 page 23 there was a softkey for Alphabetic Keyboarding by Southwestern Publishing. Personally I didn't use that method, I used insertion of a pretty long routine into normal DOS 3.3 to read the disk. What I noticed though, was that my modified DOS could read Alphabetic Keyboarding and the entire Apple Presents CTW Series (an old set of Integer BASIC classics). My guess is that the softkey in COMPUTIST for the keyboarding set will take care of the CTW set too.

## Some notes on Garfield

In COMPUTIST #55 Jerry Stevens softkeyed Create With Garfield Deluxe Edition perfectly. Thanks Jerry I lost my notes on that one but we did exactly the same thing to exactly the same code. You figured out though what the heck we were doing. You found the translation of the 1F to C0 which read the disk. For the life of me I had no idea where the working value was coming from, but I do know what the program did with it. Look at the code from \$6DD6 to \$6DDB. It has done all kinds of AND's and OR's then stored the value at \$6E67. It then compares the final value to \$5C. If carry is set, it clears, shuts off the drive and returns. If carry is clear it sets shuts off the drive and returns. The change of 38 to 18 we made means no matter what ends up at \$6E67 it does the same thing.

Jerry mentioned an unused signature on track \$23. DLM also publishes a twin program called Teddy Bearells of Fun. The protection is exactly the same as Jerry described for Garfield, but this program uses every sector from track \$02 on for data; so, its nibble count is done with track \$23. They obviously format hundreds of disks then put the programs onto them. Garfield did not overwrite track \$22, Teddy did. Big, fat, hairy deal!

Finally let's give credit where credit is due. The Create With Garfield softkey by Robert Brown in COMPUTIST #44 has been called many things, none of which are kind, since it was first published. The softkey was brilliant. It accomplished with A2 00 60 what took me 36 EA's. Refer to Mark Harris in COMPUTIST #55 for a complete list of the sectors you will need to work in if you don't want to search. Search for 8D 9E B8. This code stores the epilog byte into DOS as Garfield changes from Program disk to Data disk. Naturally since your program is now the same as the data disk, you don't want to do that anymore. You still have to tell the program that you did it though. So, 3 bytes in front of the 8D 9E B8 put A2 00 60 to load the X-register with 00 and return to the caller.

You are going to have one minor problem with your normalized copy. When it is time to initialize, if the disk you want to initialize has

Garfield epilogs, it will not initialize it. Since Garfield epilogs are now normal epilogs, that means any disk which has been previously formatted can not be reinitialized. Solution open the drive door until it reads two IO errors (that awful growling sound) then close the door. The IO error sets the carry flag just as if it found no epilogs at all, and it initializes the "unformatted" disk.

John E. Wanner

## Playing Tips for...

### Wasteland

Electronic Arts

Wasteland is a fascinating role-playing adventure game which uses Uzis and Laser weapons instead of the usual swords and magic. As with many such games, much of the fun is found in exploring the extremely large world scenario and solving its puzzles, but much of the frustration is in trying to obtain a set of survivable characters with enough offensive and defensive weaponry to handle nearly anything in the game. This tip will help you get the strongest armor and most useful weapons in a fairly efficient manner.

1. First, there are four sides to the game disks. Make several copies of each; especially at least 6 copies of disk 2. The fastest way to do this is to let the game disk copy one of each and then copy each of these sides using a fast copy program (like Locksmith Fast Copy.) Be sure to have at least 2 copies of side 1 at all times and reserve one of these. Then *anytime* you want to do something foolish (like try for the "big bucks" or enter a very difficult area) back your characters up to *both*. Each save only takes 10 seconds or so.

2. Start with the 4 players you get on disk until you know what you want. Those characters will survive *and* you can always add skills as you go through the game.

3. Make your way to the village of Needles, in the central part of the map, just across the river. Enter from the N and you'll be near a police ammo dump. One of your stock characters can probably (save off and keep trying) disarm the booby traps. (Stay away from the old TNT!!) Get more ammo and explosives. Go to the police station, pass the desk sergeant and go to the first room on the right. Have a character with the appropriate skills (keep trying!) pick the lock, search the right wall for the alarm and set it to 0. Raid the now disarmed armory. *Do not mess with the cells in the jail! This is fatal!*

4. Head S to town. You'll have some spare weapons and equipment (I found grenades were next to useless) and find LeRoy's (2nd building East of the edge of town) and sell your extra

gear. I found it convenient to have uniform ammo so I could trade when one person ran low, so I stuck with all 9mm equipment. In any case, you may find LeRoy has some useful goodies you can now afford, but *always* have a few extra inventory slots for each character.

4. Leaving LeRoy's you'll probably be confronted with "jerks"; if not, go to the alley just W and you certainly will. Kill everyone there *except* the woman. *Hire her*. Her name is Christina and she comes equipped with an armor class 5 radiation suit and Uzi. Pick up all the goodies you find and sell them back to LeRoy. You can repeat this and get 2 non-player characters (NPC) named Christina, a fairly powerful combination. Don't get a third NPC *yet!*

5. At this point, you may need medical attention. The cheapest place you can get this is the town of Highland, the closest to Ranger Central. If a player is in serious (or worse) condition have your best medic (or doctor if you have one) stabilize to unconscious. *Keep trying!* The more you use these skills, the better you get. If you can't cure a disease or condition, go to Highpool and pay for exam and cure, but not for healing. To heal cheaply, repeatedly press [ESC]. (Set something on it if you've got a IIe or IIc.) Make a snack, pop some popcorn, get a soda. In 5 minutes or so, they should have all their hit points back. (*Be sure that they're at least UNC first, however. Worse condition will kill them!*)

6. Now you're ready for the big bucks. Go to Quartz, just on the E side of the river and S of Needles. Enter from the NE and go S. There you will find a building at the N end of a road. Go S. You will find a building on the corner. With both Christinas in the lead, fight your way in (if you don't encounter a door guard, it's the wrong place), get to the back room where Felicia is held captive by Ugly. *Save off*. Kill Ugly and the guards and either crack the safe or disarm the bomb and blow it with the plastic explosive. There's about \$1000 there. Search the SW of the hideout and find and blow the other safe, getting another \$500 or so.

7. Find the courthouse (1 building S and E of the hideout) and blast your way in. Go to the second floor and free Danny Citrine and hire him. Go to the 3rd floor and open the SE cell (pick or TNT) where Danny will get you \$1000 more. (The BED!!)

8. Disband Danny permanently, save off and repeat 6 & 7 until you have \$25000 or so. In so doing, you will fight other bad guys, get more ammo, money and even an occasional sub-machine gun or so and earn experience, too. As your characters gain experience, add only IQ points for a while. Enter the libraries (there's one near the police ammo dump, for instance) and add skills. *Absolutely necessary* in the long haul will be assault rifle, doctor and

energy weapon skills. Make sure all your characters (even NPCs) have them. You should also have a couple of players with anti-tank weapon capabilities. Finally, when these are developed, add strength points to one character until they're at least at 20.

9. Now, go back and get another Christina (unless you've got another strong NPC already.) Pool all your \$ to one, disband (*Non-permanently*) your Christinas and go NE to Darwin. The fastest way to enter is from the SW, but make sure you have *radiation suits!* The building in the SW corner is the black market. Blast your way in, sell off everything but the canteens and radiation suits and buy 4 more radiation suits, 7 NATO rifles, at least 28 clips of ammo and as many Sabot rockets as you can afford. You'll also need some TNT or plastic explosive if you don't have any. Go back and distribute the goodies. You're now ready for the REALLY good stuff.

10. Enter the Citadel. (This is in the mountains to the NW of Ranger Central and N of the Agricultural station.) *Save off to both disks!* Get as close as possible, then blast anything that moves. Have your AT weapon characters alternate shots with Sabot rockets, LAW rockets or RPG-7s if you have any. Keep blasting. You may have to leave and heal your characters occasionally, but the Citadel characters you kill stay dead, so it eventually gets easier.

In the Citadel, you'll find all kinds of energy weapons and power packs. In the NW section of the second chamber, there is a library where you can get energy weapon capability if you don't already have it. As soon as possible, equip and start using them. Fight your way straight to the back of the hall near the steps and portcullis. Enter the door at the left of the steps, fight your way to the control wheel and get your strongest character to use strength on it. Enter the steps and open the safe to the E. The combination is "rosebud". This will give you more power packs and more importantly 5 suits of power armor, AC 14. Of course, you can repeat this with as many disks as you need to equip all your troops. If you always sell your extra goodies to LeRoy *on the same disk*, you can then *buy* extra power packs, weapons or armor if you run out or accidentally lose them. With these goodies, you will be able to withstand most any of the nasties you encounter in Wasteland and be able to concentrate on exploration of the world and its puzzles.

Arthur Simon

I have been a subscriber to your magazine for several years now and have found it to be of immeasurable value for backing up my valuable software. I recently switched from an Apple IIe to an Apple IIgs and, in doing so,

purchased the IIgs version of Springboard's Certificate Maker. This is a slightly different version of the program than that which was initially introduced about 2 years ago for the IIe. Although the original version would boot and allow one to create a certificate on the IIgs, it failed as soon as you tried to print your creation. The new version recognizes the capability of printing from the printer port of the IIgs.

Unfortunately, when Springboard modified the program to work on the IIgs they also apparently changed the copy protection scheme. At least it appears that way, since neither the softkey in COMPUTIST #39, page 6 nor that in COMPUTIST #43, page 24 works on the new version. In fact, track \$22 is now easily readable (this was the track that previously caused problems in making a routine copy) but track \$20 cannot be read by a sector editor.

Perhaps there is an easy fix to the previous softkeys that will work on the new IIgs version. Not being expert in this area, I would appreciate if someone could take the time to figure one out and offer me a solution. Certificate Maker is a very popular program and no doubt many other readers with IIgs's will be interested in such a softkey.

## Silicon Sorcerer

Softkey for...

### Rescue on Fractalus

Epyx / Lucasfilm

#### ■ Requirements

- Apple II computer
- Original disk
- Blank disk
- Copy program that ignores errors
- Beneath Apple ProDOS (optional)

In Rescue on Fractalus, the player must fly a jet through alien terrain searching for downed pilots. The ship is fitted with antimatter cannons and shields, and the enemy consists of mountain-top gun emplacements and flying saucers. The most striking feature of this program is the terrain. As suggested by the title, it is made up of 3-D fractal mountains. Naturally it is protected, and the protection is extremely difficult to bit-copy. Bit-copies usually died at track \$00.

#### The Protection

As usual, the first thing I did was watch and listen to the boot sequence. On booting, the disk seems to load all or most of tracks \$00 through \$07 in order, give the title screens and music, and then loaded tracks \$08 through \$12

the same way. In a normal boot, the computer reads DOS from the first three tracks and then moves the head way over to track \$11. Obviously the programmers were not using DOS.

Next I tried to copy the disk. It copied correctly, indicating that Fractalus probably had a normal RWTS. This also meant that I could easily sector edit it. However, there was a read error on track \$00 sector \$0B. At first I was afraid that my original was already going bad, but as it turned out this bad sector was a beacon signaling the method of protection. Since the protection was on track \$00, boot code tracing sounded viable.

Examination of track \$00, sector \$00 showed that it was completely normal for a slave disk. This meant that RWTS would be loaded from sectors \$00 through \$09 into memory from \$B600 through \$BFFF, and execution of RWTS would begin at \$B700.

The RWTS, on the other hand, had obviously been modified. The first line, at \$B700, had been changed from "STX \$B7E9" to "JSR \$BB00". \$BB00-\$BC55 is RWTS's buffer, where sectors are moved during disk access, so it is usually filled with garbage. In Fractalus it contains the protection code. The actual protection code resides at \$BB34-\$BBCB.

First the routine uses the normal "Read Address" subroutine to find physical sector \$08, which is logical sector \$0B. On finding the sector, it locates the address header and begins reading bytes at \$BB63. Below is a partial listing:

```

BB63- BD 8C C0 LDA $C08C,X
BB66- 10 FB BPL $BB63
BB68- BD 8C C0 LDA $C08C,X
BB6B- 30 FB BMI $BB68
BB6D- BD 8C C0 LDA $C08C,X
BB70- 10 FB BPL $BB6D
BB72- BD 8C C0 STA $C08D,X
BB75- 10 FB NOP
BB76- BD 8C C0 LDA $C08C,X
BB79- 30 FB BMI $BB76
BB7B- BD 8C C0 LDA $C08C,X
BB7E- 10 FB BPL $BB7B
BB80- BD 8C C0 LDA $C08C,X
BB83- 30 FB BMI $BB80
BB85- BD 8C C0 LDA $C08C,X
BB88- 10 FB BPL $BB85
    
```

There were a few things in this code that looked strange to me. The first, a minor one, was that the branches were alternately BPL and BMI. A BPL in this situation indicates invalid data, and why would they want to read that? In the past, I recalled seeing only BMI's. The answer, of course, is to make sure the program didn't read the same byte off the disk twice, before the hardware could begin reading the next one. An interesting side-effect is that it also disables Locksmith's Automatic Boot-Tracer.

The second thing that was strange was the bytes the program was apparently reading in while it was running. I would boot-code trace a bit further and look at the bytes it would read and store. When I tried to find them on the disk, they were completely different. For example, where the program read "EB B6 EF 9A...", I would find "FD ED BB E6..."

The third thing that puzzled me was the STA \$C08D,X. This location is usually only accessed during writing or sensing whether or not a disk is write-protected. Actually, I didn't figure it out until after I had cracked it. The thing that helped me get it was Beneath Apple ProDOS, by the authors of Beneath Apple DOS. In the back of the book there is a section on the "state-sequence ROM". If \$C08E,X and \$C08D,X are accessed, as they are here, the ROM goes to sequence #0 and stays there until the state is changed, as it is in this case by the command LDA \$C08C,X. This command comes 6 machine cycles later. So the effect this routine has is to extend the time used to read the byte by 6 cycles, or about 1.5 disk-bits (each disk-bit takes about 4 cycles to be read in).

To understand the significance of this, we will have to examine the disk-bits on a binary level. Below are the bits immediately following the Address Header (D5 AA AD) of sector \$0B:

```
1111111101011111010111110011110111...
```

Normally, the bits are divided into groups of 8 by the hardware as follows:

```
11111111 10101111 10101111 10011110 111...
```

Most nibble-editors display this bit stream in hexadecimal:

```
FF      CF      CF      9E      ...
```

This is what you see when you read the disk normally with a nibble-editor. Remember, though, that Fractalus adds 6 cycles to the third byte, giving us:

```
11111111 10101111 10101111 00 11110111...
```

The two zeros are dropped because all disk-bytes must begin with a "1". The hardware treats the third byte as a sync-byte. This gives us, in hexadecimal:

```
FF      CF      xx      F7      ...
```

The third byte is uncertain, since shifting states in the middle of a read tends to destroy the read byte.

Notice that normally there are 3 binary ones left over, which become part of the fifth byte. In the Fractalus read, these bits have been used up. This will change the next group of bytes completely, just as it changed the fourth byte here! Normally they would be "FD ED BB E6...", but now they are "EB B6 EF 9A...", which is what Fractalus looks for! In short, this is a way to hide protection information from hackers' prying eyes. In addition, bit copiers have trouble reading the "sync bytes" perfectly, so bit copies usually fail.

Now, after Fractalus reads in the next \$10

bytes, it goes through a mess of adds, eXclusive-ORs, and logical "and"'s in an attempt to confuse any would-be krackist. It then stores the results at \$3F0 and \$3F1 and zeroes locations \$70 and \$71. This is to keep us from simply bypassing the protection routine. I found the needed results easily by boot-tracing. \$3F0 and \$3F1 should be D7 and F9. Incidentally, it is possible to boot-trace on a IIC, but it is a little more tricky, as critical bytes can change mysteriously.

To softkey Fractalus, I simply overwrote part of the protection code with code to change the 4 above locations and continue. Below is a disassembly:

```
BB34:LDA #D7
      STA $03F0
      LDA #F9
      STA $03F1
      JMP $BBC5
```

Address \$BB34 is on disk at track \$00, sector \$05, byte \$34.

### Step by Step

**1** Copy with any whole-disk copier that can ignore errors. Copy II Plus's "copy disk" option is fine. Don't use an unmodified COPYA, as it will die on pseudo-sector \$0B of track \$00.

**2** Make the following sector edit to the copy:

Trk	Sct	Byte(s)	From	To
\$00	\$05	\$34-\$40	A9 10 85 FF C6	A9 D7 8D F0 03
			FF D0 03 4C C6	A9 F9 8D F1 03
			B6 20 44	4C C5 BB

**3** (Optional) Change skewing to 4-descending. This will greatly reduce loading-time.

Charles Taylor

Softkey for...

## Deathsword

Epyx

### ■ Requirements

- COPYA
- Sector Editor
- 128K Apple

Deathsword can be cracked using an adaptation of Mr. Nissel's "Arctic Antics" procedure on page 25 of COMPUTIST #56.

- 1** Boot your DOS 3.3 system disk.
- 2** Tell DOS to ignore checksum and epilog errors and use COPYA to copy the disk.

### POKE 47426,24

#### RUN COPYA

**3** Make the following sector edits to the copy you just made.

Trk	Sct	Byte(s)	From	To
\$00	\$0E	\$4A-4B	D0 F4	EA EA
		\$53-54	D0 46	EA EA
		\$5C-5D	D0 3D	EA EA
		\$6F-70	D0 F4	EA EA
		\$71-7E	A0 07 BD 8C C0	EA EA A9 FC 85
			10 FB D1 F6 D0	F0 85 F3 A9 EE
			1F 88 10 F4	85 F1 85 F2

Softkey for...

## Gauntlet

Mindscape

Gauntlet is deprotected by a scheme identical to The Twit's "Infiltrator" Softkey on page 8 of COMPUTIST #56.

- 1** Boot your DOS 3.3 system disk.
- 2** Tell DOS to ignore checksum and epilog errors and use COPYA to copy the disk.

### POKE 47426,24

#### RUN COPYA

**3** Make the following sector edits to the copy you just made.

Trk	Sct	Byte(s)	From	To
\$00	\$03	\$42	38	18
	\$08	\$EA-EC	20 00 02	EA EA EA

Softkey for...

## Cofa Courseware

Emtech

### ■ Requirements

- SREAD/SWRITE (from COMPUTIST #24)
- Fast copy program
- 10 blank disk sides
- Method of copying DOS to disk (optional)

Cofa Courseware is a series of 5 double-sided math disks aimed at senior high school students. The disks all are normal format (COPYA-able), but the copies don't run. At first the simple protection on these disks had me stumped, since the scheme hasn't been used in a number of years. Many thanks to Bluebeard for reacquainting me with this protection scheme.

The reason the COPYA copies don't work is that the catalog track has been moved from its normal track \$11 to \$23. COPYA only copies through \$22. The deprotection is simple also. It involves moving the catalog back to

track \$11, sector editing all of the track \$11 sectors so that byte \$01 is 11 instead of 23, then copying a new DOS from the system master to the copy.

I used SREAD/SWRITE to read in track \$23 from the original and write back to track \$11 of copy, sector by sector. This laborious process might be OK for one side, but ten sides? I then wrote a BASIC program to normalize the catalog track, including the correction of the catalog track numbers. (POKE D,17)

**1** Copy all ten sides with a fast copy program.

**2** Type in the BASIC program and SAVE COFACOPY on a disk containing SREAD/SWRITE. Be aware of the typo in COMPUTIST #24. The line "BSAVE SREAD/SWRITE,A\$300,L\$3A4" should read "BSAVE SREAD/SWRITE, A\$300, L\$A4"

**3** Run COFACOPY. COFACOPY will prompt you to insert the originals and copies. Be sure to write protect the originals.

**4** Copy a normal DOS to the copies. Lacking a method to copy DOS, copy all files to an initialized disk.

### Softkey for...

## Mathbusters

Spinneraker

### Requirements

- copy program
- sector editor (optional)

Mathbusters can be copied with COPYA, but the copy won't run until the protection is defeated. Here's how I found the copy protection:

A catalog of the disk shows several programs with either IIe or IIc in the title. I reasoned that the first useful function of the program is to determine whether it was running on a IIe or IIc. A program can do this by checking the bytes at \$FBB3 and \$FBC0 and comparing with the following table:

FBB3	FBC0	Type
06	00	IIc
06	EA	IIe
06	E0	enhanced IIe
<06		II or II+

A sector editor found the \$FBB3 and \$FBC0 references on track \$1E, sector \$07. A catalog of the disk showed this sector to be part of BOOTER.OBJ, which loads in at \$92D0 and is \$2BE in length. The first instruction in BOOTER.OBJ is a JMP to \$93B7. The FBB3 was located at \$9402. (At this point I had loaded BOOTER.OBJ from

DOS, rather than using the sector editor so that I could look at the actual memory locations.) I made my non-working copy into a working copy by changing the \$93B7 to \$9400.

**1** Copy Mathbusters with a fast copy program.

**2** Sector edit

Trk	Sct	Byte(s)	From	To
\$1E	\$08	\$05-06	B7 93	00 94

**2** OPTIONAL method

**BLOAD BOOTER.OBJ**

**CALL-151**

**\$2D1:00 94**

**BSAVE BOOTER.OBJ, A\$92D0, L\$2BE**

Bill Jetzer

I would like to comment about your back issue section. How about different sections for 5 1/4" disks, 3 1/2" disks, and IIgs software? It would make things less confusing, and if someone orders a back issue for a particular piece of software, he can be sure that he won't get a softkey for the wrong type of disk or computer.

*That's a good idea but it would be almost impossible to do. Many readers do not send the requirements list with their softkeys. And they often do not mention what kind of media the program is on. Unless it's part of the name, sometimes we can't even tell which computer the softkey is for. So, if we separated the ones that we know into different sections, you would still have to search the main body of text for the ones that fell thru the cracks. Your best bet is to send us a letter requesting more info on the particular software that you are interested in.....RDEXed*

### Softkey for...

## Presenter

MECC

### Requirements

- 1 blank ProDOS disk
- a file copier

MECC software has never had very intense protection, and the Presenter is no exception. It changes the epilogue bytes from DE AA to FF AA and quits to ProDOS if its altered device driver is not present in memory.

To deprotect the Presenter, boot a ProDOS disk and get into BASIC. Next, tell the device driver in ProDOS to ignore read errors. This is similar to a POKE 47426,24 in DOS 3.3 except that it only ignores checksums and epilogues, not all read errors. Here is the

procedure for ProDOS v1.1-v1.4 (v1.0 addresses are in italics):

**CALL-151**

**C089 C089**

**D3E4:18 60**

**D4CC:18**

**C08A**

(FBE4)

(FCCC)

Now execute your file copier and copy a normal version of PRODOS onto the blank disk. Then copy all the files except PRODOS from the MECC disk onto the blank disk.

All that remains to be done is to stop the routine in MECC.SYSTEM that executes the ProDOS quit command if its device driver isn't present:

**BLOAD MECC.SYSTEM,A\$2000,TSYS**

**CALL-151**

**\$3B:18**

**BSAVE MECC.SYSTEM,A\$2000,TSYS**

was 38

### Softkey for...

## Catalyst 3.0

Quark Inc.

### Requirements

- 3 1/2" disk copier
- 1 blank 3 1/2" disk

I know there have been other softkeys for this program in the past, but I don't know if they were for the 5 1/4" disk or the 3 1/2" disk. Mine is for the 3 1/2" disk.

Blocks \$08-\$17 are stored on the disk in such a way that ProDOS can't read them, but they contain code that is necessary for the program to run. It is the job of CATSTART.SYSTEM to read these blocks into memory. Once in memory, a routine is executed that checks for the abnormal blocks.

To deprotect Catalyst 3.0, follow these steps:

**1** Copy the disk, ignoring the errors on blocks \$08-\$17.

**2** Get into BASIC by executing BASIC.SYSTEM.

**3** By loading in CATSTART.SYSTEM at its relocated address, you can execute the code that reads the abnormal blocks:

**BLOAD CATSTART.SYSTEM,A\$C00,TSYS**

**4** Before executing the code, put a BReaK instruction right after the code that reads the abnormal blocks:

**CALL-151**

**D47:00**

**5** Execute the code:

**C5FG**

**6** Once you hear the beep and see the registers, all the code has been read in. Actually, CATSTART.SYSTEM has no other purpose than loading those blocks, and it won't be used anymore. Instead, you can make a system file out of the code that was just read in. First, clear some memory:

```
2000:0 N 2001<2000.20FEM
```

**7** Now enter the relocater that will move the code to its proper address:

```
2000:A2 00 BD 00 40 9D 00 5F
2008:E8 D0 F7 CE 04 20 CE 07
2010:20 AD 07 20 C9 3F D0 EA
2018:4C 00 40
```

**8** Move the disk code to its new location so that it can be saved:

```
2100<4000.5FFFM
```

**9** You may have noticed a file named CATALYST.3.0 in the catalog. This file is located on blocks \$08-\$17 and kept these blocks from being allocated to files that were later saved to the disk. This is an ideal place to save the code:

```
UNLOCK CATALYST.3.0
BSAVE CATALYST.3.0,AS2000,LS2100,TSYS
```

**10** Lastly, give CATALYST.3.0 a new name so that PRODOS will be able to find and execute it:

```
RENAME CATALYST.3.0,CATALYST.SYSTEM
```

**?** I don't know much about 3 1/2" disks, and I want to know how Catalyst could make the device driver read blocks \$08-\$17 when ProDOS couldn't. If anybody knows where I could find a book explaining 3 1/2" disk formatting and the usage of its device drivers or if you could explain it yourself, I would like to hear from you. I would also like to read more about 3 1/2" diskette protection schemes and about using the drive independent of the device driver. You may write to me at; Route 1, Plymouth, WI 53073.

MJ Howard

Here are a few cracks that have worked for me on G.I. Joe by Epyx and Tink's Subtraction Fair by Mindscape. Not exactly hot stuff, but bit copies of them would not work, therefore the protection on them had to be removed.

Using past issues of Computist (a valuable reference) I found sector edits of programs from Epyx and Mindscape that would work on my disks, but on different sectors.

I determined which sectors to be edited by scanning the disks for the indicated byte sequence to be changed.

Softkey for...

## Tink's Subtraction Fair

Mindscape

**1** Boot your DOS 3.3 system disk.

**2** Tell DOS to ignore checksum and pilog errors and use COPYA to copy both sides of the disk.

```
POKE 47426,24
RUN COPYA
```

**3** Make the following sector edits to the copy you just made.

Trk	Sct	Byte(s)	From	To
\$00	\$05	\$5F-61	20 44 B9	EA EA 60

Softkey for...

## G.I. Joe

Epyx

**1** Boot your DOS 3.3 system disk.

**2** Tell DOS to ignore checksum and pilog errors and use COPYA to copy the disk.

```
POKE 47426,24
RUN COPYA
```

**3** Make the following sector edits to the copy you just made.  
on side 1:

Trk	Sct	Byte(s)	From	To
\$00	\$01	\$00-1F	A9 0A 85 F4 A6 2B BD 89 ...	A9 FC 85 F0 A9 EE 85 F1 A9 EE 85 F2 A9 FC 85 F3 A9 E7 85 F4 A9 EE 85 F5 A9 FC 85 F6 A9 E7 85 F7 all EAs
		\$20-79 ?		

**?** While I have had much success backing up my disks using sector edits and modifying disks slightly, your IOB program really puts me off. I have never been able to get the hang of it and receive a variety of error messages whenever trying to use it. Whether installing a standard controller or typing in a new one from the magazine, it never works. Is it just me?

**I** Most of our readers don't have any problem using Super IOB (SIOB). You don't say what computer you are using or exactly what you were doing when you got the error messages, nor what the error messages were, so I can't tell what's wrong. However, here are some of the problems most often encountered the first time a reader tries the SIOB program:

1. Trying to run SIOB without a controller. The SIOB on your Starter Kit disk is bare. You must LOAD SIOB, then EXEC a controller into it, then RUN it. Use CAPTURE on the Starter Kit disk to convert an Applesoft controller into an EXECutable file.

2. Trying to run a controller without SIOB. The controller is the heart of the SIOB program. The SIOB program is a shell of useful routines. The controller calls (GOSUBs) these routines in the proper order to copy a disk. The controller is not a stand-alone program.

3. LOADING SIOB then LOADING a controller. By LOADING the controller, you dumped SIOB and only the controller is left in memory. (See #2) You must EXEC a controller after you have LOADED SIOB, then type "RUN" and press **(RETURN)**.

4. Variations on #3 result in #1 and #2. IE. LOADING the controller then LOADING SIOB, LOADING SIOB and RUNning the controller, etc.

5. Trying to use the swap controller without first saving the RWTS. The swap controller uses the RWTS from a protected DOS to read a protected disk. You must first capture and BSAVE the RWTS from that protected disk.

I'm sure we can find out why SIOB isn't working for you. Why don't you give it another shot? .....

Jack Nissel

Softkey for...

## Tower of Myraglen

PBI Software

### Requirements

- 2 blank 3.5 disks
- Any fast copy program that can ignore errors
- A sector editor

**1** Copy the original disks to your blank disks and ignore a read error if one is encountered.

**2** Make this sector edit to disk 1.

Block	Byte(s)	From	To
\$40F	\$72-75	22 CB 1B 00	EA EA EA EA

**3** Write the sector back to the disk.

Softkey for...

## Cannonball Blitz

Sierra On-Line

### Requirements

- 1 blank disk
- A sector editor
- COPYA (from DOS 3.3 system master disk)

# Readers Data Exchange

- 1 Boot your DOS 3.3 system disk.
  - 2 Tell DOS to ignore checksum and epilog errors and use COPYA to copy the disk.  
**POKE 47426,24**  
**RUN COPYA**
  - 3 Make the following sector edits to the copy you just made.
- | Trk  | Sct  | Byte(s) | From | To |
|------|------|---------|------|----|
| \$17 | \$0E | \$CD    | ?    | 60 |
- 4 Write the sector back to the disk.

Softkey for...

**PFS FILE**  
**PFS GRAPH**  
**PFS REPORT**

Software Publishing Corp.

The following softkey is from CompuServe's Apple MAUG. This information is only for the purpose of combining these three popular programs to one 3½" disk or for making 3½" copies that will boot on the IIGs.

All three programs are de-protected identically. *Make these changes to copies of the files, NOT the originals!* Six bytes need to be changed in the .SYSTEM file. Make the changes from BASIC.

**BLOAD /prefix/FILE.SYSTEM, TSYS, A\$1000**  
**CALL -151**

**12D7:A9**  
**12DE:A9**  
**12E0:EA EA**  
**12E8:EA EA**

**BSAVE /prefix/FILE.SYSTEM, TSYS, A\$1000**

Repeat the above for GRAPH.SYSTEM and REPORT.SYSTEM. Write protect the copy.

Gary Verbuch

I would like to thank the volunteers at COMPUTIST for responding so quickly with my subscription. Within two weeks after sending a check I received my first issue. Now that's service.

I've been deprotecting software for years now and I never thought of it as a hobby, more like a necessity. I will not use an original piece of software until I have produced a working copy. Unfortunately, with some products it can take some time when the disk format is completely altered.

After reading Computist #57 it's evident that readers have been complaining about the softkeys not containing enough information about the protection or the process of

deprotection. Mr. Hart's article was a good general description of the protection schemes used on Apple software. He mentioned in his article a table of changes to DOS for reading altered address and data marks in an earlier issue. Since I do not own any reference material on DOS 3.3 I had to disassemble it to find out what makes it tick. For the readers who, like I, do not have that issue, I have compiled a list of changes to read altered formats. I will also include some important DOS locations that I find helpful when deprotecting 3.3 software and some pointers on the ProDOS MLI (machine language interface).

### Important Locations

\$3EA Reconnect DOS input/output hooks.  
\$AC01 Catalog track #  
\$A884-A907 DOS command locations  
JMP to \$FAA6 Sneaky way to cause the disk to reboot  
\$AA60-AA61 Length of the last bloaded file (low byte/high byte)  
\$AA72-AA73 Address of the last binary file (low byte/high byte)  
\$A964:FF Allow a large BSAVE. On a single load program you can BSAVE the program to a normal DOS disk.

### RWTS calls

\$3E3 Return the address of the IOB (input output block). Before a call to the RWTS a JSR to 3E3 returns the address of the IOB (Y reg lo-byte, A reg hi-byte).  
\$B7E8-B7F8 The normal IOB location  
\$3D9 The actual call to the RWTS is done by a JSR to 3D9 (20 D9 03). Any track or sector can be accessed thru this call, including a check on track 36 (\$23).  
\$BA29-BAFF Read/Write translate tables. Sometimes you will notice the data on the disk looks garbaged, no readable code. Check the Write translate table (BA29.BAFF) and the Read translate table (BA96.BAFF) on a normal DOS and compare them to your protected disk. One byte changed will garbage the data and make it unreadable by a normal DOS. *The read table is usually the one altered but I've included both locations.*

### ProDOS 8

\$B8D1-B93E DOS command table  
\$BEC8-BEC9 Last BLOAD length  
\$BEB9-BEBA Address of last BLOAD

To allow DOS to ignore changes to standard address and data markers use the following.

#### To ignore Address Prolog

D5 - \$B954:29 00 POKE 47444,41 POKE 47445,0  
AA - \$B95E:29 00 POKE 47454,41 POKE 47455,0  
96 - \$B969:29 00 POKE 47465,41 POKE 47466,0

#### To ignore Address Epilog

DE - \$B993:00 POKE 47507,0  
AA - \$B99D:00 POKE 47517,0

To ignore both:

\$B988:18 60 POKE 47496,24 POKE 47497,96

#### To ignore Data Prolog

D5 - \$B8E6:29 00 POKE 47334,41 POKE 47335,0  
AA - \$B8F3:00 POKE 47347,0  
AD - \$B8FE:00 POKE 47358,0

#### To ignore Data Epilog

\$B925:18 60 POKE 47397,24 POKE 47398,96

#### To ignore RWTS errors

\$BE48:18 POKE 48712,24

#### To ignore Checksum and Epilog

\$B942:18 POKE 47426,24

#### To ignore Data Checksum

\$B920:EA EA POKE 47392,234 POKE 47393,234

### ProDOS

A good percentage of software I've deprotected that uses ProDOS 8 or ProDOS 16 looks for a bad block or has an altered disk format. A ProDOS block is the equivalent of two DOS 3.3 sectors and at the disk level there is no difference between ProDOS or 3.3. Since it is beyond the scope of this article to cover either ProDOS version in detail, I will include the MLI (machine language interface) call most often used in protection methods. Keep in mind when scanning a disk for a direct read (LDA C08C,X OR LDA C0EC), that other methods are available in a ProDOS environment.

### ProDOS 8

ProDOS 8 MLI calls begin with a JSR \$BF00 (which is the entry point to the MLI) and are immediately followed by three bytes. The first byte will contain the command code (\$80 for Read Block), followed by two bytes that form a pointer to the parameter block. What is contained in this block depends on the type of MLI call issued. After a call is completed, the code following the three bytes is where execution resumes. If an error occurred during the MLI call then the carry flag is set. In normal use I will use BCS to test for an error. On a disk checking for a bad block the code will be expecting an error and often tests for the carry bit clear. Either test of the carry bit can be used.

Here is an example of a Read Block call to test for a bad block.

## ProDOS 8 - Read Block

```

2000: 20 00 BF  MLI entry point
2003: 80      command code for Read Block
2004: ?? ??   address of parm block, defined by user
                (lo-byte/hi-byte order)
2006: 90 ??   branch on carry clear to crash -block was
                read and an error did not occur
????: 03     parm block -1st byte will be 03
????: ??     2nd byte will contain device & drive #
????: ?? ??   3rd and 4th byte will contain the pointers
                to the data buffer (lo-byte/hi-byte order)
????: ?? ??   byte 5 & 6 will contain the logical block #
                to read (lo-byte/hi-byte order)
    
```

## ProDOS 16

When a system block call is used in this environment there are a few minor differences because of the memory capacity of the IIGs. The format of an MLI call is the same but the entry point is in bank \$E1. The pointer to the parameter block and the definitions within the block have to accommodate the increased memory range. I will include an example similar to the ProDOS 8 version.

## ProDOS 16 - Read Block

```

2000: 22 A8 00 E1  JSL to MLI entry point ($E1/00A8)
2005: 00 22      read block command code ($22), 2
                bytes
2007: ?? ?? ?? ?? user defined addr of parm block
                (lo/hi/bank), 4 bytes
BCS OR BCC      carry set, block is bad - carry clear,
                block is good
????: ?? ??     PARM block, byte 0-1, device #
????: ?? ?? ?? ?? byte 2-5, pointer to data buffer (lo/hi)
????: ?? ?? ?? ?? block to read, (lo/hi order)
    
```

I hope the above information will help some of the readers who are not familiar with the ProDOS environment. In comparison to other Disk operating systems I've worked with (Unix, Xenix) in the past, I find DOS 3.3, ProDOS 8 & 16 very easy to learn and write systems software for. If you are new to deprotecting software I recommend getting a DOS reference manual and start disassembling the read, write and boot code. The DOS 3.3 table I've included is a result of studying the routines to read and write directly thru the device select address range (\$C0E0-C0EF).

By the way, are there any readers familiar with the Scheme dialect of Lisp or a version that does lexical scoping by default. I've come up with some interesting coding techniques that I'd like to share.

Craig Meekins

Broadsides by SSI is a very intriguing war game. You are in command of an authentic sailing warship out to devastate French, Spanish, English, or American Warships in one on one high seas combat. If you don't like the

vessels the computer can supply you with there are options to customize your own super frigate, or you can customize the enemy to be as tough or as weak as you want them.

Softkey for...

**Broadsides v2.0**  
Strategic Simulations Inc.

The trouble is SSI has protected this game in such a way that Locksmith 5.0, EDD 4.7, and Copy II Plus 8.2 will not give a working backup even with extensive parameters. I naturally turned to COMPUTIST back issues for help. There seemed to be no listing for it in previous issues, but several SSI games were documented. Study of these previous techniques led to this softkey.

Remember previous issues that cover the same company are a great starting point for protected programs that are not listed.

## The Protection

1. The even tracks all have standard address prologs of D5 AA 96, however the odd tracks have nonstandard address prologs of D4 AA 96.

2. The address field epilog bytes have been changed from DE AA on all tracks to read AF followed by a random byte that varies from sector to sector.

## The Procedure

**1** Boot a DOS 3.3 System Master and alter DOS to read Broadsides original disk.

### CALL-151

**B954:4A C9 6A D0 EF** to allow D5 and D4 in address prolog

**B988:18 60** to ignore address epilog errors

### 3DOG

### RUN COPYA

**2** Break out a sector editor and edit track \$00, sector \$0C to allow DOS to read then new address epilogs.

Trk	Sct	Byte(s)	From	To
\$00	\$0C	\$A9-AA	A8 D0	18 60

You should now have a unprotected BROADSIDES. If any one is using Keith Parker's PICTURE LOADER (COMPUTIST #54) you can capture the boot up screen by reading starting at track \$04, sector \$13 forwards.

Softkey for...

**Aztec**  
Datamost

Here's a quick one for a old Adventure /Arcade game. The Prologs have been changed

from D5 AA 96 to AA D5 AD for the address field and from D5 AA AD to AA D5 96 for the data field. The Epilogs have been changed from DE AA to AA AA and D5 AA for the address and data fields respectively.

**1** INIT a blank disk with a fast DOS (I used ProntoDOS).

### INIT AZTEC

**2** Enter the controller below into Super IOB 1.5 and run it answering no when prompted to format.

## Controller

```

1000 REM AZTEC CONTROLLER
1010 TK = 3:LT = 31:CD = WR:MB = 151
1015 POKE 47426,24
1020 ST = 0:T1 = TK:GOSUB 490:RESTORE:GOSUB
    190:GOSUB 210
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 170,213,173,170,213,154
    
```

## Checksums

1000 - \$356B	1060 - \$CE4A
1010 - \$9ACF	1070 - \$8E01
1015 - \$BDA8	1080 - \$A88C
1020 - \$D510	1090 - \$836E
1030 - \$DEBE	1100 - \$C53E
1040 - \$B0CF	5000 - \$4F55
1050 - \$4E18	

There you have it. Enjoy it and watch out for the dragons!

 This softkey is only slightly different from Leo & Eric Van Der Loo's softkey in COMPUTIST #53, page 12. But it is a bit easier.....RDEXed

J.L. Walters

Softkey for...

**Microzine #25**  
Scholastic

Microzine #25 has the same protection scheme as Microzine #24 which was detailed in COMPUTIST #57 on page 9. The only difference is that the file name of the nibble count routine was changed from CP.OPTIONS to TOC.6. See COMPUTIST #57 for a detailed

description of the nibble count routine.

**1** Copy both sides of Microzine #25 using your choice of copy programs.

**2** Boot your DOS 3.3 System Master.

**3** Place side one of Microzine #25 in your drive.

**LOAD HELLO**

**0**

**3**

**SAVE HELLO**

The disk is deprotected.

Softkey for...

## Microzine #26

Scholastic

Microzine #26 is normal on all even tracks and has an altered address prolog of \$D4 on all odd tracks. The solution can be seen by looking at an article by Bill Jetzer in COMPUTIST #53 on page 7. There, Bill gives the modifications needed to cause Super IOB 1.5 to read from every other track. In addition, he offers two controllers. We appropriate the second, "Colonial Conquest/Blue Powder, Grey Smoke" (only the REM statement in ours is different). See that article for explanations.

*Note: Look at lines 1020 and 1030. Bill has shown us a powerful technique in these two lines. Consider, if the protection scheme is different for every track, you can simply set up a different line for each (35 total for DOS 3.3) with the appropriate PEEKs and POKEs. Read COMPUTIST #53 for his explanation; it is truly something to consider at length.*

The following assumes you have two 5 1/4" disk drives. If you have only one, modify as appropriate.

**1** Use your favorite DOS 3.3 text editor (Applewriter DOS 3.3 is perfect) to create the following SINGLE line and store it as "CONTROLLER TO TEXT" on the disk that holds your copy of Super IOB. *Note the space before the zero!*

```
*0 D$ = CHR$(4) : INPUT "CONTROLLER NAME?": A$:
PRINT D$"OPEN" A$ ".CON": PRINT D$"WRITE"
A$ ".CON": LIST 1 - : PRINT D$"CLOSE": END
```

**2** Boot your DOS 3.3 System Master disk.

**3** Put the Super IOB disk back into drive 1. Type in the controller and save it. (Saves your typing in case things go wrong later.)

**FP**

### Controller

```
1000 REM CON.MICROZINE #26
1010 ST = 15:LS = 15:CD = WR:FAST = 1:GOSUB 640
```

```
1020 TK = 0:LT = 36:GOSUB 1050
1030 TK = 1:LT = 35:GOSUB 1050
1040 HOME : PRINT "COPY DONE" : END
1045 REM USE "TK/2 = ' IF D4 IS ON EVEN TRACKS
1046 REM USE "TK/2 > ' IF D4 IS ON ODD TRACKS (OUR
CASE)
1050 POKE 47426,24: IF TK / 2 > INT (TK / 2) THEN
POKE 47445,212
1060 GOSUB 490:GOSUB 610
1070 GOSUB 230:GOSUB 490:GOSUB 610: IF PEEK
(TRK) = LT THEN RETURN
1080 TK = PEEK (TRK):ST = PEEK (SCT):GOTO 1050
```

### Checksums

1000 - \$356B	1046 - \$8E1C
1010 - \$0A5A	1050 - \$4045
1020 - \$CF7E	1060 - \$4865
1030 - \$0C52	1070 - \$5F45
1040 - \$E9AB	1080 - \$ADAA
1045 - \$A2E6	

### SAVE CON.MICROZINE.26

**4** Execute the file you made in step 1. This will insert the line you saved as a text file in step 1 into the controller as line zero. Then run it.

### EXEC CONTROLLER TO TEXT RUN

**MICROZINE.26** *When it asks*

This will save the file as "MICROZINE.26 .CON" on the disk (without line zero). Note that text controllers have a suffix of "CON" while AppleSoft controllers have a prefix of "CON."

**5** Load Super IOB and add the following lines.

### LOAD SUPER IOB 1.5

```
630 REM READ/WRITE EVERY OTHER TRACK
640 POKE 904,32 : POKE 905,168 : POKE 906,3
: POKE 936,238 : POKE 937,14 : POKE
938,3 : POKE 939,238 : POKE 940,14 :
POKE 941,3 : POKE 942,96 : RETURN
```

### 650 REM READ/WRITE EVERY TRACK

```
660 POKE 904,238 : POKE 905,14 : POKE 906,3
: RETURN
```

### SAVE SUPER IOB 1.5

These lines modify SUPER IOB so it can read/write every other track for the this and future controllers.

**6** Now merge the Microzine 26 controller and run the program.

### EXEC MICROZINE.26.CON RUN

Follow directions and answer drive questions. Answer "Y" to format the destination disk. Remove the Super IOB disk.

**7** Put Microzine #26 in drive 1 and press **RETURN**. When the message COPY DONE is printed, remove the original MZ #26 and replace with your DOS 3.3 System Master and boot it.

### PR#6

**RUN MASTER** *new Master Create*

The greetings program is "HELLO" when prompted.

**8** Remove the System Master disk and put the copy in drive 1 when prompted. The disk is deprotected.

Ted E. Bare

Softkey for...

## One on One

Electronic Arts

I tried the softkey in COMPUTIST #49 on my version of One on One but it crashed at track \$03. I had already copied with Copy II Plus but I wanted it deprotected completely. I loaded Copy II Plus and checked the parameters they used for copying and found that tracks \$03-\$06 were skipped completely. I modified the COMPUTIST #49 controller to skip to track \$07 after track \$02 and it works perfectly. The sector edits are a little different also. Here is the modified controller and a table showing bytes changed.

*(These edits are done automatically by the controller and are shown here only for clarity, you don't need to do them yourself.)*

Trk	Sct	Byte(s)	From	To
\$02	\$03	\$47	\$BB	\$AA
		\$03	\$51	\$AD
\$01	\$06	\$08	\$03	\$62
\$0C	\$04	\$00-\$09	4C 69 BD A0 20 A0 03 A2 FF A9 88 F0 58 AD EC 09 18 60 04 1E	

Keep up the good work, the magazine looks fine without the glossy cover. Like they say 'you can't judge a book by its cover'.

I would also like to say I would like to see the hardware corner kept on. A project that might be interesting would be a digital readout of the track and sector your disk drive is on.

*I know you haven't heard much from the hardware corner, but we are working hard to turn Bobby's notes into complete projects, with step-by-step assembly instructions and a finished board. Unfortunately, that sort of talent is usually working full time already and can only spare a little each week on a volunteer basis. Hang in there.*

..... RDEXed

## Controller

```

1000 REM ONE ON ONE (NEW VERSION)
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 = 2) * 4: 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 = 2) * 4: 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 13*CHANGES
5020 DATA 2, 3, 71, 170
5030 DATA 2, 3, 81, 173
5040 DATA 1, 6, 8, 98
5050 DATA 12, 4, 0, 160
5060 DATA 12, 4, 1, 3
5070 DATA 12, 4, 2, 162
5080 DATA 12, 4, 3, 255
5090 DATA 12, 4, 4, 169
6000 DATA 12, 4, 5, 9
6010 DATA 12, 4, 6, 24
6020 DATA 12, 4, 7, 96
6030 DATA 12, 4, 8, 4
6040 DATA 12, 4, 9, 30
    
```

## Checksums

1000	- \$356B	5020	- \$06C8
1010	- \$3266	5030	- \$6604
1020	- \$2E67	5040	- \$1D08
1030	- \$3866	5050	- \$C448
1035	- \$7B0B	5060	- \$3865
1040	- \$2B80	5070	- \$0357
1050	- \$777A	5080	- \$8138
1060	- \$E3BC	5090	- \$1996
1070	- \$EBB9	6000	- \$76B9
1080	- \$FF3B	6010	- \$D559
1090	- \$46DD	6020	- \$60D6
1100	- \$4F2D	6030	- \$3E5D
5000	- \$CB59	6040	- \$4D9B
5010	- \$458A		

Bill Jetzer

Softkey for...

**Science Toolkit**  
Broderbund

## ■ Requirements

- Super IOB 1.5
- Sector editor
- 1 blank disk side for the Master Module
- 1 blank disk side for each other Module

A teacher at my school recently asked me to copy Broderbund's Science Toolkit Master Module and Module 3: Body Lab. Both disks were protected, but I managed to crack them.

In keeping with the Broderbund tradition, the protection was quite good. Both disks' epilogues had been changed from DE AA to DE BB, and both disks contained an RWTS routine which, when loaded, was located at \$9800 as opposed to the normal \$B800. The Master Module also has a few special surprises—tracks \$21-\$22 aren't formatted and track \$01 doesn't contain normal sectors.

To deprotect Module 3, type in the second controller at the end of the article and save it. Merge it with Super IOB 1.5 and run it. The 4 sector edits are made to the disk's RWTS. Since the RWTS is located at the same place on both of the disks I cracked, I am assuming that it is located there on all other module disks as well and, if so, that the controller should work with all other modules.

Deprotecting the Master Module was a little more involved. I knew that to get the data from the abnormal track I had to let the original disk do the work. But since the disk wasn't under DOS 3.3 or ProDOS, I couldn't make a file out of the data that was read in. However, since only 5 pages of data were read from the abnormal track, and there were 7 sectors free on track \$00, I simply saved the data in sectors \$09-0D and modified the code on the boot sector to load in the extra sectors. But I'm getting ahead of myself.

**1** Type in the controller at the end of the article and save it. Merge it with Super IOB 1.5 and run it. It will copy tracks \$00 and \$02-20.

Now comes the hard part—getting the abnormal track data off the original and onto the copy. Here is the procedure:

**2** Get into the monitor and move the ROM boot code in ROM to a lower page of memory so that you can modify it:

**CALL-151**  
**9600<C600.C6FFM**

**3** Have the boot code make changes so that the page \$03 vectors stay intact and the program jumps to the monitor after the first four pages of data have been read in from track \$01.

**96F8:A9 4C 8D 04 08 A9 3C 8D**  
**9700:05 08 A9 08 8D 06 08 A9**  
**9708:59 8D 80 08 A9 FF 8D 81**  
**9710:08 4C 01 08**

**9600G** *put disk in drive before pressing RETURN*

**4** Part of the code that was just read in

is used to read the fifth page of data into memory at \$8000. Install a break point and read in the last page:

**100B:00**  
**1000G**  
**COE8**

*turn drive motor off*

**5** Move the fifth page of data to the end of the first four pages and put the original byte back at \$100B.

**1400<800.8FFM**  
**100B:A9**

**6** Since you will be able to make as many backups as you need, disable the backup utility option (which wouldn't work properly on the deprotected disk anyway).

**1069:00**  
**1073:00**

**7** The way the sectors are loaded in, the first four pages don't need to be relocated when the copy is booted. However, the fifth page must be moved. To do that, just replace the routine that originally read in the fifth page of data with a simple move routine.

**1100:A2 00 8D 00 14 9D 00 08**  
**1108:E8 D0 F7 60**

**8** Insert a normal DOS 3.3 disk and boot it:

**C600G**

**9** To get the data onto the copy, type in the following program. It uses DOS's RWTS routine to write the 5 pages of data onto sectors \$09-0D.

**300:A0 15 A9 03 20 D9 03 EE**  
**308:1A 03 EE 1E 03 AD 1E 03**  
**310:C9 15 D0 EC 60 01 60 01**  
**318:00 00 09 26 03 00 10 00**  
**320:00 02 00 00 60 01 00 01**  
**328:EF D8**

**10** Insert the copy and execute the write routine.

**300G**

**11** The hard part is over. Now all that remains to be done is perform a few sector edits. Start up your sector editor and insert your copy. Science Toolkit uses the same type of routine as DOS to load in extra sectors during the boot process. It is a simple matter to load in the extra five sectors:

Trk	Sct	Byte(s)	From	To
\$00	\$00	\$62-67	FF 0E 0C 0A 08	0C 0A 08 06 04
			06	FF

**12** Lastly, change the epilogue read routines to only check for a DE. This allows the copy to read not only from itself, but from the originals too. Note that the last four edits

are the same edits performed on the extra modules disks:

Trk	Sct	Byte(s)	From	To
\$00	\$06	\$65-66	EA BD	18 60
\$08	\$08	\$65-66	EA BD	18 60
\$0C	\$09	\$38-39	EA BD	18 60
		\$94-95	EA BD	18 60

All done. Put your originals in a safe place (so the students don't play frisbee with them). If you have several computers in your lab, you can COPYA as many as you like.

## Controller

```
1000 REM SCIENCE TOOLKIT
1010 ST = 15:LS = 15:CD = WR:FAST = 1
1011 TK = 0:LT = 1:GOSUB 1020
1012 TK = 2:LT = 33:GOSUB 1020:GOTO 1050
1020 POKE 47426,24:GOSUB 490:GOSUB 610
1030 POKE 47426,56:GOSUB 490:GOSUB 610:IF
    PEEK (TRK) = LT THEN RETURN
1040 TK = PEEK (TRK):ST = PEEK (SCT):GOTO 1020
1050 HOME:PRINT "COPY^DONE":END
```

## Checksums

1000 - \$356B	1020 - \$2B37
1010 - \$88AA	1030 - \$3A5B
1011 - \$9C2D	1040 - \$AEF4
1012 - \$E95A	1050 - \$9E43

## Controller

```
1000 REM SCIENCE TOOLKIT MODULE DISKS
1010 TK = 0:LT = 35:ST = 15:LS = 15:CD = WR:FAST
    = 1
1020 POKE 47426,24:GOSUB 490:GOSUB 610:T1 =
    TK:TK = PEEK (TRK) - 1:RESTORE:GOSUB
    310:TK = T1
1030 POKE 47426,56:GOSUB 490:GOSUB 610:IF
    PEEK (TRK) = LT THEN 1050
1040 TK = PEEK (TRK):ST = PEEK (SCT):GOTO 1020
1050 HOME:PRINT "COPY^DONE":END
5000 DATA 4^CHANGES
5010 DATA 12,9,56,24
5020 DATA 12,9,57,96
5030 DATA 12,9,148,24
5040 DATA 12,9,149,96
```

## Checksums

1000 - \$356B	5000 - \$4869
1010 - \$2544	5010 - \$31B9
1020 - \$F082	5020 - \$44B0
1030 - \$C09E	5030 - \$3E8E
1040 - \$FF6C	5040 - \$2AA5
1050 - \$BB79	

## Search command for Apple II's

This routine will work for any Apple II except the enhanced Ie or IIs. They already have their own search routine.

As you may or may not know, the enhanced Ie contains a nifty search command in its \$F800 ROM. Even if you don't have an enhanced Ie, you can take advantage of this 26 byte, fully relocatable routine.

### Search Syntax

Search is called with a monitor command of the form `HHLL<ADR1.ADR2 [SY]` in which HH is the hi-byte, LL is the lo-byte, ADR1 is the start of the block you want to search and ADR2 is the end of the block. LL precedes HH in actual memory. If HH is \$00 or omitted (LL<ADR1.ADR2 [SY]), then the single byte LL is searched for. You cannot search for a two byte pair with a high byte of \$00. A list of all addresses containing the specified pattern is displayed.

I find it useful when trying to locate a string of text or a nibble count routine. You'd be surprised how valuable such a little program can be. To use it, type in the following hex code. Since the routine is fully relocatable, you can type it in at any address; I just used \$3000 as an example:

```
300:A0 01 A5 43 F0 04 D1 3C
308:D0 0A 88 A5 42 D1 3C D0
310:03 20 92 FD 20 BA FC 90
318:E7 60
```

To hook up the program through the [SY] vector, put a JMP opcode (\$4C) at \$3F8, and the address (lo byte, hi byte) of the routine at \$3F9-3FA. For \$3000, you would type:

```
3F8:4C 00 03
```

I'm sure that once you're used to it, you will find many situations where it will come in handy.

Gary Verbuch & Nemoran Pierre

## Zip Chip finally ships

Yes, the Zip Chip has finally started shipping and I was beginning to wonder if it was coming by mule. From what I understand, there were some problems with the yield after a wafer-run (more bad chips than good) which caused the shipping delay. Being an engineer I can understand the problem and apparently everything has been corrected. As a consumer I was starting to become impatient.

Well, the chip finally arrived and it was time to put it thru some testing. If there are any

problems with the device then I would like to know about them now. My main concerns are compatibility with available software and system hardware, and the actual speed improvements

Installation of the Zip Chip consists of removing the 65C02 and replacing it with the new processor. A chip removal tool is included to avoid any damage in the process.

For the readers who are not interested in the technology behind the Zip Chip, my comments and final conclusions are at the end of this article.

Zip Technology has developed a 65C02 processor to run at 4 megahertz with a 16K cache memory, whereas the processor in your Apple II, II+ or IIE is clocked at 1 megahertz without memory caching. If you are unfamiliar with the terms memory cache or system clock I will briefly cover each term. Since there are so many facets to consider when reviewing processor caching I've decided not to cover it in depth. Instead, I will let the test results be the determining factor as to the effectiveness on the Apple.

The microprocessor can be considered the heart of the computer while the clock controls it's heartbeat. During the design process, the clock rate is a vital consideration. The original 6502 processor was designed to run at 1 megahertz, although there are now 2 megahertz versions. As integrated circuit technology improved, the 65C02 was developed. Now the processor draws less current, has 10 additional instructions and can be clocked at 1 or 2 megahertz. All enhanced Apple IIE's and 2c versions contain this improved 6502.

Since the clock controls the heartbeat of the microprocessor, so to speak, it also determines the execution time, in cycles (1 microsecond), of an instruction. The most complex 6502 instruction takes seven clock cycles or 7 microseconds to execute. One important point I would like to make clear at this time is that the clock rate is not necessarily the definition of a systems speed. For example, I've run comparison tests on the 6502 running at 1 megahertz against the Z80 and the 8085 microprocessors clocked at 2 megahertz. Even though the Z80 and 8085 are clocked at twice the speed of the 6502 the results were comparable. A note about the tests, I ran the 6502 tests on the KIM-1 in early 1978. The Z80 system used was the Exidy Sourceror which came on the market shortly after the original Apple II and the 8085 was an S-100 board distributed by Cromemco.

Cache memory is relatively new to microprocessor based systems. Although there are different approaches to caching a system I will concentrate on processor caching used by the Zip Chip. The Zip Chip contains 16K of cache memory which is divided into two 8K banks, the Tag cache and the Data cache. Both

# Readers Data Exchange

banks are referred to as fast RAM, where interaction between the processor and these two banks can be up to four times the access time of the RAM external to the microprocessor. Remember, the 16K bank is contained on the Zip Chip and is not part of Apple's main memory. This RAM is available to the processor only and cannot be accessed thru software. During program execution, the Zip Chip polls the tag cache to determine whether the data required is in the data cache. Here is what is referred to as a cache hit or miss. If the data is not present the processor has to slow down to normal speed to read main memory and is referred to as a cache miss. A cache hit is when all the necessary data is contained in the data cache and main memory is not referenced. The amount of cache hits and misses will determine the actual run time of a program. Although the actual size of the Zip Chips cache is only 16K, it can take advantage of the fact that regardless of the type or length of a program, there will always be certain portions of code which is executed more than others. A program that can be contained in the data cache will run at full speed because the Zip Chip never has to slow down to reference main memory.

A common question at this point is why don't they just clock the whole system at 4 megahertz and utilize main memory. Unfortunately, the Apple was not designed to be clocked that fast and would have to be redesigned.

There are six softswitch registers available to the programmer to control certain aspects of the Zip Chip. I will include a general description of each switch rather than describe every bit pattern's effect.

## Address Function

- \$C05A** Lock or unlock the Zip Chip. Protect configuration changes.
- \$C05B** A write enables the chip & a read returns the current RAM size. Fast or sync mode, fast or normal paddle mode and control over cache updating.
- \$C05C** Control speed of slots 1-7 (fast/normal) and speaker toggle (fast/normal).
- \$C05D** Control over cache speed. (1.1111 Mhz to 4.0000 Mhz)
- \$C05E** Read Apple softswitches (control synchronous sequences). Also used to ensure your code will execute at the system speed selected.
- \$C05F** Control the caching of language card memory or bank switched memory cards.

The Zip Chip comes with a disk containing diagnostic and configuration software. The configuration program included allows you to alter the above switches without writing any code. If at any time you would prefer the Apple to run at normal speed, pressing ESC immediately after a cold or warm start will set the Zip Chip to run at 1 megahertz.

Enough about caching and softswitches, lets continue on to software compatibility. I have quite a large collection of software with just about every type of program imaginable. The easiest approach to ensuring compatibility is to simply run the software. After booting up and running approximately 50 different products, I'm convinced that Zip Technology left no stones unturned. Quite a few of the programs I ran utilize auxillary memory and every graphic mode available on the Apple has been accessed. Every program ran without incident. There appears to be no problems with hardware compatibility either.

Since some software for the Apple takes forever to boot, I'll include a time comparison table for a few popular programs. Before I continue, the Zip Chip can destroy a disk if you try to configure slot 6 faster than normal. DOS uses time critical code that can't be tampered with. In most cases, there will still be some increases in the boot process or disk access. Remember, even during a boot, there is code executing in memory.

	Ile w/ Zip Chip	Standard Ile
Appleworks	17 seconds	18 seconds
World Games	18.2 seconds	28 seconds
California Games	17.2 seconds	20.2 seconds

As you can see from the above comparisons, disk access is very unpredictable. If your main concern is to speed up disk I/O then the Zip Chip is probably not for you. There are disk caching products on the market that address that issue.

Well, so far the Zip Chip is everything Zip Technology claims. Although hardware and software compatibility are important, speed improvement is the reason for purchasing the processor.

At this point it was time to decide what method to use to test the Zip Chip's speed. The most common benchmarks utilize a couple of mathematical approaches, such as the Sieve of Eratosthenes or the Fibonacci benchmark. I've decided to use Appleworks for two reasons, first, I feel the results of one iteration of the Sieve or calculating the 24th Fibonacci number 100 times would be abstract to some readers and second, Appleworks is probably the most popular application and the results would be of more interest.

Pierre and I decided to test the Zip Chip with a simple Appleworks sort and search. Since a standard Apple IIe contains only 128K of ram we used an Applied Engineering Ram card for testing. The Zip Chip is capable of caching up to 2 megabytes of auxillary memory.

The database file used contains 1547 records consisting of 30 fields. Each field used for the alphabetic sort contained on an average of 15 characters and the numerical sort used a field with 5 digits. The text search scanned every record for the occurrence of a name which

exists only once within the database. Similar tests were run on a spreadsheet containing 181 names with 28 fields per name.

The last entry in the table is the results of a 1k Applesoft program that calculates and displays the hex and binary digits of the decimal numbers from 0 to 255.

## Time Comparison Tables (in seconds)

	Ile	Ile+Zip	Iigs
<b>Data Base file (129K)</b>			
Alpha sort (A-Z)	10	3.2	3.3
(Z-A)	10	3.2	3.3
Number sort (0-9)	14.5	4.2	4.8
(9-0)	13.6	4.1	4.8
Text search	49.9	14.6	18.9
<b>Spreadsheet file 56K</b>			
Alpha sort (A-Z)	9.5	3.0	3.3
(Z-A)	10	3.2	3.2
Text search	6.4	2.0	2.1
<b>Number Conversion program</b>			
	151	66	70

## Comments

We did run quite a few other tests, but since the results stayed consistent with the above table we chose not to include all of them. The Zip Chip consistently ran 2 to 3.5 times faster than a standard IIe. Speed improvements over the Iigs were minimal in all tests.

I originally set aside one piece of paper to record any annoyances or problems that occurred during testing. After two weeks there are only two entries on my problem sheet, and both entries are very minor annoyances.

1). During a warm boot (**⌘RESET**) you must hold the open apple key down for approximately two seconds before the boot is actually initialized.

2). When deprotecting software it is often necessary to reset out of the program. Hitting **⌘RESET** in rapid succession is as effective as tapping these keys once.

## Conclusion

If you want to increase the speed of your Apple IIe or IIc considerably without spending the money for an Accelerator card then I highly recommend the Zip Chip. I could not find one notable flaw with the processor and every program tested ran considerably faster than normal.

Installation of the device is simple and can be completed in a matter of minutes. Zip Technology offers a 30 day money back guarantee if not completely satisfied within that time period.

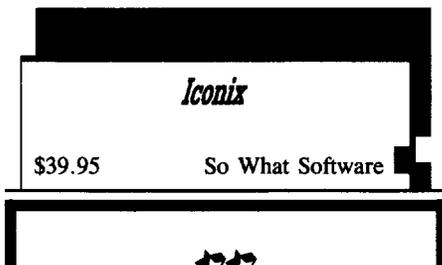
For the price of \$130.00 you will possess an Apple with a brand new personality.

Jeff Hurlburt

## The PRODUCT MONITOR

### Ratings

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



#### Requires:

- 512K Apple IIgs
- one 3½" drive
- second drive optional

No doubt you have been told that inside the IIgs are all sorts of nifty Toolbox routines which could do wonders for your BASIC/ProDOS 8 programming, if only they were easier to get to! Well the same notion occurred to the people at So What Software— as in “so what if you are supposed to use ProDOS 16”—; and the first result is a super-res text/graphics-to-BASIC interface named “Iconix”. Now you can load and display super-res screens, present messages in colored text, and manipulate user-designed icons— all, together with mouse point-and-click input, without leaving the friendly realms of ProDOS 8 and in-ROM BASIC!

The “Iconix” package consists of a 3½” diskette on which you find a thirty-plus page manual (for on-screen perusal or dumping to your printer), assorted editors, icon templates, example files, and ICONIX.GS. The latter, a mere six blocks in length, is the real workhorse

of the system and the only file you will need to transfer in order to access “Iconix” features. Unlike some ‘power programming’ add-ons, ICONIX.GS does not gobble huge chunks of Bank 0 memory or fill up your applications diskettes with accessory files. It DOES insist upon being the first Memory Manager client (i.e. you can’t boot under Diversi Cache) and will use about 8.5K in Bank 0 for buffers.

Starting with an otherwise blank bootable ProDOS 8 diskette, the simplest “Iconix” implementation requires ICONIX.GS and a BASIC program which, at some point, does a “BRUN ICONIX.GS”. Using the POKEs and CALLs outlined in the manual, you could switch the super-res screen on and off, do screen clears, create and ‘flash’ (i.e. invert) variously sized color bars, and move/read the mouse cursor. Not very interesting; but add super-res screens; and viola!, your mouse has something to point to and flashing a portion of the screen can mean something. An “SHGR Bloader” routine in ICONIX.GS lets you load-in one or more standard unpacked super-res files, one per available memory bank. Displaying any loaded screen is then just a matter of POKeing its bank number and doing a CALL.

For some applications, ICONIX.GS plus a few super-res frames may be all you will need. Most programs, however, must be able to generate text and at least some graphics ‘on the fly’; AND, since you’re on a IIgs, these might as well be in full-color super-res! ICONIX.GS gets its text font and icons from a special kind of super-res screen called an “icon sheet”. This is a frame having a font set plus up to fourteen icons, which you create using any of the popular painters, all on a black, (i.e. ‘transparent’ color 0) background. This way Toolbox routines can ‘lift off’ font characters and pictures from your ‘sheet’.

The current version of “Iconix” (2.1) is very picky about font block location (upper left) and character size (8x8 pixels), which explains, in part, why a “Blank Master” sheet with text font in place is supplied on-diskette. Your pictures, on the other hand, can vary widely in size and go just about anywhere— a very nice feature, since practically any ‘clip art’ graphic becomes an icon candidate. The only catch here is that “Iconix” expects the user to tell it, via the “Icon Edit” utility, where each icon is to be found on the sheet. This entails selecting a number identifier from the editor screen; and then, on a sheet display, clicking opposite corners of the space containing the icon picture(s). With the resulting “icon params” added, your new icon sheet is ready to be saved for future use.

Once a BASIC program loads in an icon sheet and POKEs an “icon bank” pointer, all of that sheet’s icons are available for pasting, moving, flashing, etc. anywhere on the screen,

even on top of each other! Similarly, text, in the on-sheet font and current palette colors, can start anywhere. For example: 100 SG\$= “Welcome to Iconix Tester” plus a few POKEs (to set location and color) and a CALL will display the message in the selected color at the screen location specified.

Thanks to an excellent tutorial, you can begin using ICONIX.GS in your BASIC programs almost immediately. On-diskette menu and editor utilities are all in BASIC, employ ICONIX.GS, and serve as helpful examples. In fact, slicing off the unneeded part of one such program, as done in the tutorial, leaves a kind of ‘Blank Master’ program. You have all the lines you need to start a new application; and sticking with the variable names defined in the examples (e.g. “DS” for the “Display Screen” CALL address, etc.) makes it easier to treat the ICONIX.GS interface like a set of newly acquired BASIC super-res commands.

So, can you expect ‘smooth sailing’ in your “Iconix” programming experiences? Probably not. The culprits here do not include ICONIX.GS itself— in the few short programs I’ve written using the interface, everything seems to work at least as well as advertised—; nor is there any ground for complaint regarding nuts-and-bolts documentation. The easy-reading manual is well-organized, lists all PEEKs, POKEs, and CALLs; and is very good about telling the user what does what. Unfortunately, once you get past the tutorial, “Iconix” is not nearly so effective in addressing the do’s and don’t’s, planning, and other concerns relating to ‘full-blown’ applications.

As an example of the above: reading the “Iconix” manual, it’s easy to get the impression that a user has almost complete freedom of palette colors selection. Nothing is said about the need for palette consistency among icon sheets and loaded-in screens; and choice of icon sheet background color is treated like a wide-open option. That, for some reason, the “Blank Master” sheet comes with a white background does not, of course, help matters. (Don’t forget to do a black/color 0 fill before you start painting.)

Granted, some “Iconix” users will find super-res applications development no more difficult than using the built-in BASIC hires commands, probably a good deal easier. No one, however, is likely to get through the on-diskette utilities without reaching, instinctively, for a can of Black Flag. The worst offenders are “Palette Edit”, for adjusting colors among the sixteen palettes, and “SCB Edit”, for setting resolution and palette on a line-by-line basis. Aside from observing that, as isolated functions, such capabilities are virtually useless, and that “Palette Edit”’s controls are glitchy, there is the problem that running either utility is like rolling dice. Sometimes, a perfectly good

picture file will bomb the program; sometimes not. Your finished work may be saved; but, then again, maybe not; and, if it is saved, there is no guarantee that it will be under the file name you expect. So much for 'bad news'. The 'good news' is that neither utility has a thing to do with accessing ICONIX.GS features. Playing picture file roulette is entirely optional.

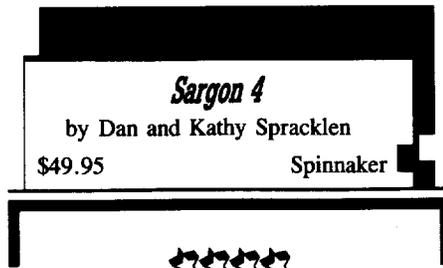
Working with "Icon Edit", your last step in icon sheet preparation is, generally, quick (one or two minutes), easy, and even fun. Still, the program fails to check the limits you click-in for each icon (too big, and it bombs); nor does it protect itself from the stuff 640-mode painters seem to dump into the "unused" portion of a picture file (i.e. again, a bomb).

Depending upon your painter choice, "Icon Edit" can also complicate the process of revising/adding-to icon sheets. While the utility will load both PIC (\$C1) and BIN type unpacked picture files, it always saves sheets as type BIN. Among popular painters, only "816 Paint" and "Deluxe Paint II" will accept BIN files; "The Graphics Studio" and "Paintworks Gold" will not. Finally, even "Icon Edit" is not 100% reliable when it comes to saves; so it's a good idea to save your stuff under a new name and do a CATALOG before exiting.

"How did you get a ProDOS 16 program to load so fast?!" Admittedly, just such a reaction from computer room visitors comes close to paying back an investment in So What's "Iconix". But, then, you won't need anyone to tell you that your program's super-res displays and mouse interface have the look and feel of a commercial product. Just two 'warnings': (1) in case you decide to rewrite the "Iconix" utilities, expect some competition from the programming fanatics at So What. Once this review arrives, you can be sure an upgrade will be in the works. (2) Working with "Iconix" can be as addictive as any game. This is one package you can 'get your teeth into'; a boon for beginning programmers, and positively a feast for veterans!

## The State of Apple Chess

Since Apple II chess has been in one of its periodic doldrums, when Leslie Daley at Spinnaker asked if I would like to review "Sargon 4" for the Macintosh, there was little hesitation: "Sure, why not?" Plenty of Mac users own a II and read Computist. Besides which, a look at the new program vis-a-vis established chess players running on the IIGs is bound to be of interest.



### Requires:

- Macintosh
- one 3 1/2" drive

Reaffirming "Sargon's status as the most feature-laden of all personal computer opponents", the latest edition offers a variety of playing levels (5 seconds through 10 minutes per move and "infinite") plus strength modifier options to limit search depth, restrict computer 'thinking time' to its own move, and turn off access to the 68,000 position opening 'library'. Add an eight-level 'all the moves' mode (i.e. a player has 5 minutes, ... 3 hours to make all of his or her moves in a game) and "Sargon 4" delivers practically every tournament and 'skittles' setup one could wish for.

A frequent user complaint is that computer chess programs don't DO ANYTHING, except, of course, make moves. (I have long suspected that one of the reasons computers seem to play as well as they do is that, at a minute or more per move, they bore their human opponents into committing errors.) Well, "Sargon 4" won't moan, squirm, or chortle with glee; but its "Window on the Search", Score, and search tree displays— all on the same screen as the 2-D board, move list, and clocks— amount to a fair substitute.

As in football and presidential debates, much of the fun in chess comes from analysis. Aside from the expected take-back, change sides, hint, game load/save, and mate-finding options, "Sargon 4" will let you transfer a game's move list to clipboard memory and/or obtain a printout. (Unfortunately, there is no provision for saving or printout of the actual board display.) A flexible Replay mode not only permits timed "automatic" or user-stepped move advance, but also allows halting at any point with an option to continue the game from the current position. Along with your own games, on diskette Replay candidates include over one hundred "Great Games" plus as nearly as many problems.

In an effort to better mimic 'the real thing', most Apple chess programs since "Mychess II" have offered a simulated 3-D board display. "Sargon 4" comes with five piece sets, color selection (!) on the Mac II, AND a design-your-own-set "Sculptor" utility. While it is the best,

clearest 'real board' simulation yet, I predict most players will stick with the main screen's 2-D format, especially since, in 3-D, you lose the search and move list displays.

So much for frills (albeit nice ones); what about the 'bottom line'? Well, according to Spinnaker, "Sargon 4 beats Chessmaster 2000 seven out of eight games"— decisive, but not altogether surprising since the tests were conducted with "Chessmaster 2000" running on a IIe. How, one wonders, would "Sargon 4" fare against the better II series programs running on a IIGs?!

After obtaining the loan of a Mac Plus, the next problem in setting up the intra-Apple confrontation was matching time spent per move. Surprisingly, with the IIGs on its fast setting, the divisor into documented "Chessmaster 2000" timings was consistently around 4.0. In a 51-move game at Level 10 ("3 min/move"), "Chessmaster 2000" actually averaged 42.2 seconds/move; in another game at Level 7 ("2 min/move") the average was 27.6 seconds/move, ... etc.. I ran "Sargon 4" with 'think on opponent's time', "Window on Search", and opening library features 'ON', because this is the setup most players will probably want to adopt. The high-activity 'show search tree' option was 'OFF'; to minimize the cost, in thinking time, of display updating. Results of a seven-game mini-tournament, are shown below. (Games are in the order played.)

Sargon 4	Chessmaster	Winner
30s/mv W	L6 B	draw
30s/mv B	L6 W	draw
1m/mv W	L9 B	draw
1m/mv B	L10 W	Sargon
1m/mv W	L10 B	CM
30s/mv W	L7 B	CM
30s/mv B	L7 W	Sargon

Result: Sargon 3.5 / CM 3.5

Anticipating that fans of other II series programs might feel left out, I also tried "Sargon III" and "Mychess II" against "Sargon 4" at 1 min/move. (Since "Colossus IV" does not run properly on the IIGs, it was disqualified.) At Level 5, "Sargon III" just about matches "Sargon 4" speed, but not its smarts. (Evidently, the claimed "new algorithms" work.) Playing white, "Sargon 4" triumphed after a hard-fought contest. "Mychess II", at Level 7, averages a few seconds below a minute per move. In two tight games, the results were even, with white winning in each case.

Naturally, as a IIGs partisan, I would like to read the above as solid evidence that the current best in IIGs-compatible chess is the equal of Mac's best chess player. I would, except that experience playing each of the programs has

convinced me that 'it just ain't so'. Granted, "Sargon 4" isn't going to overwhelm "Chessmaster 2000" or "MyChess II"; but, it is noticeably the toughest opponent. A tournament-player friend (USCF "A" rated) came to the same conclusion. Predictably, best at tactics and a tad weak in positional play, "Sargon 4" does, nevertheless, seem to play a more integrated, 'human-like' game than, for example, "Chessmaster 2000". The higher the level, especially around, 'tournament speed' settings, the less 'computer like', more entertaining its play becomes. Clearly, if you own a Mac, you're in luck. For now, the BEST in Apple chess is "Sargon 4".

## FAST FRAMES, UPDATES, ETC.

### Electing a President

There you are, digging around in your software stores for the perfect Election Night Party 'ice breaker' when suddenly you remember that neat new game. "Wow, just the thing!", you exclaim, "Boy am I glad I read Computist!" ... Hopefully, you will have already given SSI's "President Elect: 1988" (\$24.95, for 48K Apple II series) a thorough tryout. The game is a cinch to get into, with an experienced player to explain things; but starting out with just the manual (i.e. becoming the 'experienced player') takes a bit of practice.

Whether you choose candidates from the 70-plus politician data base or create your own, you're in for a realistic campaign-through-election simulation including debates, trips, advertising, polls, and 'strong/weak/toss-up' electoral maps—all culminating in dramatic east-to-west election night returns. For 1-3 players, any or all of which may be computer-directed, "President Elect" can even run 'the whole show' and predict the election's outcome! (Incidentally, SSI has recently published updated Dukakis parms. When you get the game, enter Inflation 4%, Unemployment 5.3%, GNP 3.1%, Peace, Morale 5.5, Item Ratings: 7,7,7,2,7,7,6,5,8,6,9,1,8,1,7,7,3,8,1. Speaking 5, Magnetism 5, Poise 9.)

## VENDORS

**So What Software:** 10221 Slater Avenue, Suite 103 Fountain Valley, CA 92708 (714-964-4298)

**Spinnaker:** One Kendall Square, Cambridge, MA 02139, (617-494-1200)

**Strategic Simulations Inc.:** 1046 North Rengstorff Ave., Mountain View, CA 94043 (415-964-1353)

 The following readers contributed additional data to the *Ultima V* Attributes and Items chart on page 31. Vincent Andrews, Robert Kettle, Dr. Jean-Louis Torre and Blain Johnson. Mike Horton did the actual work of combining the data and updating the chart. . . . . RDEXed

Robert Kettle

### Playing Tips for...

## Ultima V

Origin

Totally by accident I found a command not listed in the manuals that I have found to be of great assistance in playing this adventure, it is the Ctrl-@ command. This command will present you with a large multidigit number, of which the last four digits are your coordinates on the current map, be it in the kingdom of Britannia or a small village. This makes finding your way back to those special locations just a little bit easier.

### A.P.T. for...

## Ultima V

Origin

Note that for the items on block \$01D (F0-FC) to be present in your inventory you must enter a value of \$FF, with the exception of Skull Keys for which you enter a hex value equal to the number of Skull Keys desired.

Dr. Jean-Louis Torre

### A.P.T. for...

## Ultima V

Origin

When you visit houses, refrain from taking the easy to get food lying on the tables. Remember the old saying "It is too good to be true"? Well, it is. You will loose your... virtue!

Blain Johnson

### A.P.T. for...

## Ultima V

Origin

The tip in COMPUTIST #56 on page 19 that states: you must fight a shadowlord if you enter Jhelom, is not always true. If you enter a town and you get a message that an air of hatred, cowardice, or falsehood surrounds you, the town is not safe. This happens because your character is not doing well on the virtue which is associated with that town. (Look in the Book of Lore at the town descriptions.)

To enter safely, you must go to the shrine which corresponds to the town and meditate. You can enter an unsafe town, but you must

avoid the shadowlord. If you get too close, he will attack and, most likely, kill your whole party.

### Other notes:

**Attributes:** The status byte is: 00-In party, 7F-gone, FF-not in party. Any other number denotes the city they are being 'kept' in (where you leave a member).

**Level:** The optimum level is 8. Even though a character's level can be \$99, this makes battle much harder. Fighting accuracy is terrible and the monsters get 5-6 turns to your 1 turn.

**Clock/calendar:** Numbers are entered in BCD (binary coded decimal). (Example: month 11 = \$11.)

### Playing Tips for...

## Ultima V

Origin

If you look at any door handle and you see a small dot underneath the door, it is locked. If you cannot see the dot, the door is unlocked. If you have a color monitor, the fringe around the dot will be either purple or green. If it is purple, normal keys can unlock the door. If the fringe is green, you must use skull keys. Two places skull keys are needed are Lord British's Chamber and the tower in Skara Brae. If you change floors (stairs & ladders), or are attacked, all doors are changed to their original state.

⊕ I have spent many hours trying to deprotect Flight Simulator with no luck. I have purchased COMPUTIST #36, #45, and #51. It would seem that was all I needed... Wrong. After following the instructions exactly for the third time, I decided to look at the booting code to see if it might be an error I could fix. After looking at the additional code from \$869-\$8F1 added by Christopher Dean (#51), I modified it so it copied the whole F8 ROM onto the language card instead of just pages \$FC, \$FD, and \$FE. This allowed the disk to boot further, but not much further. After the program recalibrated the disk arm for the first time, it would load part of the graphic screen with garbage, change to the text screen, print "22 1D35", and then freeze. I snooped around a little and found that the garbage on the graphic screen was the simulator's DOS. I searched and searched but could not find out why it loaded the simulator's DOS in that area of memory. Could anyone help me out.

⊕ I also have had trouble trying to get a working copy of my Olympic Decathlon (#35). The whole procedure works fine and the disk will boot up to the point where you are asked if you want to begin. If the program tries disk access, it freezes. One other thing I don't understand is while booting, after the screen has turned blue, parts of the screen are blacked out. That doesn't happen with the original.

# Readers Data Exchange

**Ultima V Character data (Block \$01D, bytes \$80-17F / Track \$03, sector \$05, bytes \$80-FF and sector \$04, bytes \$00-7F)**

name	player	Shamino	Iolo	Mariah	Geoffrey	Jaana	Julia	Dupre
status	\$80	\$90	\$A0	\$B0	\$C0	\$D0	\$E0	\$F0
sex	\$81	\$91	\$A1	\$B1	\$C1	\$D1	\$E1	\$F1
class	\$82	\$92	\$A2	\$B2	\$C2	\$D2	\$E2	\$F2
health	\$83	\$93	\$A3	\$B3	\$C3	\$D3	\$E3	\$F3
strength	\$84	\$94	\$A4	\$B4	\$C4	\$D4	\$E4	\$F4
intelligence	\$85	\$95	\$A5	\$B5	\$C5	\$D5	\$E5	\$F5
dexterity	\$86	\$96	\$A6	\$B6	\$C6	\$D6	\$E6	\$F6
magic points	\$87	\$97	\$A7	\$B7	\$C7	\$D7	\$E7	\$F7
current hit points	\$88-\$89	\$98-\$99	\$A8-\$A9	\$B8-\$B9	\$C8-\$C9	\$D8-\$D9	\$E8-\$E9	\$F8-\$F9
maximum hit points	\$8A-\$8B	\$9A-\$9B	\$AA-\$AB	\$BA-\$BB	\$CA-\$CB	\$DA-\$DB	\$EA-\$EB	\$FA-\$FB
experience points	\$8C-\$8D	\$9C-\$9D	\$AC-\$AD	\$BC-\$BD	\$CC-\$CD	\$DC-\$DD	\$EC-\$ED	\$FC-\$FD
experience level	\$8E	\$9E	\$AE	\$BE	\$CE	\$DE	\$EE	\$FE
name	Katrina	Sentri	Gwenno	Johne	Gorn	Maxwell	Toshi	Saduj
status	\$100	\$110	\$120	\$130	\$140	\$150	\$160	\$170
sex	\$101	\$111	\$121	\$131	\$141	\$151	\$161	\$171
class	\$102	\$112	\$122	\$132	\$142	\$152	\$162	\$172
health	\$103	\$113	\$123	\$133	\$143	\$153	\$163	\$173
strength	\$104	\$114	\$124	\$134	\$144	\$154	\$164	\$174
intelligence	\$105	\$115	\$125	\$135	\$145	\$155	\$165	\$175
dexterity	\$106	\$116	\$126	\$136	\$146	\$156	\$166	\$176
magic points	\$107	\$117	\$127	\$137	\$147	\$157	\$167	\$177
current hit points	\$108-\$109	\$118-\$119	\$128-\$129	\$138-\$139	\$148-\$149	\$158-\$159	\$168-\$169	\$178-\$179
maximum hit points	\$10A-\$10B	\$11A-\$11B	\$12A-\$12B	\$13A-\$13B	\$14A-\$14B	\$15A-\$15B	\$16A-\$16B	\$17A-\$17B
experience points	\$10C-\$10D	\$11C-\$11D	\$12C-\$12D	\$13C-\$13D	\$14C-\$14D	\$15C-\$15D	\$16C-\$16D	\$17C-\$17D
experience level	\$10E	\$11E	\$12E	\$13E	\$14E	\$15E	\$16E	\$17E

**(Block \$01D, bytes \$180-1FF / Track \$03, sector \$04, bytes \$80-FF)**

Sex: male - \$3E, female - \$3F	\$180-181 food (0-9999)	\$1F0 shard of falsehood (00/FF)	\$1F7 skull keys (0-99)
Status: not present - \$FF, gone - \$7F, present - \$00	\$182-183 gold (0-9999)	\$1F1 shard of hatred (00/FF)	\$1F8 Lord British's amulet (00/FF)
Health: good - \$C7, poisoned - \$D0	\$184 keys (0-99)	\$1F2 shard of cowardice (00/FF)	\$1F9 Lord British's crown (00/FF)
dead - \$C4, sleeping - \$D3	\$185 gems (0-99)	\$1F3 spyglass (00/FF)	\$1FA Lord British's sceptre (00/FF)
Class: avatar - \$C1, bard - \$C2, fighter - \$C6, mage - \$CD	\$186 torches (0-99)	\$1F4 HMS Cape plans (00/FF)	\$1FB black badge (00/FF)
	\$187 grappling hook (0-1)	\$1F5 sextants (00/FF)	\$1FC wooden box (00/FF)
	\$188 magic carpets (??)	\$1F6 pocket watch (00/FF)	

**(Block \$01F / Track \$03, sector \$01)**

<b>Armour</b> 00 Leather Helm 01 Chain Coif 02 Iron Helm 03 Spiked Helm 04 Small Shield 05 Large Shield 06 Spiked Shield 07 Magic Shield 08 Jeweled Shield 09 Cloth 0A Leather 0B Ring 0C Scale 0D Chain 0E Plate 0F Mystic Armor  <b>Weapons</b> 10 Dagger 11 Sling 12 Club 13 Flaming Oil 14 Main Gauche 15 Spear 16 Throwing Axe 17 Short Sword 18 Mace 19 Morning Star 1A Bow	1B Arrows 1C Crossbow 1D Quarrels 1E Long Sword 1F 2-Handed Hammer 20 2-Handed Axe 21 2-Handed Sword 22 Halberd 23 Chaos Sword 24 Magic Bow 25 Silver Sword 26 Magic Axe 27 Glass Sword 28 Jeweled Sword 29 Mystic Sword  <b>Magic items</b> 2A Ring of Invisibility 2B Ring of Protection 2C Ring of Regeneration 2D Amulet of Turning 2E Spiked Collar 2F Ankh  <b>Spells</b> 40 In Lor <i>light</i> 41 Grav Por <i>magic missile</i> 42 An Zu <i>awaken</i> 43 An Nox <i>cure poison</i> 44 Mani <i>heal</i> 45 An Ylem <i>disintegrate objects</i>	46 An Sanct <i>unlock</i> 47 An Xen Corp <i>repel undead</i> 48 Rel Hur <i>wind change</i> 49 In Wis <i>locate</i> 4A Kal Xen <i>call animal</i> 4B In Xen Mani <i>create creature life</i> 4C Vas Lor <i>great light</i> 4D Vas Flam <i>ball of flames</i> 4E In Flam Grav <i>wall of fire</i> 4F In Nox Grav <i>wall of poison</i> 50 In Zu Grav <i>wall of sleep</i> 51 In Por <i>blink</i> 52 An Grav <i>dispell field</i> 53 In Sanct <i>protection</i> 54 In Sanct Grav <i>protection field</i> 55 Uus Por <i>upward move</i> 56 Des Por <i>downward move</i> 57 Wis Quas <i>reveal</i> 58 In Bet Xen <i>insect swarm</i> 59 An Ex Por <i>magic lock</i> 5A In Ex Por <i>magic unlock</i> 5B Vas Mani <i>great heal</i> 5C In Zu <i>sleep</i> 5D Rel Tym <i>quickness</i> 5E In Vas Por Ylem <i>tremor</i> 5F Quas An Wi <i>confuse</i> 60 In An <i>negate magic</i> 61 Wis An Ylem <i>x-ray</i> 62 An Xen Ex <i>charm</i> 63 Rel Xen Bet <i>change creature small</i>	64 Sanct Lor <i>invisibility</i> 65 Xen Corp <i>kill</i> 66 In Quas Xen <i>clone</i> 67 In Quas Wis <i>peer</i> 68 In Nox Hur <i>poison wind</i> 69 In Quas Corp <i>fear</i> 6A In Mani Corp <i>resurrect</i> 6B Kal Xen Corp <i>summon</i> 6C In Vas Grav Corp <i>cone of energy</i> 6D In Flam Hur <i>flame wind</i> 6E Vas Rel Por <i>gate travel</i> 6F An Tym <i>time stop</i>  <b>Scrolls</b> 70 VL <i>great light</i> 71 RH <i>wind change</i> 72 IS <i>protection</i> 73 IA <i>negate magic</i> 74 IFW <i>peer</i> 75 KXC <i>summon</i> 76 IMC <i>resurrect</i> 77 AT <i>time stop</i>  <b>Potions</b> 78 Blue <i>awaken</i> 79 Yellow <i>heal</i> 7A Red <i>cure</i> 7B Green <i>poison</i> 7C Orange <i>sleep</i> 7D Purple <i>change shape</i> 7E Black <i>invisibility</i>	7F White <i>quick view</i>  <b>Moonstones</b> 90 1st phase moonstone 91 2nd phase moonstone 92 3rd phase moonstone 93 4th phase moonstone 94 5th phase moonstone 95 6th phase moonstone 96 7th phase moonstone 97 8th phase moonstone  <b>Spell Ingredients</b> A0 Sulfur Ash (0-99) A1 Ginseng (0-99) A2 Garlic (0-99) A3 Spider Silk (0-99) A4 Blood Moss (0-99) A5 Black Pearl (0-99) A6 Nightshade (0-99) A7 Mandrake (0-99)  <b>Track \$03, sector \$07</b> 00 X-Coordinate 01 Y-Coordinate 1A-1B Year (0000-9999) 1C Month 1D Day 1E Hours 1F Minutes
--	--	---	--	--

# Readers Data Exchange

## Bard's Tale III — Combined Attributes and Items List

**I** This page is the result of the combined efforts of Thomas Rapheld, Brian Symonds, Harold Day, Bob Igo and Donald Jones. . . . . RDEXed

00-0F	80-8F	Name of Character	2B	AB	Sex: 00-Male, 01-Female	2F	AF	Armor Class: lowers the initial armor class of the character before normal armor is worn.		
10	90	Strength	2C	AC	Character's Picture: (Missing sequential numbers are blank pictures)	30-53	B0-D3	Items carried: the 12 items that each character can carry are stored in groups of three bytes with the first byte meaning unequipped (\$00), equipped (\$01), can't use (\$02), or special characteristics (for example \$08 means the enchanted water of life), the second byte giving the item itself, and the third byte giving the number of times the item can be used.		
11	91	Intelligence (IQ)			00 = Refugee Camp, 01 = Viper, 02 = Hobgoblin, 03 = Lizard, 04 = White Dog, 05 = Red Devil, 06 = Imp, 08 = Serpent, 09 = Nomad, 0C = Dinosaur, 13 = Wear Tiger, 15 = Fish Man, 19 = Lizard Priest, 1A = Dark Gnome, 1E = Bug Man, 1F = Giant, 20 = King on Throne, 21 = Blood Knight, 24 = Skeleton, 26 = Wizard, 27 = Mage, 28 = Ghost Armor, 2C = Lizard Mage, 2E = Tree /w Falling Apples, 2F = Old Man, 30 = Female Warrior, 31 = Robed Priest, 32 = Empty Building, 34 = Treasure, 36 = Sorcerer, 37 = Female Mage, 38 = Shore Line, 39 = Avenging Angel, 3D = Red Dragon, 45 = Study, 46 = Rock Elephant, 4A = Dead Old Man, 4E = Floating Eye, 53 = Bartender	54-55	D4-D5	Conjurer Spells		
12	92	Dexterity			Health: 00-Good, 01-Poisoned, 02-Old, 03-poisoned, 04-dead, 05-dead and go to Adventurer's Heaven	56-57	D6-D7	Mage Spells		
13	93	Constitution			2D	AD		58-59	D8-D9	Sorcerer Spells
14	94	Luck			2E	AE	???????	5A-5B	DA-DB	Wizard Spells
15-18	95-98	Experience						5C-5D	DC-DD	Archmage Spells
19-1C	99-9C	Gold						5E-60	DE-E0	Chronomancer Spells
1D-1E	9D-9E	Current Level						61-62	E1-E2	Geomancer Spells
1F-20	9F-A0	Max Level Attained						63-64	E3-E4	Misc. Spells
21-22	A1-A2	Hit Points						64	E4	Critical Hits (Hunter)
23-24	A3-A4	Condition						64-65	E4-E5	Bard's Songs
25-26	A5-A6	Current Spell Points						64-66	E4-E6	Thief Abilities
27-28	A7-A8	Max Spell Points						67-68	E7-E8	?
29	A9	Class: 00-Warrior, 01-Wizard, 02-Sorcerer, 03-Conjurer, 04-Magician, 05-Rogue, 06-Bard, 07-Paladin, 08-Hunter, 09-Monk, 0A-Archmage, 0B-Chronomancer, 0C-Geomancer, 0D-Monster, 0E-Illusion						69	E9	Multiple Attacks: critical hits modifier
2A	AA	Race: 00-Human, 01-Elf, 02-Dwarf,								

00	Empty Pockets	29	Fin's Flute	52	Breath Ring	7B	Valarian's Bow	A4	Left Key	CD	Hourglass
01	Torch	2A	Kael's Axe	53	Dragonshield	7C	Arws of Life	A5	Lever	CE	Thieves Hood
02	Lamp	2B	Mthr Arrows	54	Dmnd Plate	7D	Canteen	A6	Nut	CF	Surehand Amulet
03	Broadsword	2C	Dayblade	55	Wargloves	7E	Titan Plate	A7	Bolt #0	D0	Thieves Dart
04	Shortsword	2D	Shield Staff	56	Wizhelm	7F	Titan Shield	A8	Spanner	D1	Shrill Flute
05	Dagger	2E	Elf Cloak	57	Dragonwand	80	Titan Helm	A9	Shadowlock	D2	Angel's Harp
06	War Axe	2F	Hawkblade	58	Deathring	81	Fire Spear #0	AA	Shadow Door	D3	The Book #0
07	Halbard	30	Admt Sword	59	Crystal Sword #0	82	Willow Flute	AB	Misericorde	D4	Troth Lance
08	Long Bow	31	Admt Shield	5A	Speedboots	83	Firebrand	AC	Holy Avenger	D5	Dmnd Suit
09	Staff	32	Admt Helm	5B	Flame Horn	84	Holy Sword	AD	Shadowshiv	D6	Dmnd Flail
0A	Buckler	33	Admt Gloves	5C	Zen Arows	85	Wand of Fury	AE	Kali's Garrote	D7	Purple Heart
0B	Tower Shield	34	Pureblade	5D	Deathdrum	86	Lightstar	AF	Flame Knife	D8	Titan Bracers
0C	Leather Armor	35	Boomerang	5E	Pipes of Pan	87	Crown of Truth	B0	Red Stiletto	D9	Eelskin Tunic
0D	Chain Mail	36	Ali's Carpet	5F	Power Ring	88	Belt of Alliria	B1	Heartseeker	DA	Sorcerer's Hood
0E	Scale Armor	37	Luckshield	60	Song Axe	89	Crystal Key	B2	x #0	DB	Dmnd Staff
0F	Plate Armor	38	Dozer Fgn	61	Trick Brick	8A	Tao Ring	B3	x	DC	Crystal Gem
10	Robes	39	Admt Chain	62	Dragon Fgn	8B	Stealth Arrows #0	B4	x	DD	Wand of Force
11	Helm	3A	Death Stars	63	Mage Fgn #0	8C	Yellow Staff	B5	Dmnd Scale	DE	Cli Lyre #0
12	Leather Gloves	3B	Admt Plate #0	64	Troll Ring	8D	Steady Eye	B6	Holy TNT	DF	Youth Potion
13	Gauntlets	3C	Admt Bracers	65	Aram's Knife	8E	Divine Halbard	B7	Eternal Torch	E0	x
14	Mandolin	3D	Slayer Fgn	66	Angra's Eye	8F	Incense	B8	Oscon's Staff	E1	x
15	Spear	3E	Pure Shield	67	herb Fgn	90	l-ching	B9	Angel's Ring	E2	x
16	Arrows	3F	Mage Staff	68	Master Wand	91	White Rose	BA	Deathhorn	E3	x
17	Mthr Sword	40	War Staff	69	Brothers Fgn	92	Blue Rose	BB	Staff of Mangor	E4	x
18	Mthr Shield	41	Thief Dagger	6A	Dynamite	93	Red Rose	BC	Tesla Ring	E5	x
19	Mthr Chain	42	Soul Mace	6B	Thor's Hammer	94	Yellow Rose	BD	Dmnd Bracers #0	E6	x
1A	Mthr Scale	43	Ogrewand	6C	Stoneblade	95	Rainbow Rose #0	BE	Death Fgn	E7	x
1B	Giant Fgn	44	Kato's Bracer	6D	Holy Handgrenade	96	Magic Triangle	BF	Thunder Sword	E8	x
1C	Mthr Bracers	45	Sorcer Staff #0	6E	Masterkey	97	x	C0	Poison Dagger	E9	x #0
1D	Bardsword	46	Galt's Flute	6F	Haspin Ring	98	Hammer of Wrath	C1	Spark Blade	EA-EF	x
1E	Fire Horn	47	Frost Horn	70	Crysta Lens	99	Ferofist's Helm	C2	Galvanic Oboe	F0	Mthr Suit
1F	Litewand	48	Ag's Arrows	71	Smoky Lens	9A	x	C3	Hamonic Gem	F1	Titan Suit
20	Mthr Dagger	49	Dmnd Shield	72	Black Lens	9B	x	C4	Tung Shield	F2	Mages Gloves
21	Mthr Helm	4A	Bard Bow	73	Sphere of Lanati	9C	Helm of Justice	C5	Tung Plate	F3	Flare Crystal
22	Mthr Gloves	4B	Dmnd Helm	74	Wand of Power	9D	Sceaclu's Cloak	C6	Minstrels Gloves	F4	Holy Missile #0
23	Mthr Axe	4C	Elf Boots	75	Acorn	9E	Shadelance	C7	Hunters Cloak	F5	God's Blade
24	Shuriken	4D	Vanquisher Fgn	76	Wineskin	9F	Black's Arrows #0	C8	Death Hammer #0	F6	Hunter Blade
25	Mthr Plate	4E	Conjurstaff	77	Nightspear #0	A0	Werra's Shield	C9	Blood Mesh Robe	F7	Staff of Gods
26	Molten Fgn	4F	Staff of Lor #0	78	Tsloth's Head	A1	Strifespear	CA	Soothing Balm	F8	Horn of Gods
27	Spell Spear	50	Flame Sword	79	Tsloth's Heart	A2	Sheetmusic	CB	Mages Cloak	F9-FE	x
28	Shield Ring	51	Powerstaff	7A	Arefolia	A3	Right Key	CC	Familiar Fgn	FF	x #0

Donald Jones

A.P.T. for...

## Bard's Tale III

Electronic Arts

The characters are saved on the last two tracks (\$21-22). A text search will reveal where your character is on the disk.

Is your Mage still looking for the three miscellaneous spells? Just input a 3F 80 for bytes \$63-64 (\$E3-E4). When you start changing your warrior's into Geomancer's and you want all the spells (including the misc.), use FF for the bytes. With your magic user's change all to FF to receive max spells. To become a Chronomancer, put FF's in bytes \$5E-60 (\$DE-E0). If you want an Archmage with no Chronomancer spells, put FE at \$5D (\$DD).

Some of the abilities are stored at the same locations depending on what class your character is. For example; the Thieves abilities are stored at bytes \$64-65-66, whereas the Bard's songs are stored at bytes \$64-65. These are also the locations for the miscellaneous spells for all magic users.

Ⓢ A challenge for you graphics people. Can you find a way to make your own picture and store it to the character disk? The article in COMPUTIST #53 by Greg Poulos might give you a clue. It tells how to use E.O.A. RWTS. The pictures available for your characters are limited. For example they only give two types of Females to use. It would be nice to personalize the game a little more.

Ⓢ How do I convert an AppleWorks file to a standard DOS 3.3 text file? I've always used AW to do all my processing, and I was wondering if there was a way to convert them.

In addition has anybody been able to sector edit 'WINGS OF FURY'? I would like to rig it like 'RESQUE RAIDERS'.

Jim S. Hart

Softkey for...

## The Calculus Toolkit

Addison-Wesley

### Requirements

- A blank disk for each Toolkit disk
- Standard disk copy program
- A DOS 3.3 initialized disk with no hello program
- A fast DOS, such as Diversi-DOS or Pronto DOS (optional)

You can copy any of the disks in the Calculus Toolkit series with a standard disk copier such as COPYA. However, when you try to boot the copied disk, it hangs and tells you that you had a copy protection error. This hangup and message comes right after the BASIC prompt appears so the protection scheme or call to a scheme must be in the HELLO program. Loading the HELLO program and listing it reveals that it BLOADs a file called 'C', does a few POKEs, and then executes a USR function. The rest of the program is gibberish. Hmmmm. The 'C' file must contain the decoding routine and probably also contains the copy protection scheme. Going into the monitor, BLOADing the 'C' file, and then checking \$AA72.AA73 (starting address of a binary file) and \$AA60.AA61 (length of a binary file), I found 00 60 at \$AA72.AA73. I did a 6000L to take a look at the beginning of the code. Here is the disassembly you will find on DISK 1, DISK2, DISK3, and DISK4:

```
6000 A9 FF   LDA #$FF set RUN flag
6002 85 D6   STA $D6
6004 A9 00   LDA #000
6006 8D F2 03 STA $03F2 set up RESET vector
6009 A9 60   LDA #$60
600B 8D F3 03 STA $03F3
600E 20 31 61 JSR $6131 to protection scheme
6011 A5 01   LDA $01 is this an original?
6013 C9 00   CMP #000
6015 D0 04   BNE $001B no, go bomb
6017 20 5F 60 JSR $605F decode routine
601A 60     RTS go back to Applesoft
601B 20 F5 60 JSR $60F5 print message & bomb
...
```

Look around \$6131 if you want to see some genuine signature check code. What I did was to remove the call to the protection and to load memory location \$01 with a 00 if case any more checks were made down the road. The changes to make are:

```
600E EA EA EA NOP
6011 A9 00   LDA #$00
6013 85 01   STA $01
```

I then BSAVED the file back to disk using the values I found and the disk was deprotected!

1 Copy the Calculus Toolkit disks onto the blank disks using the whole disk copier.

2 Boot the blank initialized DOS 3.3 disk. Insert a copied toolkit disk from step 1.

```
CALL -151 enter monitor
BLOAD C
600E.6014 verify 20 31 61 A5 01 C9 00
600E:EA EA EA A9 00 85 01 and modify
BSAVE C,A,$6000,L$1E4
```

3 Repeat step 2 for DISK1, DISK2, DISK3, and DISK4.

4 The fifth disk in the toolkit is called "A Calculus Student's Microcomputer

Toolkit". Change the modify in step 2 to 600E:EA EA EA A9 00 and deprotect this disk too.

You're done!

## Put a deprotected "Where in Europe is Carmen Sandiego" onto a 3 1/2" disk

### Requirements

- Deprotected Where in Europe is Carmen Sandiego disk (2 sides)
- 3 1/2" disk
- ProDOS Copy II Plus (versions 6-up)
- Disk with the BASIC.SYSTEM system file on it

I recently had the chance to play a deprotected version of Where in Europe is Carmen Sandiego over at a friend's house. It continues the Carmen Sandiego tradition of fun with learning. I noticed that the disk was ProDOS based and came on both sides of a 5 1/4" floppy disk. My friend has a IIGs with a 3 1/2" disk drive and her young children use it alot. I thought to myself, "Why not fix the program so that it would run off of a 3 1/2" disk so the kids wouldn't have to worry about whether or not they put the correct side in?" After an hour or so of seeing what was on both sides, I came up with this procedure.

- 1 Boot Copy II Plus.
- 2 Format the 3 1/2" disk with a dummy name of "/DUMMY".
- 3 Copy the ProDOS system file from your Copy II Plus disk to the newly formatted 3 1/2" disk.
- 4 Now copy all of the files except PRODOS from the Where in Europe is Carmen Sandiego disk to the 3 1/2" disk.
- 5 Create a subdirectory on the 3 1/2" disk with the name "SIDE.2".

6 Copy the DATA file from the second side of the Carmen disk into the newly created subdirectory "SIDE.2".

Now we need to change the pathnames within the Carmen program to look for the DATA program in the subdirectory SIDE.2 instead of the disk named /SIDE2. This allows the DATA file to be on the same disk as the main program.

7 Get into BASIC from ProDOS (execute the BASIC.SYSTEM file, in other words).

8 Insert the 3 1/2" disk into a drive.

```
RENAME /DUMMY,/CARMEN.EUROPE
BLOAD B,A$1B00
```

CALL -151  
1D34.1D38  
1D34:73 69 64 65 2E  
BSAVE B

verify 2F 73 69 64 65  
and modify

Note: If the bytes did not match, then use a block editor to search for 2F 73 69 64 65 and change to 73 69 64 65 2E.

Now you can use the 3 1/2" disk to play the game and never have to flip disks over again for it. A side benefit is that the program loads quicker (especially if you have DSR's Diversi-Cache installed).

Softkey for...

## Spiderbot

Epyx

### Requirements

- 1 blank disk
- COPYA
- Sector editor
- COMPUTIST #57 (for reference only)
- Bag of Tricks (if you want to speed up the loading of the game)

Spiderbot, another in the Maxx Out series from Epyx, is an interesting little game in which you, the spiderbot, have to collect parts of yourself and ensnare a variety of baddies using your spider webs. Overall it is an OK game, but it is not up to the standards that it's predecessor, Deathsword, established.

The protection here is essentially the same as other recent Epyx releases. I refer you to my softkey for Boulderdash Construction Set which appeared in COMPUTIST #57 (pp. 17-20) for a more thorough explanation of the protection scheme. It has been slightly altered here, in that zero page locations \$F8-\$FF are used instead of \$F0-\$F7. Also, the values to put into these locations are in a different order and the location to jump to after loading the correct values is slightly different. Aside from those few changes, the rest of the softkey is the same.

**1** Boot your DOS 3.3 system disk.

**2** Tell DOS to ignore checksum and epilog errors and use COPYA to copy the disk.

POKE 47426,24  
RUN COPYA

**3** Make the following sector edits to the copy you just made. Don't forget to write the modified sector back to disk.

Trk	Sct	Byte(s)	From	To
\$00	\$05	\$00-18	?	A9 E7 85 F8 85 FB A9 FC 85 F9 85 FC 85 FF A9 EE 85 FA 85 FD 85 FE 4C 70 BB

**4** Use Bag of Tricks (Quality Software) and re-skew the disk to a 9 descending skew. This really speeds up the loading of the game.

Softkey for...

## Micro Addition/Subtraction

Hayden Software Company

### Requirements

- 1 blank disk
- COPYA
- Sector editor

Micro Addition/Subtraction is an educational program designed for the younger children. It teaches elementary addition and subtraction skills, but the problems are really elementary so they are best suited for younger children who are just learning to add and subtract.

The protection scheme used here is the standard epilog change. The epilog bytes for both address and data fields are AA AA instead of the normal DE AA. This is a simple protection to overcome.

**1** Boot your DOS 3.3 system disk.

**2** Tell DOS to ignore checksum and epilog errors and use COPYA to copy the disk.

POKE 47426,24  
RUN COPYA

**3** Make the following sector edits to the copy you just made.

Trk	Sct	Byte(s)	From	To
\$00	\$03	\$35	AA	DE
		\$91	AA	DE

You're done.

## A self-booting Thexder

### Requirements

- Backup copy of the original THEXDER disk
- Iigs System Disk v3.1 or later
- 3 1/2" disk drive
- ProDOS file copy program
- ProDOS formatted disk with PRODOS 8 and BASIC.SYSTEM on it

The arcade game Thexder is an excellent game for the Iigs owner; top notch graphics, sound, and play action are woven together to produce an unforgettable gaming experience. The one problem, so to speak, of Thexder is that it does not boot on its own. You are required to first boot into a program selector, such as the Launcher or Finder, and then insert your Thexder disk to run the game. A few disk swaps later and your game is ready to go. I

found this lack of the ability to autoboot to be annoying to say the least. I thought there must be some easier way to go about starting up my Thexder game. I found the answer and if you follow the steps below, your Thexder disk can be fixed up so that it too will autoboot.

**1** Boot up your ProDOS 8 disk with BASIC.SYSTEM on it and get into BASIC.

**2** Insert your backup Thexder disk into the 3 1/2" drive.

PREFIX /THEXDER

**3** Now rename the Thexder file so that it will be used as the ProDOS 16 boot file.

RENAME THEXDER,THEXDER.SYS16

**4** Create a subdirectory to hold all of the ProDOS 16 setup files.

CREATE SYSTEM

**5** Go into the SYSTEM subdirectory and create two more subdirectories to hold the Tool setup file and the necessary tools.

PREFIX /THEXDER/SYSTEM

CREATE SYSTEM.SETUP

CREATE TOOLS

**6** Boot your ProDOS file copying program. Copy the files below from the Iigs System Disk (/SYSTEM.DISK) to the Thexder disk (/THEXDER).

PRODOS

SYSTEM/P16

SYSTEM/TOOLS/TOOL025

SYSTEM/SYSTEM.SETUP/TOOL.SETUP

After you have finished, your Thexder disk should be arranged as follows:

```

/THEXDER
  prodos
  thexder.sys16
  /DATA (none of these files were altered)
  /SYSTEM
    p16
  /SYSTEM.SETUP
    tool.setup
  /TOOLS
    tool025
    
```

or, if you have Copy II Plus, a catalog will reveal:

```

/THEXDER---->DATA
!
!---->SYSTEM----->SYSTEM.SETUP
!
!---->TOOLS
    
```

Your Thexder disk can now boot by itself and the game will load in automatically. As an extra bonus, pressing the 'J' key will toggle Joystick mode. No longer do you have to always use the numeric keypad. I have found that the joystick tends to center to the lower right hand corner, so you may have to re-adjust your trim pots.

## Capturing a protected disk's RWTS with RWTS Worm

Looking through the various softkeys from issues past, you notice that many of them require a copy protected disk's RWTS. This RWTS is used to read in the copy protected disk and then write out the information just read in onto a normally formatted 16 sector disk. This is an easy way to deprotect a disk, but obtaining the necessary RWTS is often difficult. I present here a way to get that elusive RWTS. The program that accomplishes the feat is called RWTS Worm and is written in machine language. This program works for all protected disks that employ a standard DOS 3.3 first stage boot, i.e. track \$00 sector \$00 is the same as a normal DOS 3.3 disk, and run off of a semi-normal DOS 3.3. With most semi-normal DOSs, a BASIC prompt (>) appears after the first three tracks have been loaded in.

### What It Does

When run, RWTS Worm first disconnects DOS via a JSR \$9EE0, which replaces DOS' I/O hooks with the normal BASIC I/O hooks. The DOS 3.3 in memory is moved down so the upcoming boot does not overwrite it. Next, the disk controller ROM that lives at \$C600-\$C6FF is moved down into RAM memory and modified to the extent that only the disk's RWTS is loaded in. Once this has been done, the newly read in RWTS is moved to \$1900-\$20FF. Finally, the DOS 3.3 that was in memory when you loaded the RWTS Worm program is moved back to its normal resting spot and it is reactivated via a JMP \$9DBF (DOS 3.3 warmstart location).

RWTS Worm will probably not work on a disk that has a non standard DOS 3.3 track \$00, sector \$00 or whose BOOT1 does not load in at \$B700, but you can try it anyways. A word of caution: the program assumes your disk controller card is in slot 6. If it is in any other slot, RWTS Worm will not work.

**1** Boot a DOS 3.3 disk and go into the monitor. Type in the machine language hexdump and save it to disk.

**CALL -151**

### RWTS WORM

9500:	20 E0 9E A2 00 BD 00 9D	\$05E4
9508:	9D 00 6D E8 D0 F7 EE 07	\$1E10
9510:	95 EE 0A 95 AD 07 95 C9	\$12AF
9518:	C0 D0 E8 A2 00 BD 00 C6	\$D75D
9520:	9D 00 96 E8 D0 F7 A9 33	\$BA6B
9528:	8D F9 96 A9 95 8D FA 96	\$8BD6
9530:	4C 00 96 A9 90 8D 0E 08	\$F51A
9538:	A9 4C 8D 4A 08 A9 4C 8D	\$DDB9

9540:	4B 08 A9 95 8D 4C 08 A2	\$3F59
9548:	60 4C 01 08 2C E8 C0 A2	\$D90B
9550:	00 BD 00 B8 9D 00 19 E8	\$A813
9558:	D0 F7 EE 53 95 EE 56 95	\$5133
9560:	AD 53 95 C9 C0 D0 E8 A2	\$D5C5
9568:	00 BD 00 6D 9D 00 9D E8	\$AF53
9570:	D0 F7 EE 6B 95 EE 6E 95	\$3267
9578:	AD 6E 95 C9 C0 D0 E8 4C	\$F97A
9580:	BF 9D	\$D5F9

**BSAVE RWTS WORM, A\$9500, L\$82**

### SAMPLE RUN

**1** Boot a DOS 3.3 formatted disk and load RWTS WORM.

**BLOAD RWTS WORM, A\$9500**

**2** Take out the DOS 3.3 disk and insert your protected disk into slot 6, drive 1.

**3** Execute the RWTS WORM program.  
**CALL 30144** *from monitor 9500G*

**4** The disk drive will sound like it is booting and after a few seconds the BASIC prompt (>) will return. The protected disk's RWTS has been relocated to \$1900-20FF.

**5** All you have to do now is to save the RWTS to disk. Insert your Super IOB disk.

**BSAVE programname.RWTS, A\$1900, L\$800**

You're done. You can now use Super IOB and the SWAP or NEW SWAP controller to deprotect the disk. If you want to use RWTS Worm on another disk, you will first have to reload it into memory because the code it contains is self-modifying.

*Softkey for...*

**Math Shop**  
Scholastic Inc.

#### ■ Requirements

- 2 blank sides
- Whole disk copy program, such as COPYA
- ProDOS file utilities or ability to get into ProDOS BASIC

Math Shop, by Scholastic, is a math oriented educational program. In the Math Shop game, you are required to go to one of several different stores and "serve" the customers by figuring out math problems. After you have satisfied all of the customers in one store, you must go to another store and continue on "serving" them. This becomes quite a task when you have 10 stores each with 6-7 customers! If you ignore the customers in a store for too long, they will leave thus depriving you of their "business".

Copying both sides of the disk with COPYA results in no problems. Upon booting

the front side, the ProDOS title screen comes up and then almost immediately the program reboots. Since the rebooting occurs right after the ProDOS title screen, a good place to look for the copy protection code is in the first ".SYSTEM" file in the directory.

The first ".SYSTEM" file is called MATH.SYSTEM and the file right after it is called BOOT.SYSTEM. The MATH.SYSTEM file is only one block long, hardly enough to do anything except maybe check for an original disk?!?!? That's exactly what it does. After verifying the presence of the original disk, the file BOOT.SYSTEM is then executed.

Well, it seems apparent what our course of action is: alter the disk so that MATH.SYSTEM is not executed and instead BOOT.SYSTEM is executed. This is easily accomplished. After ProDOS is loaded, it looks in the volume directory for the first ".SYSTEM" file and executes it. If we rename MATH.SYSTEM to MATH, then it is never executed! Since BOOT.SYSTEM is now the first ".SYSTEM" file, it is executed which is exactly what we want.

**1** Copy both sides of the MATH SHOP original.

**2** Go into BASIC or use a ProDOS disk utility program and rename MATH.SYSTEM to MATH.

**RENAME MATH.SYSTEM, MATH**

*Softkey for...*

**ARCHIVEmath**

D.C. Heath & Company

#### ■ Requirements

- 2 blank disk sides
- FID
- Super IOB v1.5

ARCHIVEmath is a rather good college algebra test generating program. You may choose from several areas in math and print up to 99 different tests in each area. Examining the disk's format with my trusty nibble editor, the only protection is that the third byte of the data prolog (05 AA AD) is an AA instead of the normal AD. Aside from that, the only inconvenience is that you need to use the original DOS on the disk because it contains routines that the programs need to run.

**1** Load Super IOB and install the FAST controller into it.

**2** Fix the controller so it only copies DOS (tracks \$00-02).

**1015 LT = 3**

**3** Run Super IOB and answer "YES" to the format disk prompt.

**4** Repeat step #3 for the back side of the ARCHIVEmath disk.

**5** Now, from the BASIC prompt (]), load FID.

**BLOAD FID**

**6** Enter the monitor and alter DOS so that it can read ARCHIVEmath's altered format.

**CALL -151**  
**B8FB:29 00** ignore last byte of data prolog  
**B942:18** ignore any checksum/epilog errors

**7** Run FID. When asked for the filename, enter "=" and answer "NO" to the "DO YOU WANT PROMPTING" message:

**B03G**

**8** You have just copied all of the files from the first side of ARCHIVEmath to the first blank disk. Now, repeat this step for the other sides.

You're done. No changes need to be made to the ARCHIVEmath DOS, so put your original away and use your backup.

Captain Dan

*Softkey for...*

**Dondra**

Spectrum Holobyte

■ Requirements

- Apple II
- Super IOB v1.5
- Sector editor
- Three blank disks

Dondra is a text/adventure game. The game was easily copied but upon booting the program would die. At this point I was certain that a nibble count was being used.

Scanning the disk for the byte sequence 8C 00 (\$C08C, it's backwards remember) I found some interesting code on track \$00, sector \$05. It seemed to be looking for a bunch of \$E7's. It was looking for a signature. It didn't matter what I changed the disk would still not boot. I then decided that since I could not bypass the nibble count, I would give it what it wanted.

See COMPUTIST #60, page 20, "Some notes on Epyx" by Stephen Lau for an explanation of Prolog signature protection.....RDEXed

I turned on my computer, hit **RESET**, entered the monitor, and cleared memory:

**RESET**  
**CALL -151**  
**B00:00**

**B01<800.BFFFM**

I then inserted the Dondra original and moved the boot ROM code down to RAM and changed it so it would stop after the first stage load.

**9600<C800.C6FFM** move boot down  
**96F8:4C 59 FF** make it exit to monitor  
**9600G** run it to load boot  
**C0E8** turn of drive  
**B01L** list what we got

I then learned that at address \$84A the program JMPed to \$BB00 for the second stage load. OK let's find out what's there.

**96F8:A9 59 8D 4B 08 A9 FF 8D** modify boot1  
**9700:4C 08 4C 01 08** to exit to monitor  
**9600G**  
**C0E8**  
**B800L** list boot2

Suprise, the nibble count is at the beginning of the second stage load. I noticed that after the program was finished finding the \$E7's it loaded more data from the same track (\$0A) and stored it at \$F0-F7. I then decided that I had to have that data. The data was not loaded if the nibble count failed and the program needed the data to continue. I had to alter the second stage load so it would stop after loading the data.

**96F8:A9 00 8D 4B 08 A9 09 8D**  
**9700:4C 08 4C 01 08**  
**0900:A9 00 8D 7C BB 4C 00 BB** insert a break  
**9600G**  
**C0E8**  
**F0.F7** see what we got

I wrote this information down and then made a copy of the original using the controller below, which ignores address epilog bytes. Since I was going to give the program the information it wanted, it didn't matter what the nibble count loaded. I changed it to ignore whatever information it found.

**1** Merge the controller into Super IOB and copy the original disk.

**2** Use your favorite sector editor to make some changes.

Trk	Sct	Byte(s)	From	To
\$00	\$05	\$45-46	D0 ?	EA EA
		\$4E-4F	D0 ?	EA EA
		\$57-58	D0 ?	EA EA
		\$66-7B	?	A0 07 B9 74 BB 99 F0 00 88 10 F7 4C 7C BB FC EE EE FC E7 EE FC E7

This is the disassembled form of the changes:

**BB66 LDY #507**  
**BB68 LDA \$BB74,Y**  
**BB6B STA \$00F0,Y**

**BB6E DEY**  
**BB6F BPL BB68**  
**BB71 JMP \$BB7C**  
**BB74 FC EE EE FC E7 EE FC E7**

**3** Copy the remaining sides with Super IOB's standard controller or COPYA.

**Controller**

**1000 REM DONDRA CONTROLLER**  
**1010 TK=0:ST=0:LT=35:CD=WR:POKE 47507,0:**  
**POKE 47517,0**  
**1020 T1=TK:GOSUB 490**  
**1030 GOSUB 430:GOSUB 100:ST=ST+1:IF ST<DOS**  
**THEN 1030**  
**1040 IF BF THEN 1060**  
**1050 ST=0:TK=TK+1:IF TK<LT THEN 1030**  
**1060 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:IF BF=0 AND TK<LT THEN**  
**1070**  
**1090 IF TK<LT THEN 1020**  
**1100 POKE 47507,174:POKE 47517,164:HOME:**  
**PRINT "COPY DONE":END**

**Checksums**

1000	- \$356B	1060	- \$0A25
1010	- \$9F9C	1070	- \$0220
1020	- \$08C0	1080	- \$8942
1030	- \$1EC1	1090	- \$37BB
1040	- \$5E00	1100	- \$DE1C
1050	- \$769B		

*Softkey for...*

**King's Quest III**

Origin

■ Requirements

- Apple II with 128K
- COPYA
- Sector editor
- Five blank disks

King's Quest III is like its predecessors except that it is bigger and we no longer have to watch the screens being drawn, but alas, it is still extremely slow. The protection is the same as King's Quest I & II just in a new place.

**1** Copy all sides with COPYA.

**2** Sector editor the boot disk only.

Trk	Sct	Byte(s)	From	To
\$02	\$08	\$12-14	20 00 FF	EA EA EA
\$0B	\$0C	\$FC-FE	20 00 FF	EA EA EA

Softkey for...

## Ultima V Origin

### Requirements

- Super IOB v1.5
- Four blank disks

Please note: For this softkey to be completed properly, you will need COMPUTIST #28, #42, and #51. No I'm just kidding. Worried you didn't it.

The protection scheme for Ultima V is a modified PRODOS that changes the sector values on the disk, and alters address epilg bytes. Each sector beyond track \$02 is increased by \$11.

For example, sector \$06 would be encoded as sector \$17 (\$06 + \$11). Since there can only be 16 (\$0F) sectors on a track, any value over \$0F is unreadable by an unmodified program.

I wrote the controller below, which ignores address epilg bytes and converts the sectors to normal values, and copied the disk. Upon booting the disk, the protected PRODOS operated fine with the normal sectors and epilg bytes. The program did not bother to check the changed data.

**1** Type in the machine language program and save it.

**CALL -151**

1900: A2 00 BD B8 BF 18 69 11	\$E96C
1908: 9D B8 BF E8 E0 10 D0 F2	\$784F
1910: 60 A2 00 BD B8 BF 38 E9	\$353D
1918: 11 9D B8 BF E8 E0 10 D0	\$B7FB
1920: F2 60	\$E0D7

**BSAVE ULTIMA 5.SC, A\$1900, L\$022**

**2** Merge the controller into Super IOB and copy the original disk.

**3** Copy the remaining sides with COPYA or Super IOB's standard controller.

### Controller

```

1000 REM ULTIMA V CONTROLLER
1010 TK = 0:ST = 0:LT = 35:CD = WR
1020 POKE 47507,0: POKE 47517,0
1030 UB = 0:T1 = TK: GOSUB 490
1040 IF UB = 1 THEN 1060
1050 IF TK = > 3 THEN CALL 6400:UB = 1
1060 GOSUB 430: GOSUB 100:ST = ST + 1: IF ST < DOS
    THEN 1060
1070 IF BF THEN 1090
1080 ST = 0:TK = TK + 1: IF TK < LT THEN 1040
1090 UB = 0: GOSUB 490:TK = T1:ST = 0
1100 IF UB = 1 THEN 1120
1110 CALL 6417:UB = 1
1120 GOSUB 430: GOSUB 100:ST = ST + 1: IF ST < DOS
    THEN 1120
    
```

```

1130 ST = 0:TK = TK + 1: IF BF = 0 AND TK < LT THEN
    1100
1140 IF TK < LT THEN 1030
1150 POKE 47507,174: POKE 47517,164: HOME :
    PRINT "COPY DONE" : END
10010 PRINT : PRINT : PRINT CHR$(4); "BLOOD"
    ULTIMA*5.SC,A$1900"
    
```

### Checksums

1000 - \$356B	1090 - \$D00D
1010 - \$3266	1100 - \$68D6
1020 - \$5917	1110 - \$6483
1030 - \$51D2	1120 - \$4681
1040 - \$8CFB	1130 - \$2068
1050 - \$DE3C	1140 - \$D390
1060 - \$D138	1150 - \$33C7
1070 - \$0586	10010 - \$129A
1080 - \$752E	

## MOST WANTED Softkeys

ABM Muse  
 Agent U.S.A. Scholastic  
 Airheart Broderbund  
 Algeblaster Davidson & Associates  
 Algebra I Intelligent Tutor  
 Alien Mind PBI  
 Anhk Datamost  
 Apple Super Pilot ?  
 Artificial Intelligence Scholastic  
 Balance of Power Mindscape  
 Bandits Sirius Software  
 Bank Street Filer Broderbund  
 Bank Street School Filer Sunburst Communications  
 Barron's Computer SAT ?  
 Battlegroup SSI  
 Battlezone Atarisoft  
 Brain Bank The Observatory  
 Burgertime ?  
 Captain Goodnight ?  
 Caverns of Callisto Origin  
 Certificate Library Vol. 1 ?  
 Certificate Maker Iigs Springboard  
 Chuck Yeager's Advanced Flight Trainer Electronic Arts  
 Colossus IV Firebird  
 Creature Venture Softsmith Co.  
 Cross Country Rally Softsmith Co.  
 David's Midnight Magic ?  
 DB Master V4.0 Stoneware  
 Deathlord Electronic Arts  
 Epidemic S.S.I.  
 F-15 Strike Eagle Microprose  
 Fay: The Masked Woman Didatech Software  
 Fay's Word Rally Didatech Software  
 Fortress S.S.I.  
 Fun Bunch Unicorn  
 Galaxian Atarisoft  
 Galactic Gladiators S.S.I.  
 Game Show Advanced Ideas  
 Gemstone Healer SSI  
 GradeBusters 1-2-3 Gradebusters  
 Gutenberg Jr. Micromation LTD  
 Handicapping System Sports Judge  
 Impossible Mission Epyx  
 Jigsaw Microfun  
 Joust Atarisoft  
 Killed until Dead Accolade  
 Legacy of the Ancients Electronic Arts  
 Little Computer Peoples House on a Disk Activision  
 Lollipop Dragon Society for Visual Ed  
 Cursor Control Adventures  
 Plotting & Programming Adventures  
 Letter & Number Key Adventures  
 Function Key Adventures  
 Magic Spells The Learning Company  
 Maxi Golf Thunder Mountain  
 Micro League Baseball Micro-league Sports  
 Mr. Do Datasoft  
 Mr. Pixel's Cartoon Kit Mindscape  
 Ms. Pac-Man Atarisoft  
 Never Ending Story Datasoft  
 Odin Odesta  
 On Balance Broderbund  
 Peeping Tom Microlab  
 Pensate Penguin  
 Personal Finance Manager (PFM) Apple Computer  
 Prime Plotter Primesoft Corp.  
 Principals Assistant Library Mindscape  
 Print Master Unision World  
 Pro-Football Sports Judge  
 Publisher Springboard  
 Puzzles & Posters MECC  
 Quiz Castle Didatech Software  
 Ruski Duck Softsmith Co.  
 Scrabble Electronic Arts  
 S.E.U.I.S. S.S.I.  
 Snoggle Broderbund  
 Space Eggs Sirius  
 Space Journey Mindscape  
 Spare Change Broderbund  
 Stellar 7 Penguin Software  
 Stellar 7 Software Entertainment  
 Success with Typing v1.2 Scholastic Software  
 Super Factory Sunburst  
 Tomahawk ?  
 Think Tank Living Video  
 Toy Shop Broderbund  
 Universe Omnitrend  
 Universe II Omnitrend  
 Visiblend Microlab  
 Wargle Hayden  
 Wheel of Fortune ?  
 Where in the USA is Carmen San Diego Broderbund  
 Where in the World is Carmen San Diego Broderbund

# COMPUTIST back issues

To order, Use the back issue order form on page 42

**60** 1988 • **Features:** • Reading from Protected Disks • EDASM Transfer Utility • Amper QuickDraw II Routines • Slow Speeds for Arcade Games • A Patch for Copy II Plus • The Product Monitor • Low Cost Alternate Languages • Another method for Copy-Protecting DOS 3.3 • Change Appleworks Cursor from a blinking underline to a blinking apple in one quick patch • **Softkeys:** • Ace Programmer • Aesop's Fables (GS) • Apple GEOS v2.0 • Artic Antics • Creating the Constitution • Crossword Magic 4.0 • Death Sword • Dome Bookkeeping • Essential Data Duplicator III (EDD 3) • Garfield Deluxe Edition • Gauntlet Igs • J & S Gradebook • Kings Quest Igs • Kings Quest I (GS) • Lazer Maze • Leisure Suit Larry • Magical Myths • Mastertype v2.1 • Mavis Beacon Teaches Typing • Mind Prober/Mind over Minors • NATO Commander v1.2 • Paperboy • Pegasus • Pinball Construction Set • Pool 1.5 • RAD Warrior • Railroad Works • Road Rally USA • Robomath • Science Tool Kit • Search and Rescue • Showoff • Social Studies Vol 1 • Social Studies Vol 2 • Solo Flight • Space Quest • Springboard Publisher • Spy vs Spy III • Star Fleet I v2.1 • Tales of Fantasy: The Dark Tower, Frog & The Fables • Tapper • Troll Courseware: Maps & Globes, Latitude & Longitude • Type! • Ultima V • Wagons West • Zorro • **APTs:** • Kid Niki Radical Ninja • Lode Runner • Space Quest II • Thexder • **Playing Tips:** • 2400 A.D. • Captain Goodnight • Space Quest I • **Notes:** • Defending the Crown with "Visit Monitor" • Formatting extra tracks • Merging controllers with Super IOB • More on Tower of Myrarglen • Some notes on Electronic Arts • Some notes on Epyx • Tips for cracking Igs 3 1/2" disks **IBM** • **Softkeys:** • Clipper • DoubleDOS v1.0 • LoadCalc v4.13 • Mind Prober

**59** 1988 • **Features:** • Boot IIC Utilities on any compatible • Help with Ultima V • Convert Printmaster graphics into Print Shop graphics • The Invincible Bard (III) • The Deprotection Game • Patching ProDOS 1.1.1 • The Product Monitor • AppleWorks and the ImageWriter II • **Softkeys:** • APBA Major League Players Baseball • Better Working: Spreadsheet • Better Working: Word Processor • Beyond Pinball • BoulderDash Construction Kit • Calendar • Calendar Crafter • California Games • Championship Baseball • Clip Art Collection vol 1 • Cours Et Gagne v1.0 • Dazzle Draw • Everybody's Planner • GATO v1.3 • GoldFinger • Jumpman • King's Quest • Management Edge • Movie Monster Game • MultiScribe v3.0 • Negotiation Edge • Nightmare Gallery • Patchworks • Picture Phonics • Printographer • Reader's Treasure Chest • Real Math • Sea Strike • Shifty Sam • Simulation Construction Kit • Snoopy's Reading Machine • Snoopy's Skywriter Scrambler • Snoopy to the Rescue • SongWriter • Spell It! • Stickybear Reading Comprehension • Super BoulderDash • **APTs:** • Bard's Tale III • Super BoulderDash • Ultima V • **Playing Tips:** • Alternate Reality • Auto Duel • Ultima V • Where in the World is Carmen Sandiego **IBM** **Softkey:** • Chartmaster v6.04 • Clout v1.0 • Copywrit • Graphwriter • Signmaster v5.04

**58** 1988 • **Features:** • Mixing ProDOS with Thexder • Infocom Decoder Revisited • Cracking on the IIC • Might & Magic Revisited • Might & Magic Character Editor • An in-depth guide to Ultima IV • Computing for 1-3 year olds • The Product Monitor • **Softkeys:** • A.I. • BoulderDash • BoulderDash Construction Set • DeathSword • Design Your Own Home: Architecture • Design Your Own Home: Interior • Design Your Own Home: Landscape • Facemaker • Gauntlet • Kings Quest II • Mastery Arithmetic • Microzine #26 • Muppet Slate v1.0 • PFS: Graph • Polywriter 1.2 • Rad Warrior • Rings of Zilfin • SeasPELLer • Smart Eyes • Spell It! • Wings of Fury • **APTs:** • Castle Wolfenstein • Marble Madness Igs • Might & Magic • Ultima IV • Ultima V • **Playing Tips:** • Coveted Mirror • Deathlord • Might & Magic • Oo-Topos • Ultima IV • **Notes:** • A better way to print Starter

Kit DOC's • Curing Fatal System Error #0911 • Realtime Situation Control using CDAs • Chuck Yeager's Advanced Flight Trainer • **IBM Notes** • Help Wanted!

**57** 1988 • **Features:** • Electronic Art's Protection Language • How to find hidden code with EOR DiskScan • Here's another look at • Lower Case & Infocom Games • Monsters of Might & Magic • A Character Editor for Rings of Zilfin • A Single Data Disk for all your Print Shop Graphics • The Product Monitor • **Softkeys:** • American Challenge • Arctic Fox • Bard's Tale II • BoulderDash Construction Set • California Games • Championship Wrestling • Chief of Detectives/Drawing Conclusions • Deep Space • Dome Simplified Bookkeeping System • Dr. Ruth's Computer Game of Good Sex • Earth Orbit Stations • Factory • Galaxy Search/Predicting Outcomes • Game Maker • Hacker II • Hardball Igs • Ikari Warriors • Labyrinth • Marble Madness • Master Diagnostics IIC • Math Blaster • Mickey's Space Adventure • Micro-Computer Learning Games • Microzine #14 • Microzine #24 • Milliken Math Series • Mind Prober • M - ss - ng L - nks • Morning Star Spelling • Mountain Climbing/Cause and Effect • Movie Monster Game • Pond • Race Track/Reading for Detail • Reading Comprehension Main Idea & Details • Rings of Zilfin • Roadwar 2000 • School Days/Inference • Ski Craze • Softswitch • Sub Mission • Time Capsule/Reading Skills • Tuesday Morning Quarterback • Typewriter • Where in Europe is Carmen Sandiego • Wortgefecht • Xevious • **APTs:** • Deep Space • H.E.R.O. • Moebius • Rings of Zilfin • Roadwar 2000 • **Playing Tips:** • Arcticfox • Castle Wolfenstein • Conan • Donkey Kong • Ultima IV **IBM Softkeys:** Symphony v1.00 • TK!

**56** 1988 • **Features:** • Apple Igs Secret Weapon • 5 second fastboot into Locksmith 6.0 Fastcopy • The Product Monitor • Taking the grind out of Championship Wrestling • Making some improvements to The Nibbler • **Softkeys:** • 2400 AD • 40" Graphics Studio • Accolade Comics • Aesop's Fables • American People • Animal Hotel • Appewriter IIC • Arcade Album #1 • Arctic Antics • Ballblazer • Bard's Tale II: The Destiny Knight • Bard's Tale Igs • Cat'n Mouse • Championship Wrestling • Charlie Brown's 1,2,3's • Cobra Cavern • Color Me • Create With Garfield • David Winfield's Batter Up! • Destroyer • Disk Optimizer II • Dragonworld • Electronic Arts Software • En Vacances • En Ville • Fantavision gs • Fight Night • Forbidden Castle • G.I. Joe • Garfield Double Dares • General Manager • Goonies • GraphicWriter 2.0 • Gutenberg, Sr. • Hacker II • Hardball • Hardball gs • Infiltrator • James Bond 007: A View to a Kill • Keyboard Kadet • Kids on Keys • Lazer Maze • Le Demenagement • Le Francais par Ordinateur • Leisure Suit Larry in the Land of the Lounge Lizards • Les Sports • Lion's Workshop • Microzine #21 • Microzine #22 • Milliken Word Processor • Millionair II • Multiscribe v2.0 • Multiscribe v3.0 • Multiscribe Igs v3.01c • Paris En Metro • Pitfall II • Racter • Railroad Works • Rambo: First Blood Part II • Realm of Impossibility • Same or Different • Sea Dragon • Sea Strike • Shanghai • ShowOff • Shutterbug • Silent Service Igs • Snoopy to the Rescue • Snoopy's Reading Machine • Snoopy's Skywriter Scrambler • Space Quest I • Space Station • Spy Hunter • Spy's Adventures in Europe • Spy's Adventures in North America • Stephen King: The Mist • Story Maker • Street Sports Basketball • Sub Battle Simulator • Super Sunday Football • Talking Text Writer • Type! • Un Repas Francais • Voodoo Island • Where in Europe is Carmen Sandiego • Winnie the Pooh • Winter Games • Winter Games gs • Wordzsearch • World Games gs • **APTs:** • 2400 AD • Kung-Fu Master • Lady Tut • Ultima V • **Playing Tips:** • 2400 AD • Deathlord • Space Quest • Ultima IV • Ultima V • Wrath of Denethor **IBM Softkeys:** Execu-Vision • MS Word • PC-Draw • Zork I & II • **Notes:** • Not too happy with Copy II Plus v8.1 • Data disks cause Appleworks crash

**55** May 1988 • **Features:** • A Utility to Save the Lower 8 Pages of Memory • Bard's Tale Effects Locator • How to Capture Phantasie Screen Maps • Alternate Reality Character Editor • Updating the ProDOS Block Editor • Loading Flashcalc onto your RAMcard • A Copy-protection Scheme for ProDOS • The Product Monitor • Autoduel Car Editor • **Softkeys:** • Alphabet Sequencing • Animal Alphabets and Other Things • Arctic Antics • The Boars' Store • The Boars Tell Time • Career Focus • Castle Wolfenstein • Charlie Brown's 123's • Charlie Brown's ABC's • City Country Opposites • Coveted Mirror • Create With Garfield • Crypt of Medea • Customized Alphabet Drill • Customized Flash Spelling • Dig Dug • Digital Paintbrush System • Estimation • Fay: Word Hunter • Fix It • Focusing on Language Arts • Fundamental Capitalization • Fundamental Punctuation Practice • Fundamental Spelling Words in Context • The Hobbit • Homonyms in Context • Individualized Study Master • Inside Outside Shapes • Inside Outside Opposites • Leisure Suit Larry in the Land of the Lounge Lizards • Master Diagnostics II & II+ • Mastertype v2.1 • Mathematics Series • Mr. and Mrs. Potato Head • Paper Models - The Christmas Kit • Peanuts Math Matcher • Peanuts Maze Marathon • Peanuts Picture Puzzlers • Perry Mason: The Case of the Mandarin Murder • Railroad Works • Random House Library • Management Programs • Rocky's Boots v.4 • Sensible Speller • Snoopy's Reading Machine • Snoopy's Skywriter Scrambler • Snoopy to the Rescue • Snoopy Writer • Spelling Demons • Stock Market Simulation • Story Builder • Story Starter • Studio II • Test Maker • Think Quick v1.0 • Tournament Bridge • Tutorial Comprehension • Typing is a Ball, Charlie Brown • Under Fire • Word Blaster • Word Count • Word Mount • Your Personal Net Worth • **APTs:** • Under Fire **IBM Feature:** • Flight Simulator RGB Modifications

**54** April 1988 • **Features:** • Picture Loader • How To Make DEMUFFIN PLUS • Convert Print Shop graphics into Print Master graphics • lower case letters For Your Apple II Plus • The Product Monitor • Apple IIC Paddle Fix • Softkey for Daisy Professional "Most-Protected" Award • DOS EOR Maker • **Softkeys:** • A2-PB1 Pinball • Animate • Bank Street Music Writer • Boulderdash Construction Kit • California Games • Countdown to Shutdown • Coveted Mirror • Create with Garfield • Daisy Professional • Destroyer • Donkey Kong • Expedition Amazon • General Chemistry Disk #8 • Graphics Studio • Green Globes & Graphic Equations • Kalamazoo Teacher's Record Book 2.0 • Kids on Keys • Marble Madness • Math Blaster • Maxwell Manor • Peanuts' Maze Marathon • Petro-Calc • Poet Artist • Practical Grammar • Rendezvous • Ring Quest • Roadwar Europa • Roadwar 2000 • Rocky Horror Show • Sesame Street Electric Coloring Book Series • Sesame Street Letters for You • Sesame Street Numbers • Seven Cities of Gold • Snoopy's Reading Machine • Spy's Adventures in Europe • Spy's Demise • Super Sunday Football • Talisman • Tellstar II • Top Draw v1.01A • The American Challenge • The Dam Busters • The Science Professor • Tubeway • Vocabulary Adventure I • Winter Games • Wizards' Crown • Zero-Gravity Pinball • **APTs:** • Expedition Amazon • Might and Magic • **Playing Tips:** • Beauracracy • King's Quest II • Lurking Horror • Maniac Mansion • Stationfall **IBM Feature:** • Introduction to IBM Disk Format, Access, and Copy-protection • Putting Sargon III on harddisk **IBM Softkeys:** • Prokey 3.0 • R:base 400 • Time Manager

**53** March 1988 • **Features:** • Modify Super IOB to read/write every other track • APT for Rings Of Zilfin: Turn yourself into a lean, mean fighting Machine • More Softkeys for M.E.C.C. software (1987) • How To Use The Electronic Art's RWTS • APT for Realms Of Darkness: Realm's Wrecker!

# COMPUTIST back issues

To order, Use the back issue order form on page 42

•Putting Super Boulder Dash onto a hard disk • **Softkeys:** •2400 A.D. •Age Of Adventure •Apple's Core II •Arcade Boot Camp •Arctic Fox •Aztec •Ballblazer •Bard's Tale Igs •Blue Powder Gray Smoke •California Games •Championship Wrestling •Colonial Conquest •Comprehension Skills I,II •Conquering Whole Numbers •Coordinate Math •Countdown To Shutdown •Dataquest: The World Community •Destroyer •Dream House •Dream Zone •Earth Orbit Station •Equation Math •Forecast: Your At-Home Weather Station •Fraction Concepts Inc •Fraction Munchers •Fraction Practice Unlimited •GBA Championship Basketball •Genesis •GFL Football •Ghost Rider •Goonies •Grade Manager v2.3 •Great American Cross-country Road Race •Hardball Igs •Ikari Warrior •Jenny's Journeys •Kid Niki Radical Ninja •Kung-Fu Master •Learning To Tell Time •Leisure Suit Larry •Let's Learn About Money •Let's Learn About The Library •Letters For You •Lords Of Conquest •Magic Spells •Math Blaster •Money Works •Maps & Globes: Latitude & Longitude •Marble Madness •Microzine 18,19,20,21,22,23 •Mist •Morning Star Math •Movie Monster Game •Multiplication Puzzles •Multiscribe v3.0c •Murder On The Mississippi •Music Made Easy •Mystery Sentences •Number Munchers •Numbers Count •Odell Lake •Operation Frog •Opposites Attract •Oregon Trail v1.4 •Phonics Prime Time: Blends & Digraphs •Phonics Prime Time: Vowels I, II •Puzzles & Posters •Quotient Quest •Reader Rabbit •Reading Style Inventory •Realm Of Impossibility •Sesame Street 'Crayon' series •Shanghai •Sons Of Liberty •Space Quest v2.2 •Story Book: Pixelworks •Story Tree •Subtraction Puzzles •Super Huey •Super Wordfind •Tass Times In Tonetown •Those Amazing Reading Machines III, IV •Timothy Leary's Mind Mirror •To Preserve, Protect and Defend •Tower Of Myrarglen •Troll's 'MicroCoarseware' series •Webster: The Word Game •Word Munchers •Words At Work: Compound It •Words At Work: Suffix Sense •World Games •World's Greatest Baseball Game •World Karate Championship •Writer Rabbit •Zoyon Patrol • **APTs:** •Buck Rogers •Ikari Warrior •Kung-Fu Master •Leisure Suit Larry Igs •Marble Madness •Realm Of Darkness •Rings Of Zilfin •Space Quest Igs •Super Boulder Dash • **Playing Tips:** •2400 A.D. •Donkey Kong •Infiltrator •Space Quest Igs •Spy Hunter •Swashbuckler •Thexder •Ultima II • **IBM Softkeys:** •EasyWriter 1.0, II •Zork III

**52** February 1988 • **Features:** •The Product Monitor •Unprotecting The Unprotectable: Macintosh Softkeys! •A.P.T. Cornucopia •APT: Alternate Reality-Dungeon: Create A Super-human •Softkey for SSI's RDOS disks: 1.ProDOS RDOS, 2.RDOS Transfer Utility •Making Cracked II Plus Disks Work On The IIC • **Softkeys:** •Apple Gradebook v2.6 •Award Maker Plus •Black Cauldron •Black Magic •California Games •Car Builder •Color Print Shop •Computer Ambush •Concepts In Science •Disney's Comic Strip Maker •Elite •Empire I, II •European Nations & Locations •Fooblitsky •Grid Designer •H.E.R.O. •Ikari Warriors •Infiltrator II •Le Francais par Ordinateur •Little Computer People's House on a Disk •Main Street Filer •Master Diagnostics Iie •MegaFiler •MegaMerge •Microzine 23 •Might & Magic •Millionaire •Mindplay software •Music Construction Set •Nibbler •Operation Market Garden •Phantasia •Planetfall •PrintMaster Plus •Print Shop •Questron •Regatta •Ring Quest •Ringside Seat •Rings Of Zilfin •Shanghai •Silent Service •Snooper Troops •Spy's Adventure in N. America •Super Print •Tass Times In Tonetown •Think Quick •Transylvania •Ultima I re-release •Where in the USA is Carmen Sandiego •World Games •Zork I (★ *Macintosh softkey*) • **A.P.T.s:** Alternate Reality: The Dungeon •Arctic Fox •Bard's Tale II •Beyond Zork •Black Magic •Cavern Creatures •Drol •Goonies •Ikari Warriors •Zorro • **Playing Tips:** •Beyond Castle Wolfenstein •Championship Lode Runner •Conan •King's Queen II •Lode Runner •Lurking Horror •Station Fall •Ultima IV •Zork **IBM Softkeys:** •Lotus 1-2-3

•Flight Simulator •PFS Report IBM APTs: •Bard's Tale

**51** January 1988 • **Features:** •The Crypt-arithmetic Helper •Using EDD IV to Modify Tracks And Sectors •Bard's Tale APT: Dungeon Mapper Revisited •RAMfactor mod for Laser 128 •Ultima IV APT edit-tables •The Product Monitor •Get Better Sound by using the cassette jacks •Making A Fast Boot Disk •Might & Magic APT edit-tables • **Softkeys:** •2400 AD •Aliens •Alphabet Zoo •Amnesia •Bag Of Tricks •Bard's Tale I •Bard's Tale II •Battle Cruiser •Beach-head II •Below The Root •Black Magic •Body Awareness •Bridge 4.0 •Carriers At War •Catalyst 3.0 •Centipede •Championship Boxing •Championship Wrestling •Chessmaster 2000 •Combining The Elements •Commando •Creative Contraptions •Einstein Compiler •Fat City •Fight Night •Flight Simulator v2.0 •Fun with Direction •GBA 2-On-2 Championship Basketball •GraphicWriter v1.1RA •Growing Up Small •House-on-a-disk •Intrigue •Jet •Jungle Hunt •Kindercomp •Knowing Numbers •Kung-fu Master •Law Of The West •Learning Well series •Letters And Words •Little Computer People •Make Your Own Murder Party •Manic Mansion •Master Diagnostics •Movie Maker •Music Construction Set •Pinball Construction Set •Pitstop •Print Shop Graphics Library Holiday •Print Shop Igs •Rendezvous •Shapes And Patterns •Silent Service •Sorcerer •Spy vs Spy I & II •Stargate •Stellar 7 •Stickybear ABCs •Stickybear Drawing •Stickybear Numbers •Stickybear Printer •Stickybear Printer Library I & II •Stickybear Townbuilder •Super Boulderdash •Temple Of Apsah Trilogy •Tomahawk •Thexder •Walt Disney's Card And Party Shop •Walt Disney's Cartoon Maker •Wings Of Fury •Word Maze •World's Greatest Baseball Game •Zork III **A.P.T.s:** Bard's Tale •Lode Runner •Might & Magic •Ultima IV •W. Disney's Card And Party Shop •Wizardry III •Wizardry IV • **Playing Tips:** •Autoduel •King's Quest •Manic Mansion •Summer Games •Tass Times In Tonetown •Thexder •Where In The World is Carmen Sandiego?

**50** December 1987 • **Features:** •Super Boulderdash APT-writer •Softkeys to Activision/MECC/ and PFS ProDOS/ software •Double F-8 ROM space w/o motherboard surgery •Ace-Apple bimodal Switch •Using Sider hard drives 3 1/2", 800K drives, & 5 1/4" drives in DOS 3.3 • **Softkeys:** •Aliens •Alter Ego •Alternate Reality •Amazing Reading Machines •Amazon •American Challenge •Arcade Album #1 •Arithmetic Critters •Award Maker •Baseball Database •Bard's Tale II: Destiny Knight •BC's Quest for Tires •Bop & Wrestle •Champ. Boxing •Champ. Wrestling •Clock Works •Commando •Computer Prep for SAT •Conflict In Vietnam •Counting Critters •Crisis Mountain •Dataquest 50 States •Deluxe Paint II •Dino Eggs •Disney Card & Party Shop •Disney Comic Strip Maker •Draw Plus •Eidolon •Electric Crayon ABCs •Expedition Amazon •Facemaker •First Letter Fun •Fish Scales •Fun From A-Z •Game Maker •GBA Champ. Basketball •GFL Champ. Football •Graphicwriter 1.0R/1.1R •Great Road Race •Hacker II •Hardball •Infiltrator II •Instant Music •James Bond 007: A View To A Kill •Jenny's Journeys •Kung Fu Master •Little People •List Handler •Manic Mansion •Mastery Arithmetic Games •Market Place •Master of Lamp •Math Rabbit •Microzine #17 •Might and Magic •Mission In Solar System •Moebius •Music Construction Set •Music Studio •Number Munchers •Paint With Words •Paintworks Plus •Path Tactics •pfs:File •pfs:Graph •pfs:Plan •pfs:Report •pfs:Write •Phonics Prime Time •Portal •Principal's Assistant •Print Shop ProDOS 8 v1.4 •Print Shop Holiday Edition •Quickflash! •Reader Rabbit •Realm Of Impossibility •Robot Odyssey I v2.0 •Rocky Horror Show •Rocky's Boots v4.0 •Saracen •Shanghai •Silent Service •Skylab •Sound Tracks •Speedy Math •Spindizzy •Street Sports Baseball •Sub-Mission •Super Boulderdash •Tass Times in Tonetown •Thexder •Top Fuel Eliminator •Word Handler •Word Munchers •Words at Work •World Karate Champ. •Writer's Choice: Elite •Zardax v5.2.1

**49** November 1987 • **Features:** •Eliminate some ProDOS erroneous error messages •Date/time without a clock card •Sector surgery: recover lost files •Generating Applesoft programs 'on-the-fly' •Product Monitor reviews • **PLUS:** How to convert List Handler files into standard text files •How to make GRAPHIC.GRABBERv3 run on the Igs •Laser 128 'absolute' RESET • **Playing Tips:** •Bard's Tale II •Conan •Donkey Kong •Hacker I •Hard Hat Mack •Orbitron •Print Shop Companion •Spellbreaker •Spy Hunter •Ultima 4 **A.P.T.s:** Infiltrator •Lode Runner •Montezuma's Revenge •Swordthrust series • **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 Comp. Game Of Good Sex •Echo 1.0 •E.D.D. 4 •GEMmaker •Hard Ball •Infiltrator •List Handler •Locksmith 6.0 Fastcopy •Magic Slate •Math Critters •Millionaire •Mind Mirror •One On One •Paintworks Plus v1.0 •Paintworks Plus v1.1 •PHM Pegasus •Portal •Quotient Quest •Reader Rabbit •Saunders's Chemistry CAI •Science Toolkit •Shanghai •Strip Poker •Super Bunny •Super Sunday •Swordthrust series •Term Paper Writer •Thief •Top Fuel Eliminator •Typing! •Up-n-Down •Willy Byte •Writer's Choice Elite v1.0 •Writing A Character Sketch •Writing A Narrative

**48** October 1987 • **Features:** •Dungeon Editor & Encounter Editor for Ultima III •APT for Shadowkeep •Softkey for Shadowkeep •Softkey for Apple Business Graphics • **Softkeys:** •816 Paint GS •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 •Deluxe Paint GS •Destroyer •Hacker II •Hacker II GS •Hardball •Infiltrator •Instant Music GS •J-Bird •Mabel's Mansion •Marble Madness •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 1.0 GS •Paul Whitehead Teaches Chess •PHM Pegasus •Poetry Express •Print Shop color version •Rambo: First Blood part II •Rocky Horror Show •Sargon III •Shanghai GS •Spindizzy •TelePorter •Temple Of Apsah trilogy •Top Draw GS •Transylvania •Ultima I •World's Greatest Baseball Game

**47** September 1987 • **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 • **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 •Champ. Golf

**46** August 1987 • **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** July 1987 • **Softkeys:** •Mouse Calc •Sands of Egypt •Number Farm •Agent U.S.A. •Wavy Navy •Kindercomp

# COMPUTIST back issues

To order, Use the back issue order form on page 42

•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** June 1987 *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 •Pikit Printer Drivers *Review*: Z-RAM Memory Expansion Board •Reading the Joystick

**43** May 1987 *Softkeys*: •Graphics Expander •Information Master •Certificate Maker •Elite •Catalyst 2.0 and 3.0 •Murder On The Mississippi •Temple Of Apshei 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 I 2: a Review *APTs* •Sword of Kadash & Rescue Raiders •Ultimaker IV

**42** April 1987 *Softkeys*: •Light Simulator •Beach-Head •Monty Plays Scrabble •Racter •Winnie the Pooh •Infocom Stuff, Kabul Spy, Prisoner II •Wizardry 1 & 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** March 1987 *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** February 1987 *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** January 1987 *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** December 1986 *Softkeys*: •Cyclod •Alternate Reality •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 IIe •Looking into Flight Simulator's DOS *Core*: •Appavarex •Installing a RAM disk into DOS 3.3

**37** November 1986 *Softkeys*: •Under Fire •Pegasus II •Take 1 (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 //c •Track Finder •Syk to Dif *Core*: •Breaking In: tips for beginners •Copy II Plus 6.0: a review •The DOS Alterer

**36** October 1986 *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** September 1986 *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

**34** August 1986 *Softkeys*: •Crisis Mountain •Terripin Logo •Apple Logo II •Fishies 1.0 •SpellWorks •Gumball •Rescue at Rigel •Crazey Mazey •Conan •Perry Mason: The Case of the Mandarin Murder •Koronis Rift *Feature*: •More ROM Running *Core*: •Infocom Revealed

**33** July 1986 *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** June 1986 *Softkeys*: •Revisiting Music Construction Set •Cubit •Baudville Software •Hartley Software •Bridge •Early Games for Young Children •Tawala's Last Redoubt •Print Shop Companion •K 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** May 1986 *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** April 1986 *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** March 1986 *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** February 1986 *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** January 1986 *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 1 •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 •Appicillin 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-PB1 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 V1.1 •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 1

**1** *Softkeys*: •Data Reporter •Multiplan •Zork •*Features* •PARMS for Copy II Plus •No More Bugs •APT's for Choplifter & Cannonball Blitz •*Copycard*'s Reviews •Replay •Crackshot •Snapshot •Wildcard

# Out-of-print back issues

To order, Use the back issue order form on page 42

Due to popular demand, these sold-out issues are available now as 'zerixed' copies, full-sized and center stapled.

**21** ■ *Softkeys*: •DB Master version 4+ •Dazzle Draw •Archon •Twerps ■ *Readers' Softkeys*: •Advanced Blackjack •Megaworks •Summer Games •College Entrance Exam Prep •Appewriter revisited ■ *Features*: •Demystifying The Quarter Track ■ *Core*: •Proshadow: A ProDOS Disk Monitor

**18** ■ *Softkeys*: •Scholastic Version of Bank Street Writer •Appewriter IIe •SSI's Non-RDOS Disks ■ *Readers' Softkeys*: •BPI Accounting Programs and DesignWare Programs ■ *Features*: •Installing a Free Sector Patch Into Appewriter IIe •Simple Copy Protection ■ *Core*: •The Games of 1984: In Review •65C02 Chips Now Available •Checksoft v2

**17** ■ *Softkeys*: •The Print Shop •Crossword Magic •The Standing Stones •Beer Run •Skyfox •and Random House Disks ■ *Features*: •A Tutorial For Disk Inspection and the Use Of Super IOB •S-C Macro Assembler Directives (reprint) ■ *Core*: •The Graphic Grabber For The Print Shop •The Lone Catalog Arranger Part Two

**15** ■ *Softkeys*: •Mastertype •Stickybear BOP •Tic Tac Show ■ *Reader's Softkeys*: •The Financial Cookbook •Escape from Rungistan •Alien Munchies •Millionaire •Plato ■ *Features*: •MREAD/MWRT Update ■ *Core*: •A Boot from Drive 2 •DB Master's Data Compression Techniques ■ *Whiz Kid*: •DOS and the Drive - Part One ■ *Adventure Tips*: •Time Zone •Mission Asteroid •Enchanter •Zork I •Ultima •Ultima II •Death in the Caribbean •Gruds in Space •Zork III •Starcross

**14** ■ *Features*: •Super IOB v1.2 Update •Putting Locksmith 5.0 Fast Copy Into a Normal Binary File •Batman Decoder Ring •A fix for DiskEdit ■ *Softkeys*: •Seadragon •Rocky's Boots •Knoware •PFS Software •Computer Preparation SAT •MatheMagic ■ *Review*: •Boulder Dash

**13** ■ *Softkeys*: •Laf Pak •Beyond Castle Wolfenstein •Transylvania •The Quest •Electronic Arts •Snooper Troops (Case 2) •DLM Software •Learning With Leeper •TellStar ■ *Core*: •CSaver: The Advanced Way to Store Super IOB Controllers •Adding New Commands to DOS 3.3 •Fixing ProDOS 1.0.1 BSAVE Bug ■ *Review*: •Enhancing Your Apple ■ *Feature*: •Locksmith 5.0 and Locksmith Programming Language.

**12** ■ *Softkeys*: •Zoom Graphix •Flip Out •Lion's Share •Music Construction Set ■ *Reader's Softkeys*: •Hi-Res Computer Golf II •Suicide •Sabatage •Millionaire •Time is Money •Type Attack ■ *Features*: Pseudo-ROMs on the Franklin Ace ■ *Core*: •Psychedelic Symphony •The CORE Disk Searcher •The Armonitor ■ *Adventure Tips*: •Cranston Manor •Enchanter •Kabul Spy •Colossal Caves •The Witness •Pirate Adventure •Ultima III-Exodus •Adventureland

**11** ■ *Softkeys*: •Sensible Speller •Exodus: Ultima III ■ *Readers' Softkeys*: •SoftPorn Adventure •The Einstein Compiler v5.3 •Mask of The Sun ■ *Features*: •Copy II Plus v4.4C: Update Of An Old Friend •Parameter List For Essential Data Duplicator ■ *Core*: •Ultimaker III •The Mapping of Ultima III •Ultima II...The Rest Of The Picture

**10** ■ *Softkeys*: •Arcade Machine •Bank Street Writer •Minit Man ■ *Reader's Softkeys*: •Sensible Speller IV •EDD IV •\*Krell LOGO •Canyon Climber ■ *Features*: •The Controller Saver •Examining Protected Applesoft BASIC Programs •Crunchlist II ■ *Core*: •Appear - Voice Aynthesis •Introducing the 65SC802 and 65SC816 Chips •Review - Dino Eggs ■ *Adventure Tips*: •Cranston Manor •Zork I •Planetfall •Mission Asteroid •Time Zone •Suspended •Critical Mass •Zork II •Castle Wolfenstein

**9** ■ *Softkeys*: •Sensible Speller •Sierra-On-Line Software •The Visible Computer: 6502 ■ *Reader's Softkeys*: •Visidex •Music Construction Set •Gold Rush •Visiterm •Cosmic Combat ■ *Features*: •Super IOB ■ *Adventure Tips*: •Pirate Adventure •Mask of the Sun •Colossal Caves •Transylvania •Death in the Caribbean •Zork II ■ *Core*: •Word Search Generator •ProDOS to DOS •ProDOS on a Franklin Ace

**8** ■ *Softkeys*: •Robotron •Legacy of Llylgamyn •The Artist •Data Factory v5.0 •EDD IV ■ *Reader's Softkeys*: •Spy Strikes Back •Hayden Software •Apple LOGO ■ *Features*: •Review of the Bit Copiers ■ *Core*: •COREfiler •ProDOS Data Encryptor ■ *Adventure Tips*: •Ulysses and The Golden Fleece •Serpentine •Ultima II •Castle Wolfenstein •Death in the Caribbean •Zork I •Zork II •Gruds in Space •Enchanter •Infidel •Serpent's Star ■ *Whiz Kid*: •How Data is Stored on Disk

**7** ■ *Softkeys*: •Zaxxon •Mask of the Sun •Crush •Crumble & Chomp •Snake Byte •DB Master •Mouskattack ■ *Features*: •Making Liberated Backups That Retain Their Copy Protection •S-C Assembler: Review •Disk Directory Designer ■ *Core*: •COREfiler: Part I •Upper & Lower Case Output for Zork

**6** ■ *Softkeys*: •Pandora's Box •Donkey Kong •Caverns of Freitag •Visifile ■ *Features*: •Program Enhancements: Quick.Bug •Personalizing A Program •Modified ROMs ■ *Review*: •Essential Data Duplicator •The CIA ■ *Core*: •Data Bases

**5** ■ *Softkeys*: •Homeward •Aztec •iBag of Tricks •Egbert II •Starcross •Hard Hat Mack •The Home Accountant ■ *Reader's Softkeys*: •Dark Crystal •Screenwriter II •Visifile •Lancaster •Bill Budge's Trilogy of Games •Sammy Lightfoot •Amper-Magic •Buzzard Bait ■ *Feature*: •Getting on the Right Track

**4** ■ *Features*: Ultima II Character Editor ■ *Softkeys*: •Ultima II •Witness •Prisoner II •Pest Patrol ■ *Adventure Tips*: •Ultima II & III ■ Copy II Plus Parms Update

**3** ■ *Softkeys*: •Bag of Tricks •Multiplan ■ *Readers' Softkeys*: •Visiplot /Visitrend •Sneakers •Wizardry ■ *Features*: •No More Bugs: The Sequel •Hidden Locations Revealed •Map Maker ■ *A.P.T.s* •Choplifter ■ *Adventure Tips*: •Cranston Manor •Strange Odyssey

**2** ■ *Softkeys*: •Magic Window II •Multiplan ■ *Features*: •Parameters for Locksmith 4.1 •Page Flipper •String Plotter •Three-D Wall Draw ■ *Core*: Checksums ■ *Input*: •Reviews of unprotected commercial software

# 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 typed in 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 61. If you wish to purchase a library disk not listed on the left (under the DISK column), used the out-of-print back issues ad on page 42.

## 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 Utilites:** 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.....

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

Address \_\_\_\_\_

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

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

VISA  MC \_\_\_\_\_ Exp. \_\_\_\_\_

Signature \_\_\_\_\_ CP61

• US funds drawn on US bank. • Most orders 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. • Send check/money order to:

COMPUTIST PO Box 110846-T Tacoma, WA 98411 (206) 474-5750

Send these issues and/or library disks:

Issue	Mag \$4.75	Disk \$9.95	Both \$12.95
60	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
59	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
58	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
57	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
56	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
55	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
54	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
53	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
52	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
51	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
50	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
49	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
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"/>
■ 21	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
20	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
★ 19	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
■ 18	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
■ 17	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
★ 16	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
■ 15	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
■ 14	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
■ 13	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
■ 12	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
■ 11	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
■ 10	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
■ 9	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
■ 8	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
■ 7	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Core 3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
■ 6	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
■ 5	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
■ 4	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
■ 3	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Core 2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
■ 2	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Core 1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Best of Hardcore Computing...     
 Only the disk is available

Core Special Combo \$10.....     
 (All three CORE magazines; no disks)

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.

- ★ Limited supply; first-come-first-serve basis.
- Not available.
- Out-of-print; only 'Zeroxed' copies for sale.

# the COMPUTIST shopper

Software Package	I I e	I I gs	Software Package	I I e	I I gs
816 Paint.....	\$45.00 <input type="checkbox"/>	\$45.00 <input type="checkbox"/>	Multiscribe 3.0.....	\$48.00 <input type="checkbox"/>	\$66.00 <input type="checkbox"/>
Alternate Reality: The Dungeon.....	\$28.00 <input type="checkbox"/>		Music Studio 2.0.....	\$52.00 <input type="checkbox"/>	
Alternate Reality: The City.....	\$20.00 <input type="checkbox"/>		Newsroom.....	\$38.00 <input type="checkbox"/>	
Appleworks.....	\$190.00 <input type="checkbox"/>		Paint Write Draw.....		\$120.00 <input type="checkbox"/>
Artic Fox.....	\$28.00 <input type="checkbox"/>		Paintworks Plus.....		\$47.00 <input type="checkbox"/>
Bank Street Writer 64K.....	\$46.00 <input type="checkbox"/>		Paintworks Gold.....		\$65.00 <input type="checkbox"/>
Bank Street Writer Plus (128K).....	\$46.00 <input type="checkbox"/>		Pegasus.....	\$23.00 <input type="checkbox"/>	
Bard's Tale.....	\$30.00 <input type="checkbox"/>	\$35.00 <input type="checkbox"/>	The Print Shop.....	\$34.00 <input type="checkbox"/>	\$40.00 <input type="checkbox"/>
Bard's Tale II.....	\$35.00 <input type="checkbox"/>		Print Shop Companion.....	\$27.00 <input type="checkbox"/>	
Bard's Tale III.....	\$35.00 <input type="checkbox"/>		Print Shop Graphics Library:		
California Games.....	\$26.00 <input type="checkbox"/>	\$26.00 <input type="checkbox"/>	Disk One.....	\$16.50 <input type="checkbox"/>	
Certificate Maker.....	\$25.00 <input type="checkbox"/>		Disk Two.....	\$16.50 <input type="checkbox"/>	
Championship Karate.....	\$14.00 <input type="checkbox"/>		Disk Three.....	\$16.50 <input type="checkbox"/>	
Chessmaster 2000.....	\$28.00 <input type="checkbox"/>		Pro-Byter.....	\$32.00 <input type="checkbox"/>	
Clip Art Library (for Paintworks Plus).....	\$20.00 <input type="checkbox"/>	\$20.00 <input type="checkbox"/>	Reader Rabbit.....	\$27.00 <input type="checkbox"/>	\$34.00 <input type="checkbox"/>
Copy II Plus v8.0.....	\$23.00 <input type="checkbox"/>	\$23.00 <input type="checkbox"/>	Silent Service.....	\$23.00 <input type="checkbox"/>	
Create with Garfield.....	\$20.00 <input type="checkbox"/>		Space Quest.....	\$32.00 <input type="checkbox"/>	\$32.00 <input type="checkbox"/>
Create w/ Garfield Deluxe.....	\$28.00 <input type="checkbox"/>		Space Quest II.....	\$32.00 <input type="checkbox"/>	
Crossword Magic.....	\$32.00 <input type="checkbox"/>		Star Fleet.....	\$35.00 <input type="checkbox"/>	
Dark Lord.....	\$15.00 <input type="checkbox"/>		Star Trek- The Kobayashi Alternative.....	\$27.00 <input type="checkbox"/>	
Dazzle Draw.....	\$40.00 <input type="checkbox"/>		Star Trek II- The Promethean Prophecy.....	\$27.00 <input type="checkbox"/>	
Deathlord.....	\$28.00 <input type="checkbox"/>		Stickybear Series:		
Draw Plus.....	\$52.00 <input type="checkbox"/>		ABC's.....	\$24.00 <input type="checkbox"/>	
F-15 Strike Eagle.....	\$24.00 <input type="checkbox"/>		Math I.....	\$24.00 <input type="checkbox"/>	
Fantavision.....	\$34.00 <input type="checkbox"/>	\$40.00 <input type="checkbox"/>	Math II.....	\$24.00 <input type="checkbox"/>	
Flight Simulator II.....	\$36.00 <input type="checkbox"/>		Numbers.....	\$24.00 <input type="checkbox"/>	
Force 7.....	\$15.00 <input type="checkbox"/>		Reading.....	\$24.00 <input type="checkbox"/>	
Halls of Montezuma.....	\$28.00 <input type="checkbox"/>	\$28.00 <input type="checkbox"/>	Shapes.....	\$24.00 <input type="checkbox"/>	
Hardball.....	\$22.00 <input type="checkbox"/>	\$28.00 <input type="checkbox"/>	Typing.....	\$24.00 <input type="checkbox"/>	
Hitchhiker's Guide.....	\$20.00 <input type="checkbox"/>		Opposites.....	\$24.00 <input type="checkbox"/>	
Jet.....	\$29.00 <input type="checkbox"/>		Thexder.....	\$23.00 <input type="checkbox"/>	\$23.00 <input type="checkbox"/>
Karateka.....	\$24.00 <input type="checkbox"/>		TimeOut Superfonts.....	\$42.00 <input type="checkbox"/>	
King's Quest.....	\$32.00 <input type="checkbox"/>	\$32.00 <input type="checkbox"/>	Tomahawk.....	\$23.00 <input type="checkbox"/>	
King's Quest II.....	\$32.00 <input type="checkbox"/>	\$32.00 <input type="checkbox"/>	Topdraw.....	\$66.00 <input type="checkbox"/>	
King's Quest III.....	\$32.00 <input type="checkbox"/>	\$32.00 <input type="checkbox"/>	Type.....	\$30.00 <input type="checkbox"/>	
King's Quest IV.....	\$32.00 <input type="checkbox"/>	\$32.00 <input type="checkbox"/>	Typing Tutor IV.....	\$32.00 <input type="checkbox"/>	\$32.00 <input type="checkbox"/>
Knight of Diamonds.....	\$23.00 <input type="checkbox"/>		Ultima V.....	\$40.00 <input type="checkbox"/>	
Legacy of the Ancients.....	\$28.00 <input type="checkbox"/>		Visualizer.....	\$53.00 <input type="checkbox"/>	\$59.00 <input type="checkbox"/>
Legacy of Lylgamyn.....	\$26.00 <input type="checkbox"/>		Where in USA is Carmen San Diego.....	\$30.00 <input type="checkbox"/>	
Leisure Suit Larry.....	\$26.00 <input type="checkbox"/>	\$26.00 <input type="checkbox"/>	Where in World is Carmen San Diego.....	\$30.00 <input type="checkbox"/>	
Lode Runner.....	\$24.00 <input type="checkbox"/>		Where in Europe is Carmen San Diego.....	\$30.00 <input type="checkbox"/>	
Macroworks.....	\$20.00 <input type="checkbox"/>		Wings of Fury.....	\$25.00 <input type="checkbox"/>	
Math Blaster.....	\$28.00 <input type="checkbox"/>		Winter Games.....	\$25.00 <input type="checkbox"/>	\$27.00 <input type="checkbox"/>
Math Blaster Plus.....	\$29.00 <input type="checkbox"/>	\$29.00 <input type="checkbox"/>	Wizardry.....	\$32.00 <input type="checkbox"/>	
Math Rabbit.....	\$27.00 <input type="checkbox"/>		Word Attack.....	\$28.00 <input type="checkbox"/>	
Math Talk.....	\$35.00 <input type="checkbox"/>	\$35.00 <input type="checkbox"/>	Wordperfect w/ Spelling Checker.....	\$95.00 <input type="checkbox"/>	\$95.00 <input type="checkbox"/>
Marble Madness.....	\$25.00 <input type="checkbox"/>	\$25.00 <input type="checkbox"/>	World Games.....	\$25.00 <input type="checkbox"/>	\$27.00 <input type="checkbox"/>
Mean 18, Ultimate Golf.....	\$28.00 <input type="checkbox"/>		Writer's Choice Elite.....	\$60.00 <input type="checkbox"/>	
Merlin 8/16.....	\$80.00 <input type="checkbox"/>		Writer Rabbit.....	\$24.00 <input type="checkbox"/>	
Millionaire II.....	\$40.00 <input type="checkbox"/>		Yeager's Advanced Flight Trainer.....	\$28.00 <input type="checkbox"/>	
Mousewrite.....	\$99.00 <input type="checkbox"/>	\$99.00 <input type="checkbox"/>			

**COMPUTIST**  
is proud to  
offer these  
**LOW software**  
prices\* for  
**Apple II and**  
**IIgs software**

\* Prices subject to  
change without notice.

Because we do not 'stock' but  
order only the software needed,  
the COMPUTIST Shopper keeps  
operating costs to a minimum  
and can pass the savings on to  
you.

In most cases, your order is  
shipped within two weeks. The  
COMPUTIST Shopper will NOT  
cash your check nor charge  
your credit card until your  
software is ready to be shipped.  
If the software you ordered is  
not available, you will have the  
option to cancel your order, or  
make an alternative selection.

### How To Order

- **US orders:** Check the box for your selection. For Apple IIgs software, check the 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.
- **Foreign Orders:** Please inquire as to appropriate shipping fees

Name \_\_\_\_\_ ID# \_\_\_\_\_  
 Address \_\_\_\_\_  
 City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_  
 Country \_\_\_\_\_ Phone \_\_\_\_\_  
 Signature \_\_\_\_\_ CP61

● Software  
SUBTOTAL \_\_\_\_\_  
 ● Plus  
SHIPPING \_\_\_\_\_  
(see above)  
 ● WA residents  
Sales Tax \_\_\_\_\_  
7.8 %  
 ■ TOTAL  
Enclosed \_\_\_\_\_

SoftKey Publishing PO Box 110816-T Tacoma WA 98411

Neal Jhala

Softkey for...

## Test Drive

Accolade

This softkey works for TDCGA.EXE dated 10/26/87 4:43 PM. It removes all copy protection from the program. We will use DEBUG for this.

From the DOS prompt:

**REN TDCGA.EXE PROGRAM  
DEBUG PROGRAM**

Now you're in DEBUG.

**D 70D6 L5**

You should see 55 56 57 06 1E. If not then you have a different version. (See note.)

**E 70D6 31 C0 C3**

**D 70D6 L5**

You should see 31 C0 C3 06 1E.

**W**

You should see Writing E080 bytes.

**Q**

Now you're back in DOS.

**REN PROGRAM TDCGA.EXE**

*NOTE: If your version is different or you wish to patch the EGA version of TDCGA.EXE, search for the hex string 55 56 57 06 1E and try changing it to 31 C0 C3 06 1E. It will probably work.*

C.J. Blanchard

I recently acquired an AE PC Transporter which is a super addition to my IIGs as it allows me to take work home when necessary. All of the software that I have tried — DBase III, Wordstar, Windows, PC Paint, Printshop, Sidekick, Enable and Grand Slam Bridge — has run without a hitch except Gunship.

Ⓟ Gunship by Microprose runs well enough but the highlighted menu bars don't work for some reason. The copy protection is well beyond my skill. I have been reading thru the IBM RDEX to see if someone else has solved this problem. The protection seems to revolve around bad blocks. There are no "INT 13" codes in the GS.COM file but the code is self modifying. While I do not understand 8086 code, running GS.COM with DEBUG and then retracing the code produces many "INT 13" instructions. (Thanks to David Alexander and his "Introduction to IBM Disk Format, Access and Copy Protection", I refer to it often.) If someone has a fix or crack for this program, I would appreciate hearing from you.

Keep the IBM RDEX going. I would be interested in a DEBUG tutorial, IBM softswitch or other important locations and general machine language tips.

Kevin Dobbins

## Protected files

I've been working for a local computer dealer for a few months as a support and training technician. One day, a salesman challenged me to list one of his small batch files. I used the "TYPE" command and got a "File Not Found" error. He laughed and walked away. After some time, I discovered his clever security lock. In place of a character in the filename (or extension), he typed "ALT KEY" -255. This had the effect of printing a blank in a directory listing. By extending this idea, you could make a filename extension be invisible, or, by using ASCII codes 126-255, make a disk appear to be garbage when it is really OK. This idea even works with sub-directory names. The possibilities are limitless! Maybe this is a well-known security device for file names, however, I thought it was interesting.

Mike Basford

## MS DOS Copy Protection

The tide against copy protection is heavy in the IBM field, with more and more magazines recommending people don't buy protected software. More companies each month drop protection, although some of the alternatives are pretty strange.

For the remaining copy protected software, here are some heavyweight tools of the trade, all of which I have and recommend.

The number one tool is from our friends at Central Point Software (CPS), the **Option Board**. This is inserted between the floppy controller and the drives, and *will copy any disk*. I have found a couple that gave me a hard time, but they fell to Copy II PC, also from CPS. By the way, the Option Board will copy most floppy disks, even if they aren't IBM (are you reading this Apple people?) I sometimes have to recopy track \$00 with Copy II Plus. A further feature of this amazing piece of hardware, is that it copies MAC disks (as well as copying files to and from MAC - IBM). It has all sorts of nice utilities such as disk editors etc..

Copy II PC will copy many disks, but it's best feature is a program called NOGUARD. This will determine the type of protection used, and, in many cases, deprotect it!

Other than these, I use several public domain programs obtained from your local BBS. These include CRACKER (a de-

compiler that generates *commented* source code from programs), FSDEBUG (an excellent replacement for DOS's DEBUG), various memory mapping programs and, of course, PC TOOLS!

## Words words words

I recently met a key programmer from one of the largest game software companies, and had an interesting chat. He was on Vancouver Island for the "Worlds Best Fishing". We talked a little about graphics, "C" programming, copy-protection, etc.

Anyway, it seems that we have beaten the protectors, and there should be virtually NO copy protection in a year or so. I hope COMPUTIST changes its format enough to survive in those times!

He told me that the main problem that the software companies have, is with the people who openly advertise pirated software for sale. They are difficult to catch and prosecute, and cost companies lots in lost sales. He was not concerned with collectors and traders of software, since most of their software would never have been purchased in the first place.

He also said that the restrictions of having to produce their software in 128K versions (ie: Apple II) held back better graphics. It seems that everyone is waiting for memory prices to come back down to Earth, and the requirements for State of Art (instead of, State of Ark) systems will be 30 - 50 MHz processors, with a minimum of 16 Megabytes of RAM. If the operating system of choice is Unix or similar, programs can quickly be ported between any two machines. We will see less differences between computers, as the best things from each system are ported to all others.

Of course, this will cause even more pressure for un-protected software, and any remaining vestiges will quickly be swept away.

I hope that this generates some replies, and maybe we could have a column where we discuss general subjects, much like a BBS in print? And please keep up the technical, and hardware related articles. Even if I don't actually build them, they provide some insight into the inner workings of my favorite machines!

Playing Tips for...

## Leisure Suit Larry in the Land of the Lounge Lizards

Sierra On-Line

In order to by-pass the annoying questions at the start of Land of the Lounge Lizards, type in your age, (lie, if you like!), and then answer the first question with ALT-X.

# IBM Reader's Data EXchange IBM

Softkey for...

## Print Shop

Broderbund

There are several versions of Print Shop around, and I'll tell you how to crack a couple of them. A third, more recent, copy is quite different, and I'm still working on it!

PS.EXE has the first layer of protection, and when it is removed, the program will run. However, when the program is run from hard disk, PSINIT.OVR has to be cracked too.

Before you start, use DISKCOPY to back-up the disk. Don't work on the original disk! Also, since DEBUG won't work with .EXE files, rename PS.EXE to PS before you start.

### RENAME PS.EXE PS

#### DEBUG PS

Start DEBUG

#### S0 9000 CD 13

Search for INT 13

XXXX:3AC6

These are the Interrupts

XXXX:3ADF

that I found.

### U 3AC6

Unassemble code to check it

XXXX:3AC6 INT 13 Test for disk

XXXX:3AC8 MOV BYTE PTR [0A91], 02

XXXX:3ACD DEC BYTE PTR [0A91]

XXXX:3AD1 JZ 3AE6

XXXX:3AD3 MOV DH, 00 <

XXXX:3AD5 MOV DL, 00

XXXX:3AD7 MOV CH, 09 Look for special nibble track

XXXX:3AD9 MOV CL, 0A

XXXX:3ADB MOV AL, 01

XXXX:3ADD MOV AH, 04

XXXX:3ADF INT 13 <

XXXX:3AE1 CMP AH, 00 Compare, and crash if not there

XXXX:3AE4 JNZ 3ACD

XXXX:1B21

### U 0479

XXXX:0479 INT 13

Look for disk in A

XXXX:047B MOV BYTE PTR [CD57], 02

XXXX:0480 DEC BYTE PTR [CD57]

XXXX:0484 JZ 04DB

XXXX:0486 MOV CL, 11 <--

XXXX:0488 MOV AH, 04

Look for special nibble track

XXXX:048A MOV DH, 00

XXXX:048C MOC CH, 09

XXXX:048E MOV DL, 00

XXXX:0490 MOV AL, 01

XXXX:0492 INT 13 <--

XXXX:0494 CMP AH, 00

Do the check

XXXX:0497 JNZ 0480

and crash if not there

### U 04BD

XXXX:04BD INT 13

Same old thing

XXXX:04BF CMP AH, 00

XXXX:04C2 JNZ 04A5

XXXX:04C4 ADD BX, 018B

XXXX:04C8 MOV CL, 05

XXXX:04C8 ES:

XXXX:04CB MOV CL, [BX]

XXXX:04CD CMP AL, 41

XXXX:04CF JNZ 04DB

XXXX:04D1 INC BX

XXXX:04D2 DEC CL

XXXX:04D4 JNZ 04CA

XXXX:04C6 MOV AX, 0000

If check is OK

XXXX:04D9 JMP 04DE

Jump to start

### A 0479

XXXX:0479 NOP

Remove Interrupt 13

XXXX:047A NOP

XXXX:047B

### A 0490

XXXX:0490 MOV AH, 00 required by AH register

XXXX:0492 NOP

XXXX:0493 NOP

Remove Interrupt 13

XXXX:0494

### A 04BD

XXXX:04BD JMP 04D6 Jump past check to start

XXXX:04BF

(On the second copy of PS I looked at, there was a second copy of this code at 1ADD, 1AF6 and 1B21. If you have this, simply repeat above modifications, with the appropriate address changes.)

One further modification is optional. The program will still ask you to put the original disk in A:, although it doesn't check. You may use PCTOOLS, or similar, to search for this message, and remove it. Leave the "Press any key" message in though, or the user won't know why the program is just sitting there.

Have fun!



At last, I'm starting to receive some letters. Let's keep on writing with the latest tips and techniques. Some of that 3 and 4 year old BBS data was really questionable. I'm sure there are old versions of popular programs floating around but most of us have the latest version of any program that we use a lot.

This issue marks the first entry in the IBM Most Wanted List. If you're having a problem backing up a program, tell us so we can add it to the list.

I'd like to put together a FREE Starter Kit for us IBM users. If you have any Public Domain Software (PDS) that you would like to donate, how about sending it to me. No copyrighted software please.

Above all, send letters telling about any neat and nifty things that you have learned. The more you write, the more others will write. We need to expand these pages and that's the only way to do it.....RDEXed

Send all contributions to:

IBM-RDEX editor  
COMPUTIST  
PO Box 110846-T  
Tacoma, WA 98411

# MOST WANTED

## Softkeys

Gunship Microprose

IBM RDEX END

## Part 2 for hard disk users

### DEBUG PSINIT.OVR

DEBUG copy of program

#### S0 2000 CD 13

Look for Interrupt 13's

XXXX:0479

XXXX:0492

XXXX:04BD

XXXX:1ADD

These 3 on second version I tried

XXXX:1AF6

# unCLASSIFIED ADS

Trade your unwanted software. Send your list of programs to trade. I have over 70 originals to trade. Byron Blystone, P.O. Box 1313, Snohomish, WA 98290.

## PUBLIC DOMAIN

Plus more!!!

100's of disks. 1000's of PRINT SHOP graphics, APPLEWORKS files, PRODOS files, EDUCATIONAL, UTILITIES, plus much more. School purchase orders welcome. Send \$1 for catalog to Continental Software Co., Box 1511, So. Glens Falls, NY 12803

## HackerDOS

Fast DOS 3.3 plus undelete, space on disk, print text file to screen/printer, fast text I/O, addr. & length of loaded files, hex/dec translation, 15 more sectors/disk. Nothing removed, takes no extra space! Support programs copy any range of sectors, change lower → uppercase, & customize DOS. Options incl. binary/text boot file, show deleted files, >35 tracks, & 2-column catalog. \$5 to: Phil Goetz  
4023 Huckleberry Row  
Ellicott City, MD 21043

The COMPUTIST SUPER INDEX now has over 3200 records and 14,000 entries! Three colossal data bases (15 fields each) list every item ever published in COMPUTIST. See advertisement - issue #57, page 45. Prices good through 12/31/88. D.R. Hopkins, 3495 W. Hoyer Pl, Denver, CO 80219

## Original Apple Software

New and used. Trade/sell. Over 100 games and adventures. Also GS software, hardware, and misc. Send stamp for list. Steve Wadsten, 20318 Fremont, Livonia, MI 48152

## How to place an UnClassified Ad

For short ads, use this order form. For larger ads, submit typed copy with appropriate instructions. Use every character space, up to 30 characters per line, we will adjust word wrap. \$5 for the first line, \$1 per line after that.

Special Graphics Instructions: The first three words of the first line are printed in bold for free. If you want other words bolded, use two spaces per letter and circle the entire word. (Bold letters count as two letters.) If you want a line centered, write CENTER next to that line. There is no charge for centering any line.

----- \$5  
----- \$6  
----- \$7  
----- \$8  
----- \$9  
----- \$10

Name \_\_\_\_\_ ID# \_\_\_\_\_  
Address \_\_\_\_\_  
City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_  
Country \_\_\_\_\_ Phone \_\_\_\_\_  
Exp. \_\_\_\_\_  
Signature \_\_\_\_\_ CP53

Total Number of lines: \_\_\_\_\_ \$ \_\_\_\_\_ . 0 0

Run my ad \_\_\_\_\_ number of times. Each run that I pay for now will cost me only 1/2 the total initial cost. \$ \_\_\_\_\_

Total \$ \_\_\_\_\_

\*Washington orders add 7.8% sales tax. \$ \_\_\_\_\_

Final Total \$ \_\_\_\_\_

Send this form and a check or money order (funds drawn on US bank only) for the entire amount to:

COMPUTIST unCLASSIFIEDS PO Box 110846 Tacoma, WA 98411

NEW!

## LOW FACTORY DIRECT PRICES!

DEALERS CALL  
UNIV & SCHOOL P.O.'S WELCOME

### MEGABYTE RAM BOARDS!

The Ultimate in RAM Power!

Super Expander GS, a 100% replacement for Apple Iigs 1 meg board. Fully Iigs hardware & software compatible. #K \$39

Super Expander E, for U.S. and European Iie. Licensed by Applied Engineering. 100% replacement for extended 80 column card. Expands Appleworks desktop to 998K and allows instant switching between Appleworks modules. #K \$59

Super Expander C, expands Iic to 1152K and Appleworks desktop to 1062K, plus allows instant switching between modules. #K \$69

Super Expander +, operates in any Ii+ slot, plus 0. Expands Appleworks desktop to 1014K, & allows instant switching between modules. (80 Col req for Appleworks). #K \$79

ABOVE w/256K, 512K, or 1MEG CALL

### QUIET COOLING FANS!

SuperFan II for the Ii+/Iie. Kensington style w/surge protect. \$26  
Super Cooler GS for the Iigs. No audio line interference. \$24

### APPLE COMPATIBLE PERIPHERALS!

64K/80 Column Board expands Iie to 128K and adds 80 columns & hi-res. \$35

Super 64K/80 Column Board w/64K \$48

Super 64K/80 Column Board w/256K call

16K RAM Board expands Ii+ to 64K \$34

128K RAM BD expands Ii+ to 192K. Combine w/Plusworks 2 for Ii+ Appleworks desktop of 136K. \$64

Plusworks 2 Software allows Ii+ Appleworks operation. (req 80 col). \$38

80 Column BD expands Ii+ display to 80 columns. (Videx compatible). \$48

Super Serial Board 100% replacement for the Super Serial card. \$48

Z80 CP/M Board opens the world of CP/M for your Ii+/Iie/Iigs. \$34

Graphic Parallel Board w/Cable does text & graphic screen prints. \$44

Parallel Board w/Cable for use where graphics is not required. \$35

Disk Drive 5 1/4 w/speed adjust, specify your model w/order. \$119

A/B Switchbox parallel or serial \$29

A/B/C/D Switchbox (par or ser) \$39

Numeric Keypad II for Iie \$34

Numeric Keypad II for Iic \$39

Joystick (specify your II model) \$17

Joystick III similar to Mach III \$24

TWO YEAR WARRANTY ON ALL PRODUCTS

CALL OR WRITE FOR OUR COLOR CATALOG

ADD \$3 SHIPPING/COD OKAY, VISA/MC NO EXTRA FEE



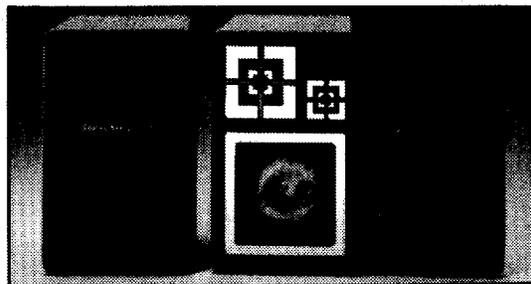
# NEXO DISTRIBUTION (619) 474-3328 10am-6pm/Mon-Fri

914 East 8th Street, Suite 109, National City, California 92050

Apple, Appleworks and Macintosh trademarks of Apple Computer, Inc. Bose & Roommates are trademarks of Bose Corp.

## SOUND SYSTEM™ 2

NEW! 2-Way Stereo Amplified Speaker System for your Apple Iigs, Iie, Iic, Ii+, and Macintosh



More versatile than the Roommates by Bose, this thunderous amplified speaker set will make all Apple speech, music, games, and other sound programs come to life on any Apple model computer.

Prepare to be blown away when you plug in this super powered speaker system, because the advanced design amplifier of the SoundSystem 2 provides over 3 times the power of any amplifier board.

And the sound? You won't believe your ears, because the built in 2-way stereo super heterodyne speakers, coupled with state of the art filtering, provide an enhanced high quality sound, crystal clear to any listener.

Release the full sound potential of your Apple computer with the versatile SoundSystem 2. You never knew your Apple could sound so good!

### FEATURES:

- Built in advanced design amplifier.
- Volume control (you'll need it).
- 2-way super heterodyne speakers.
- State of the art noise filtering.
- No slot operation.
- 100% Apple software compatible.
- 100% Apple hardware compatible. (specify your model with order)
- Complete Documentation.
- Full two year warranty.

LOW FACTORY  
DIRECT PRICE \$129  
(PAIR)

# The Hacker's Ultimate Copy & Deprotection Utility

# Super IOB Collection !

ALL of our Super IOB controllers (through March 1988) 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:

▶ Three disks (supplied in DOS 3.3). Each disk contains at least 60 Super IOB Controllers including the standard, swap, newswap and fast controllers. Also included is version 1.5 of Super IOB, the Csave 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.\*\*

\*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.

## 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 newswap 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, Muplot, Newsroom, Newswap controller, Penguin Software, Print Shop Graphic Library, Print Shop, Rendezvous with Rama, Rockys' Boots, Sargon III, Sea Dragon, Shieia, 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, Newswap 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

**Volume 3** of the Super IOB collection covers all the controllers from COMPUTIST No. 39 through No. 53. The following 89 controllers are on volume 3:

2400 A.D., Alternate Reality: The Dungeon, Address Book, American Challenge, Apple Gradebook V2.6, Arcade Album #1, Arcade Boot Camp, Aztec, Bard's Tale II, Beachhead, Beyond Castle Wolfenstein, Black Magic, Blue Powder/Grey Smoke, Borrowed Time, Castle Wolfenstein, Cat-N-Mouse, Catalyst 2.0 & 3.0, Centipede, Championship Golf, Championship Wrestling, Charlie Brown 123'S, Colonial Conquest, Comprehension Skills, Computer Preparation for the SAT V3, Coveted Mirror, Coveted Mirror RWTS Capture, CPU68000V.4Q2, Cranston Manor, Crime Stopper, Dam Busters, Decimals, Destroyer, Dino Dig, Earth Orbit Station, Encyclopedia Britannica, Escape, Fish Scales, Foundation Course in Spanish, Frog, GFL Football, Goonies, Great American Cross-Country Road Race, Handlers, H.E.R.O, Inferno, Information Master, J-Bird, Jenny's Journeys, Joe Theisman Football, Kindercomp, Kung Fu, Light Simulator, Manuscript Manager, Master Diagnostics II, Mastering Math, MECC, MECC Software, MECC Swap, MECC Without RWTS, Missing Links, No Error Check & No DOS, Ogre, One On One, Oregon Trail, Penguin Software, PFS ProDOS, Pitfall II, Puzzles&Posters, Racter, Randamn, Reading Style Inventory, Ring Quest, Science Toolkit, Science Toolkit Rev1, Skyfox, Snooper Troops, Speed Reader II, Spy's Adventure, Starcross, Super Bunny, Temple Apshai Trilogy, The American Challenge, Transylvania, Troll Courseware, Ultima I, Ultima IV, Wizard's Crown, Word Attack, Word Maze, World Karate Championship, Rings of Zilfin, Zorro

### ■ Yes, please send me The Super IOB Collection

Includes all three disks with Super IOB version 1.5, COMPUTIST #32, PLUS a reprint of "Disk Inspection and the Use of Super IOB".

- US/Canada/Mexico — All three volumes ... \$24.00  
 Other Foreign — All three volumes ..... \$30.00  
 US/Canada/Mexico — Volume #3 only ..... \$9.00  
 Other Foreign — Volume #3 only ..... \$12.00

Send to: **Super IOB Collection**  
**PO Box 110846-T**  
**Tacoma, WA 98411**

or call: (206) 474-5750 for  or  orders.

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

Address \_\_\_\_\_

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

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

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

Signature \_\_\_\_\_ CP61

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

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 within 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 Division. •DLM software •EA (Electronic Arts) software •Einstein Compiler 5.3 •Escape From Rungistan •Financial Cookbook •Flip Out •Hi-res Computer 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 • Csaver • The Core Disk Searcher • Modified ROMs.

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

Address \_\_\_\_\_

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

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

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

Signature \_\_\_\_\_ CP61

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 per book 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) 474-5750